

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE 1995	3. REPORT TYPE AND DATES COVERED March 1995 Interim Report	
4. TITLE AND SUBTITLE The O.C.A Schmidt telescope CCD camera controller			5. FUNDING NUMBERS F61708-93-W0076	
5. AUTHOR(S) Alain Maury (Amaury@obs-azur.fr)				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Observatoire de la Cote d'Azur 06460 Caussols France www.obs-nice.fr			8. PERFORMING ORGANIZATION REPORT NUMBER SPC-93-4007	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) EOARD PSC 802 Box 14 FPO AE 09499-0039			10. SPONSORING/MONITORING AGENCY REPORT NUMBER SPC-93-4007	
11. SUPPLEMENTARY NO				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for Public Release			12b. DISTRIBUTION CODE A	
12. ABSTRACT (Maximum 200 words) This document describes a new CCD camera controller adapted to Schmidt telescopes. If several large CCD detectors can be adapted in the focal plane of a large Schmidt telescope, deeper digital images can be obtained, the operating cost of a CCD camera is several orders of magnitudes smaller than that of glass photographic plates. This also opens new ways of using Schmidt telescopes; i.e., real time detection of celestial sources. This report contains the following sections: Requirements analysis Description of the Loral CCD442A CCD Description of the camera controller Physical implementation of a mono CCD camera Physical implementation of a multi CCD camera Appendix 1: Electronic schematics of the O.C.A. CCD camera controller Appendix 2: Data sheets of the main components of the O.C.A. controller Appendix 3: 87C750 based sequencer software				
13. SUBJECT TERMS EOARD, Foreign reports, Telescopes			15. NUMBER OF PAGES	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT UNCLASSIFIED	18. SECURITY CLASSIFICATION OF THIS PAGE UNCLASSIFIED	19. SECURITY CLASSIFICATION OF ABSTRACT UNCLASSIFIED	20. LIMITATION OF ABSTRACT UL	

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18
298-102

20030213 099

93-4007

The O.C.A. Schmidt telescope CCD camera controller

Alain Maury
March 1995

This document describes a new multi CCD camera controller adapted to Schmidt telescopes.

It contains the following sections :

- Requirements analysis
- Description of the Loral CCD442A CCD
- Description of the camera controller
- Physical implementation of a mono CCD camera
- Physical implementation of a multi CCD camera

The schematics of the controller can be found in Appendix 1, and the data sheets of all the major components in Appendix 2. Information related to the microcontroller and its software can be found in Appendix 3. Appendix 4 contains printouts showing the cosmetic quality of the 10 CCDs we purchased in 1994.

Reproduced From
Best Available Copy

Copies Furnished to DTIC
Reproduced From
Bound Originals

The main reason leading to the replacement of photographic plates by CCD detectors in wide field telescopes is the progressive discontinuation of photographic plates by Eastman Kodak. The second factor is that the price of large CCDs is decreasing, and it becomes possible to build a large multi CCD camera for a relatively low price. If several large CCD detectors can be adapted in the focal plane of a large Schmidt telescope, then deeper digital images can be obtained. This can also open new ways of using Schmidt telescopes, i.e. real time detection of celestial sources. This project has been mainly oriented toward the real time detection of Earth Grazing Asteroids (EGA).

Requirements analysis

The first step is to determine the different properties of the electronic system (the "controller") required to drive several CCDs at the focal plane of a Schmidt telescope.

Schmidt focal sphere : A typical "big Schmidt" telescope focal plane is in the order of 30 to 35 cm wide on a side, with a focal length not too different from 3.15 meters, giving a scale of slightly larger than 1 arc minute per mm, or 15 microns per arc second. In the Schmidt telescope optical design, the focal locus is a sphere, and this requires to bend the photographic plates, or to use field flattening elements in front of flat detectors, unless they intercept a part of the sphere small enough to be considered flat (typically less than 10 mm wide). In our case, using individual CCDs which are 30.72mm on a side would lead to a defocalisation of +/- 37 microns across the CCD diagonal. Such a defocalisation is usually clearly visible in focus plates, and could also cause deterioration of image point spread function (psf) across the field (which in turn could give inaccuracies in psf based photometry).

Space requirement : The small size of CCD detectors, combined with the large size of the camera dewars and electronics have prevented their use inside Schmidt telescopes until now when technology and reduced price has allowed to design multi CCD cameras. The group working at the University of Tokyo is the current leader in the development of multi CCD arrays for wide field telescopes with an array of 4x8 1K CCDs in use at Las Campanas Observatory in Chile and another 2x8 1K at the Kiso Schmidt telescope. Elsewhere, when CCDs have been used with Schmidt

telescopes, they have mainly been regular observatory cameras used either at the "newtonian" Schmidt focus, (Brorfelde, CTIO, KPNO, Xinglong) i.e. the camera being mounted alongside the telescope's tube, and the optical path being diverted by a flat mirror at 45° angle, or at the Cassegrain focus (Uppsala), an hyperbolic mirror bringing the optical path behind the main mirror. A "true" Schmidt CCD camera has to contain many large chips, using very compact electronics and non obstrusive cooling systems. Usual cameras are mounted at the back of the telescope, and space is not a major problem. A typical plate holder is less than 10 centimetres thick, and cannot generate much heat in the optical path without affecting the quality of the images.

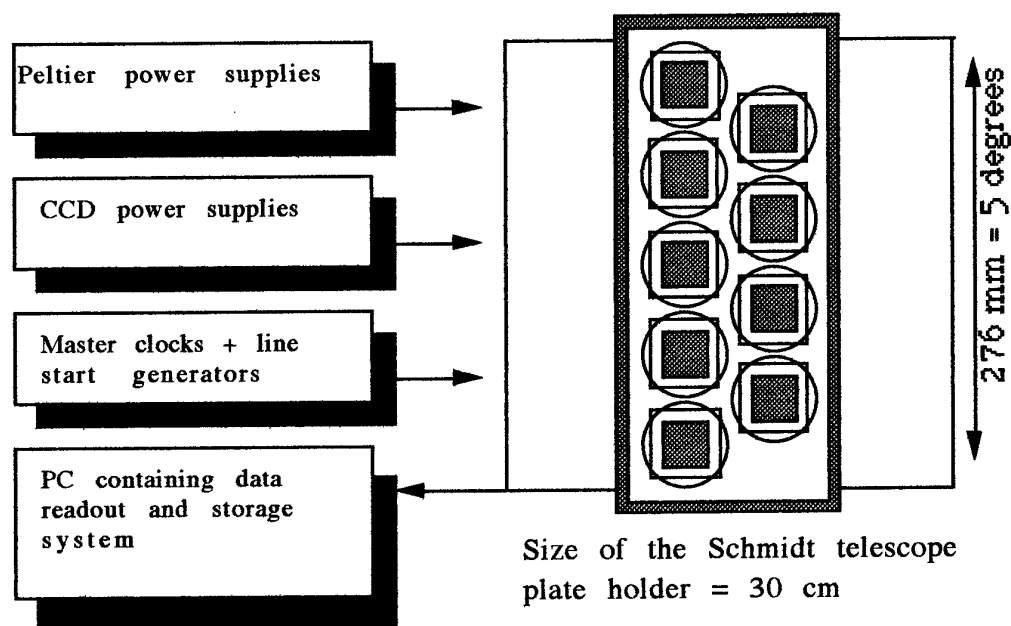
Budget requirement : Because these telescopes are usually not "hot new" telescopes, and because of the high price of the technology used, it is important to find ways to dramatically lower the price of a multi CCD camera so as to be able to convert from photography to CCDs. For example, a typical price for a grade 1 thinned 2048*2048 pixel chip from SITE/Tektronix is in the order of \$80,000. A typical commercial price for a single camera controller without CCD is in the order of \$10,000. The complete price of such a multi CCD camera using commercial hardware, not taking into account the price of the associated computer system could easily cost several years of the regular operating budget of these instruments.

Mechanical requirement : Since flat field correction lenses should be used, it is preferred to place small plano convex lenses in front of each CCD instead of a single large lens which would have a much larger chromatic aberration. We use silica plano convex lenses of 1040 mm radius of curvature. Simulations made by an optician here have shown that distorsion and other aberrations created by such a small lens are negligible. Because the focal surface of a Schmidt telescope is spherical, each CCD must be positioned precisely tangent to this sphere. This involves a mechanism able to place the CCD in height (focus), local tilt in X and Y, rotation (in order to get the CCD lines oriented in respect of the right ascension and declination), and translation in X and Y in order to place the CCD correctly with respect to the other CCDs. This mechanism has to be relatively compact in order to fit inside the telescope's focal plane. This can be done either during assembly, the CCDs

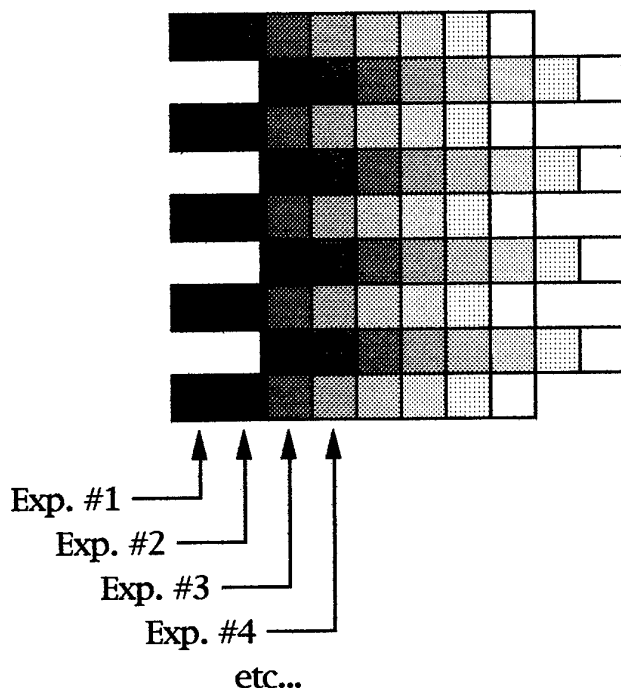
being glued in place, (preferably the right one), or the CCDs can be installed on adjustable support, which can be adjusted afterwards if necessary. Test images, similar to focus plates allow to measure the relative positioning of the CCDs and permit precise corrections to be made. An iterative alignment process should be completed in less than a week.

CCD arrangement : Off the shelf CCDs are generally not bootable. We chose to use a staggered array design (see following page), which allows independent assembly of each individual CCD camera. 9 CCDs will be used in this camera, giving a vertical field of view of 5 degrees, i.e. compatible with the regular field of view of a Schmidt telescope. Each CCD is separated from the others in the matrix by exactly one CCD field. This way a given CCD will image an area of the sky comprised in declination between the one above it and the one below it in the other column with a small overlap. Dealing with the data flow of such a camera is possible using today's computer technology.

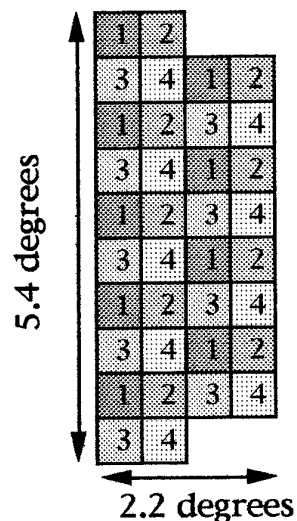
Positioning of 9 CCD modules in the focal plane of the Schmidt telescope



RA shift mode : Several exposures with a 30 arc minutes drift in right ascension give several overlapping images of the sky.



Quad-exposure mode : 4 individual exposures recreate a contiguous 11 square degrees exposure



Readout rate : A typical Schmidt telescope has a very fast F/ratio, leading to bright images which are always photon noise limited. When a small amount of Moon and cirrus are present in the sky, it becomes very difficult to obtain images which are not saturated. Under these conditions, it can be proven that the readout time of the CCD must be relatively short. Astronomers tend to read CCDs at a typical 40000 pixels per seconds rate so as to preserve a low readout noise. Increasing this rate increases the readout noise of the camera, but it is necessary to realise that the value which needs to be improved is the final signal to noise ratio (SNR) of the image. And because of our fast F/ratio, time is better spent collecting photons than reading them with the uttermost precision. The slight loss in SNR is easily compensated by a slightly longer integration, much shorter than would have required a long readout time. The total time required for the exposure is the exposure time " T_e " plus the readout time " T_r ". The readout time is equal to the total number of electrons collected divided by the electron flux per second. This number of electrons is equal to the

number of incident photons on the detector times its quantum efficiency in the given passband.

$$T = T_e + T_r = \frac{E}{\psi} + T_r \quad (1)$$

A simplified expression for the Signal to Noise Ratio of the sky in a given CCD exposure, not taking thermal current into account, is :

$$SNR = \frac{E}{\sqrt{E + RON^2}} \quad (2)$$

Where RON is the readout noise of a CCD controller. It decreases with the square root of the readout time. If the readout noise has a given value at a given readout time, the readout noise for another time is given by :

$$RON_t = RON_o \times \sqrt{\frac{T_o}{T}} \quad (3)$$

Squaring equation 2, and inputing equation 3 in order to express E in function of a "reference" readout time gives equation 4 :

$$E^2 = SNR^2 \times (E + RON_o^2 \times \frac{T_o}{T}) \quad (4)$$

Solving equation 4 in E gives equation 5 :

$$E = \frac{SNR^2 + SNR \times \sqrt{SNR^2 + 4RON_o^2 \times \frac{T_o}{T}}}{2} \quad (5)$$

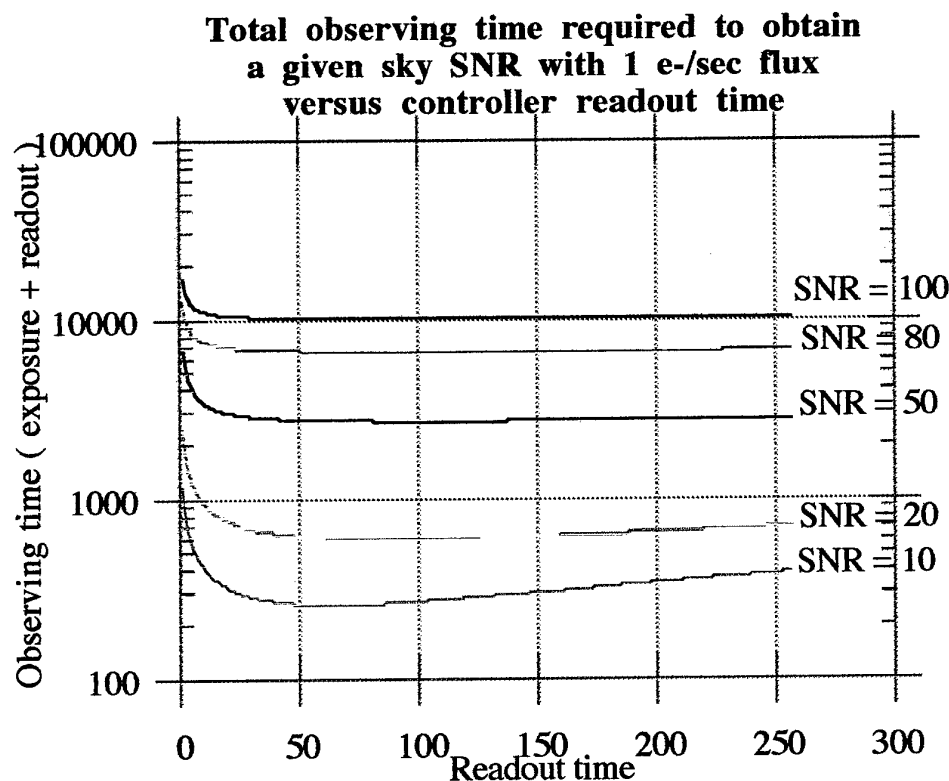
Using equation 1 and 5, we can obtain equation 6 which gives the total exposure time versus the photon flux, the controller "reference" readout noise and the readout time :

$$T = \frac{SNR^2}{2\psi} + \frac{SNR}{2\psi} \times \sqrt{SNR^2 + 4RON_o^2 \times \frac{T_o}{T}} + T_r \quad (6)$$

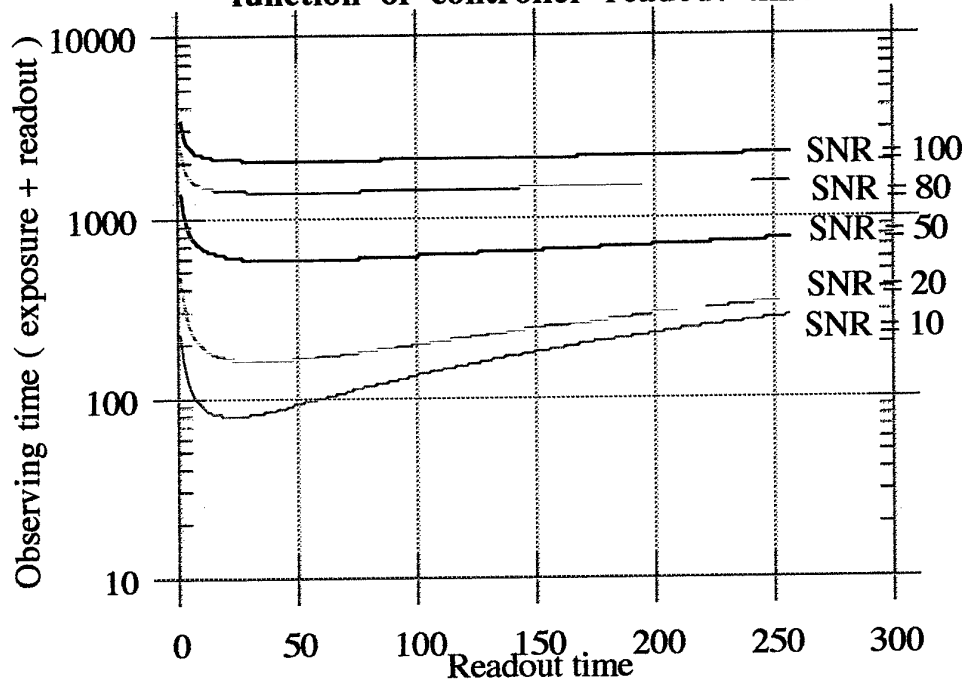
I ran a short simulation using a spreadsheet program using equation 6 with typical values in order to plot the required observing time (exposure plus readout) versus readout time for different sky signal to noise ratio. To

obtain these tables I used the case of a controller which would have a readout noise of ten electrons at 35000 pixels per second readout rate (28 microseconds per pixel). This would cause a 2K CCD to be readout in two minutes. I took 3 different electrons fluxes which are representative of the average sky brightnesses at our site in different conditions (without and with filters, depending of the filter).

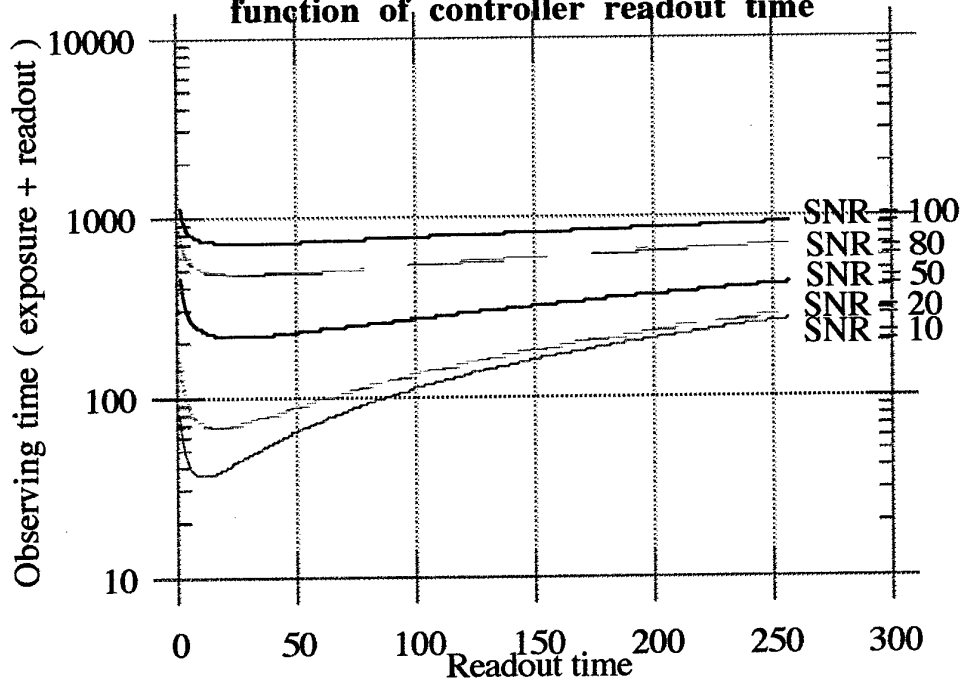
I obtained the data of the fourth plot using the optimum points obtained in each curve of the first three. They show obviously that a good CCD controller should be able to adjust its readout rate with the expected sky flux of the exposure being read. It also means for example that an exposure taken with an optical filter should not use the same readout rate as an unfiltered image, provided that telescope time is considered important or expensive. This optimisation of telescope time should be mandatory on large, expensive, telescopes.

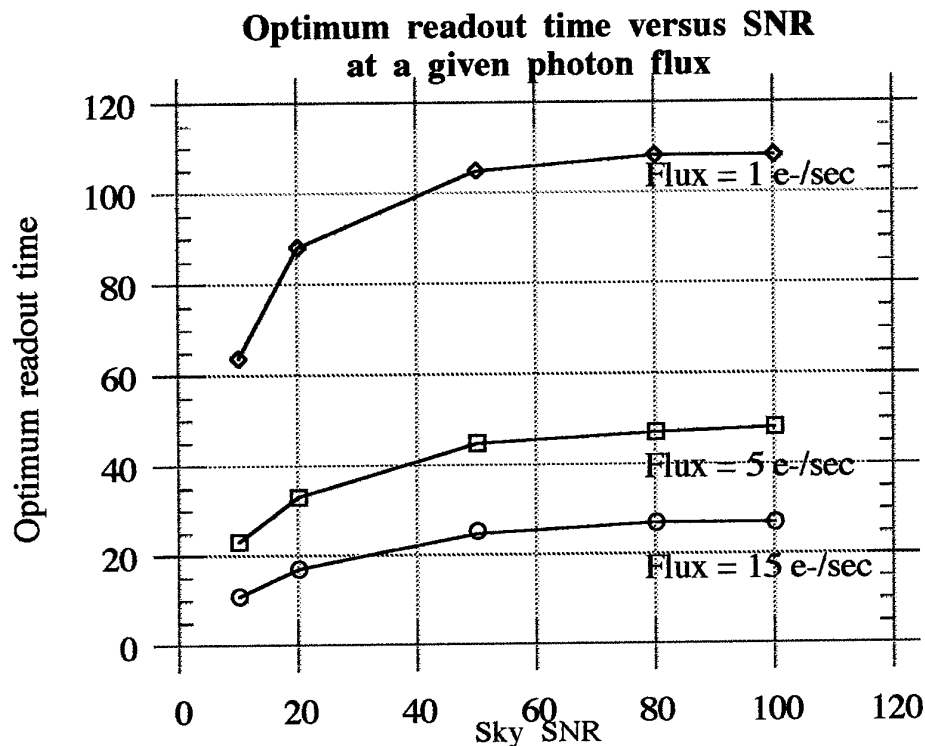


**Total observing time required to obtain
a given sky SNR with 5 e-/sec flux in
function of controller readout time**



**Total observing time required to obtain
a given sky SNR with 15 e-/sec flux in
function of controller readout time**



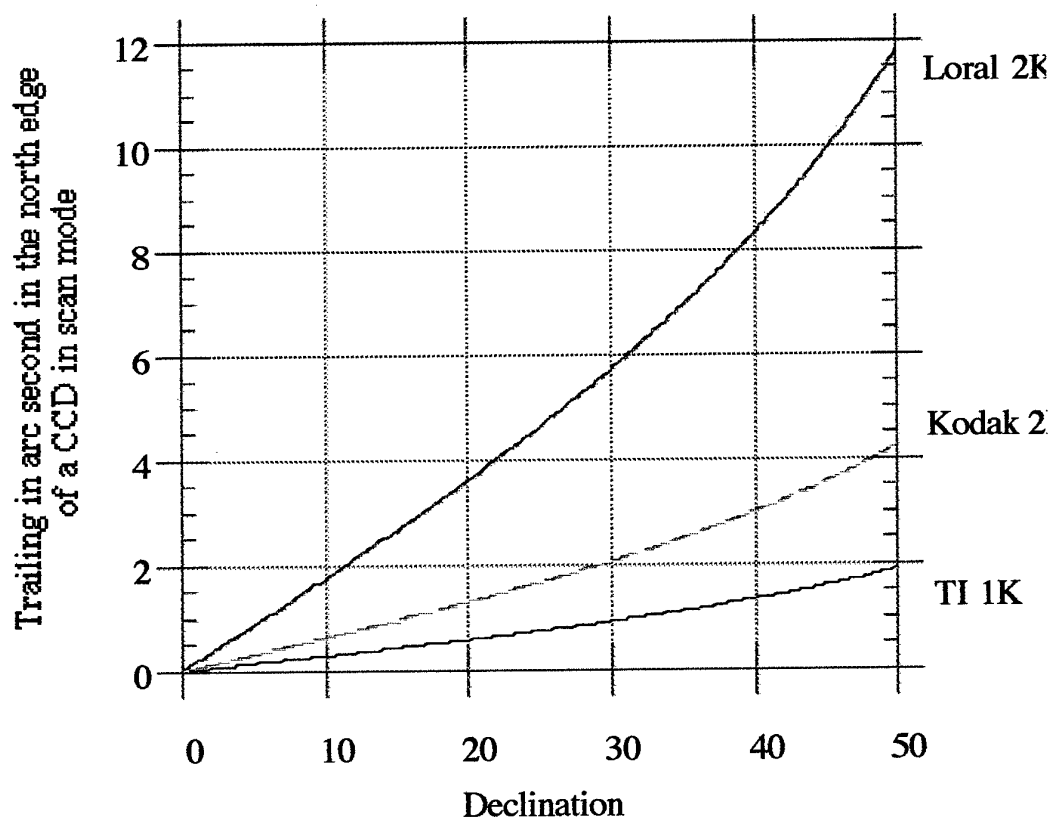


Scan mode : In order to decrease the effect of loss of observing time because of CCD readout periods while covering large sky areas, it is also possible to use the CCD in scan mode. In this mode, the electronic charges are shifted across the CCD in synchronicity with the drift of the stars across the CCD, and a continuous readout is performed. This mode is also called Time Delayed Integration (TDI). If the telescope is at rest (so called sidereal scanning), the frequency of charge shift is proportional to the pixel scale, and the sky's rotation period divided by the cosine of the declination. From this, we can understand that this scanning frequency is different between the bottom and the top of the telescope field of view which are 5 degrees apart on the sky. Hence each individual CCD must be clocked at a different rate. Because of the asynchronicity between each detector, great care must be taken in order to avoid crosstalk between each CCD. There are other effects which are to be taken into account. The scanning performs a projection of a curved sky onto a flat detector. Two effects occur because of this : differential trailing is caused by the fact that the ideal scanning speed should be different between the top and the bottom of a CCD, and field curvature occurs because at high declination a star will not stay on the same line during its motion across the CCD. The

beginning of the scan image shows a signal ramp caused by the fact that objects have integrated during a time which depended on their position on the chip when the shutter opened. Similarly, there is a ramp down at the end of the scan. The following diagram plots the trailing in arc seconds in the upper part of the CCD when clocked so that the central pixels are correctly drifted with 3140mm of focal length for 3 types of CCDs, i.e. Loral 2K (15 microns pixels, 30.72 mm on a side, 134 seconds of exposure time in sidereal scanning with our telescope), Kodak 2K (9 microns pixels, 18.4mm, 81 seconds) and TI 1K (12 microns pixels, 12.3 mm, 54 seconds).

If α is the field of view of the CCD, δ the declination of the image, the trailing in arc second (if the angles are expressed in degrees) is :

$$t = 3600 \times \alpha \times \cos\left(\delta + \frac{\alpha}{2}\right) \times \left(\frac{1}{\cos\left(\delta + \frac{\alpha}{2}\right)} - \frac{1}{\cos(\delta)} \right)$$



In practice, with the CCD we use, differential trailing is visible at

declinations higher than 10 degrees, and becomes unbearable at declinations higher than 25 degrees, as shown in the following diagram.

With smaller CCDs, the range of accessible sky is much higher, but the limiting magnitude is unfortunately also much smaller. In our case, field curvature is not seen at declinations higher than 35 degrees, so is not a real problem. A big advantage of scanning is related to the fact that since a given pixel in the final image is the average of all the pixels in a given line, images tend to be much cleaner in terms of cosmetic quality.

One of the requirements is that the readout time of a line be shorter than the time interval between two line transfers. One can increase the readout time (thereby decreasing readout noise) to the largest possible value. This is another argument for being able to change the controller readout rate on the fly. On the other hand, it is possible to move the telescope toward the east so that the line transfer interval decreases to line readout time. In our case (pixel acquisition time of 5.5 microseconds, line readout time of 11.35 milliseconds), one can scan at the equator at about 6 times higher than sidereal rate, and cover an area of 48.6 square degrees per hour, to the expense of course of an integration time reduced to 23 seconds. This mode is very interesting in order to detect fast moving objects.

The main limitation in our case is the fact that the sky zone which can be scanned is limited in declination around the celestial equator.

In order to avoid these effects there could be three possible solutions :

- Scan along great circle (any circle containing the center of the celestial sphere). This is not possible in our case since the motorisation system of our telescope does not allow continuous motion in declination. Replacing the declination drive for example with a direct motor drive would be a possible solution.
- As shown above, use smaller CCDs, since this will limit the viewing angle of the CCD. This solution is not practical since smaller CCDs provide shorter exposure times in scan mode or less angular field in stare mode. Using small CCDs increase the cost of the camera per square centimeters of detector surface, since larger chips tend to cost less per sq. cm. than small ones and that each small CCD has to have some electronics around it.
- Use a fast readout controller, so that the proportion of time spent reading out the CCD in stare mode be relatively small compared to the exposure time. The limit of this system is that readout noise increases with

the square root of readout time, and that fast conversion time analog to digital converters of high accuracy are very expensive, bringing the price of the system much higher. I used a compromise by choosing a converter which main application is audio conversion, thereby allowing a low price, fast readout rate (1 pixel / 5 microseconds), and high accuracy (18 bits precision, truncated to 16 with 14 bits of linearity). In term of time spent, scanning starts to be more efficient with our controller as soon as the length of the sky is larger than 6 degrees.

Reliability : It seems clear to many users that the most often encountered failures are related to power supplies and to connections inside the camera. The actual tendency is to limit the number of boards of the camera electronics to the minimum, and to use connectors soldered to the printed circuit board whenever possible.

Anti blooming : Because the field of view of each CCD at our telescope (34 arc minutes), and because their limited dynamic range, there are always bright stars in the field of view which end up being saturated. The typical saturation magnitude in a 2 minutes unfiltered frame is around magnitude 13. The charges bleed along the column giving the familiar aspect of bright stars in CCD images. Apart from the purely cosmetic problem, this may cause bright stars to hide other faint stars, and creates two more serious problems : The detection software will tend to detect fake stars across the blooming. In turn, this fake stars will tend to be aligned in successive frames, which will create fake asteroids detections. The second problem is even more serious : The best astrometric references are those of the Hipparcos catalogue, and are all brighter than the 10th magnitude. These stars are fully bloomed and are not measurable. The second best choice for an astrometric catalogue is the Space Telescope Guide Star Catalog, which is known to have precisions of only 0.3" in the best cases, compared to 0.001" for the Hipparcos stars. This is why a Schmidt telescope camera controller must include a provision for an anti blooming system. Such a system involves using a peculiar clocking pattern of the CCD during integration. This mode, also called partially inverted mode has been invented by Jim Janesick at NASA's Jet Propulsion Laboratory. Tested thoroughly on Loral CCDs by a group at the University of Bonn led by Dr Reif, it has also the very interesting property of doubling the potential well of the CCD. The penalty is that the

thermal current is multiplied by 8 compared to the regular MPP mode operation.

Temperature control : This leads to the last consideration of this "wish list", i.e. that of temperature control. To decrease the CCD thermal current it is necessary to refrigerate it. The rule of thumb is that the thermal current doubles every 7 degrees C. There are two classical ways of cooling a CCD : Thermoelectric cooling and liquid nitrogen cooling. The first technique is able to lower the CCD temperature to -50° in the best cases (lower temperatures can be obtained with smaller CCDs, but 2K CCDs are relatively large), and the other is typically used at -100° which removes all the problems associated with thermal current, since it is virtually inexistent at such temperatures (thermal current of 0.8 e-/hours have been measured for partially inverted CCDs at such temperatures). Whereas thermoelectric cooling is inexpensive, they are limited in their lowest temperature, and the heat generated by the Peltier modules has to be evacuated outside the telescope. On the other hand, liquid nitrogen systems are more expensive on the long run (up to 10 liters of LN2 may be used per night), but are very easily temperature regulated. We are investigating thoroughly both avenues, and will decide after more testing which of the two configurations will be used. Temperature stability is very important in precision photometry. Typically, good photometry requires temperature stability better than 0.5° , and this level of stability seems difficult to achieve using peltier cooling. The first mono CCD camera built using this controller is Peltier cooled.

We can derive from these consideration that a CCD camera controller adapted to a Schmidt telescope has the following properties :

- it has to be very compact
- it should not generate excessive heat (Cmos parts whenever possible)
- it should be able to clock and read its several CCDs in a totally independent manner.
- It should be built so as to minimise the crosstalk between each CCD.
- It should allow positioning of the different detectors in 6 degrees of freedom (height, tilt in X and Y, translation in X and Y, rotation)
- It should allow stare and scan mode with fast readout (200k pixels/seconds) using dynamic anti blooming mode.
- Finally it must be relatively inexpensive to build.

Apart from these "must have" considerations, we added a few bells and whistles, some of which are usually found in other controllers, some others are quite new.

- Possibility to use the camera in 2x2 binned mode.
- Possibility of reading only a sub window of the frame. The main use of this mode is to perform a fast readout of a predefined portion of the image, for example to test that a given star or star pattern is visible before starting a series of exposures. Several simultaneous windows could be user defined and this could allow fast photometry for example.
- Tests mode allowing to test for traps in the device, to test full well potential using J. Janesick method.
- Geostationary satellite detection mode, which is a partially scan/stare method which we designed and should try very soon here.
- On the chip autoguiding, following an idea expressed by F. Harris (U.S.N.O. Flagstaff).

These modes are described in more details in appendix 3.

Description of the Loral CCD442A CCD

We chose to use Loral 442A CCDs. Their characteristics are listed below

Size	2048*2048
Pixel size	15 microns
Physical size	31.72 mm
Angular field of view	34.5 '
Number of CCDs/5 degrees	9
Pixel scale	0.979 "
Full well potential	220,000 e- in partially inverted mode
Price per sq. mm	\$2.12

These CCDs are produced in Milpitas (CA - USA) by Loral Fairchild. (1801 Mac Carthy Bd, Milpitas CA 95035, Tel 408 433 2550, Fax 408 433 2508, contact person Mr Jim Johnson). These CCDs are thick front side illuminated chips, which do not have a good blue sensitivity, but overall provide a quantum leap compared to photography since a 2 minutes exposures detect stars which would have required more than 30 minutes of exposures using a photographic plate.

Price considerations led to the use of grade 4 CCDs. The current price is \$2,000 per chip. They should have a relatively high number of cosmetic defects, but most of these are hot spots (pixels generating a very high thermal signal), which dissapears when the CCD is cooled down (lower

than -30°C) and the CCD becomes almost as good as a grade 1 CCD, with only a few defects such as dead lines and the like. However, the price difference is such that 9 grade 4 CCDs are less expensive than a single grade 1 device. In fact, the production of these CCDs have made such progresses that most of the chips sold as grade 4 are in fact grade 3 or better. The cosmetic quality of the 10 chips we purchased and received from Loral are visible in appendix 4. These CCDs have two readout registers, but because of a manufacturing options, only one can be read at a time, i.e. it is not possible to read the CCD as two $1\text{k} \times 2\text{k}$ CCD. Their amplifiers have a peculiar double stage structure optimised for low noise at fast readout time.

Description of the camera controller

In order to achieve a fast readout rate, and because of the asynchronous scan mode, it was decided to use individual low cost controllers for each camera module linked together by a single master board able to give orders to each modules.

Each controller is made of two main boards. One contains all the power supplies and bias generation for the CCD and is usually located outside the telescope's tube. This euroboard sized card fits into a G64 backplane. Such backplanes are commercially available. They use DIN41612 connectors. The connection of this backplane are given in a following figure. A 34 conductor flat ribbon cable connects to the other board. No periodic signal which could perturb the CCDs are routed on this cable. The connection plan of this cable is given in a following figure. This power board contains voltage regulators to provide for $+5\text{v}$ and ± 15 volts. It contains also voltage followers to generate adjustable power supplies. This board is inspired by the design of the Palomar controller (Gunn et al P.A.S.P. 99, 518) which we have used at the O.C.A. for many years. It contains outputs which can be set between $+$ and $- 15$ volts, and four others which can be set between 0 to 24 volts. We followed two suggestions by Dr F. Harris at the U.S. Naval Observatory Flagstaff, i.e. instead of using LF347 as in the original design, we use pin/pin compatible OPA470 which have a much lower noise. The higher voltage generators layout have also been slightly modified . The voltage reference is still an LM399. Potentiometers controlled voltage sources are more stable, less noisy and less expensive than D/A based voltage sources. It is nevertheless envisaged

to build a D/A based voltage board in order to automatically test the CCDs and find the best setting of a given CCD in laboratory.

Connexion to the euroboard backplane.

Pin	Name	New	Pin	Name	New
1b	GND	GND	1a	GND	GND
2b	A8	Reset	2a	A0	32MHz
3b	A9		3a	A1	
4b	A10		4a	A2	
5b	A11		5a	A3	
6b	A12		6a	A4	
7b	A13		7a	A5	
8b	A14		8a	A6	
9b	A15		9a	A7	
10b	*BRQ		10a	BGRT	
11b	*RRQ		11a	RGRT	
12b	*BGACK		12a	*HALT	
13b	Enable		13a	MCLK	
14b	*RES		14a	*VPA	
15b	*NMI		15a	RDY	
16b	*IRQ		16a	*VMA	
17b	*FIRQ		17a	R/*W	
18b	IACK		18a	Halt Ack	
19b	*D12		19a	*D8	
20b	*D13		20a	*D9	
21b	*D14		21a	*D10	
22b	*D15		22a	*D11	
23b	*D4		23a	*D0	
24b	*D5		24a	*D1	
25b	*D6		25a	*D2	
26b	*D7		26a	*D3	
27b	*Berr.		27a	*Page	
28b	Chain In		28a	Chain Out	
29b	+5 Batt.	+30v	29a	-5V	
30b	-12v	+20v	30a	+12V	-20v
31b	+5v	+10v	31a	+5V	+10v
32b	GND	GND	32a	GND	GND

Notes :

- Pins 1 a and b, 29a, 30,31 and 32 a and b are used for power supplies. Please note the inversion between the polarity of 12->15 volts signals (-12 becomes +15 and +12 becomes -15v)
 - Never use pin 28 a and b. In the original G64 bus, they are used as a way to daisy chain signals (i.e. 28b of a given board is connected to 28a of the next one on the backplane and so on).
 - All the other pins can be used to send reset, clocks and communication signals from the master board to the power boards
- We use an extension board in order to be able to set the tensions on the power board.

Connexion of the 34 flat ribbon cable.

1	VOD	2	VATG+
3	VRT	4	RG+
5	VRD	6	VA3-
7	VOG	8	VOSG+
9	VRG-	10	VH3+
11	VOSG-	12	VH3-
13	+15Va	14	-15Va
15	VA1+	16	GNDa
17	VH1+	18	VH2-
19	VH2-	20	VA2+
21	VH1-	22	VA1-
23	GNDd	24	VA2-
25	GNDd	26	+5Vd
27	P0-1	28	P0-0
29	Reset	30	P0-2
31	GNDd	32	GNDd
33	32 MHz	34	GNDd

Notes :

The negative voltages ramps of the clock signals all connects to the NC (normally closed) pins of the Max333A analog switches. The positive ones connects to the NO (normally open) pins of the switches.

The other board is 60mm x 150mm long. It contains all the electronics driving the CCD as well as its acquisition chain. The sequencer is a 87C750 Philips microcontroller. In the early phases of developement of

this project, I chose to use an IFX780 Flexlogic circuit made by Intel, but it proved to be not flexible enough to allow easy reprogramming on the fly in the different modes required by the camera. The 87C750 is a limited version of the industry standard 80C51 microcontroller, with a master clock running at up to 40 MHz. This allows to clock the I/O ports with pulses as short as 375 ns. It is a small integrated circuit (24 pin DIL), having just the requisite functions for our task. This part and its programming is fully described in Appendix 3.

The microcontroller communicates via a synchronous serial line implemented with 3 of the I/O pins to either a PC or a Master board, which is also an 80C51 based board able to drive up to 15 camera.

Other I/O pins generate the timing required by the CCD. Because of the anti blooming mode, it is required that the vertical drivers be relatively slow. This allowed us to use simple analog switches to drive both the vertical and horizontal clocks of the CCD. I used Max333A switches, which receive both voltage ramps of the CCD clock from the power supply board, a logic signal coming from the 87C750, and drive the CCD pin through an RC filter.

I/O port assignment :

port 1 :

pin 0 : A1	Vertical clock 1
pin 1 : A2	Vertical clock 2
pin 2 : A3	Vertical clock 3
pin 3 : NC	No connect
pin 4: NC	No connect
pin 5: NC	No connect
pin 6: NC	No connect
pin 7 : Casc	Cascade

Note :

- Pins 3 to 6 of port 1 which are unused are brought to a small header which could be used for various purposes, such as controlling a shutter or rotating a filter wheel.
- Pin 7 is used to put the AD converter in the cascade mode, this allows to send the result of the conversion on both channels as a single 32 bits word. This allows to read channel 2, i.e. perform a temperature read.

port 3 :

pin 0 : H1	Horizontal clock 1
pin 1 : H2	Horizontal clock 2
pin 2 : H3	Horizontal clock 3
pin 3 : OSG	Output Source Gate
pin 4 : RG	Reset Gate
pin 5 : CL1	1st Clamp
pin 6 : CL2	2nd Clamp
pin 7 : STCVT	Start convert

Port 1 drives the three vertical clocks, and port 3 drives all the signals related to horizontal clocks and pixel conversions. In order to avoid jitter, it was decided not to use interrupts in the program. Another reason for this was that the 16 bits timer is too fast for our purpose.

The other bias voltages for the CCD come from the power supply board, and are linked to the CCD via an RC filter.

The acquisition chain is made of 2 OPA627 operationnal amplifiers by Burr Brown. A load resistor is of course connected to the video output of the CCD. Then comes a coupling capacitor hooked to an analog switch to provide an input clamp. This capacitor is brought to a divider bridge made of two resistors which need to be adjusted in order to clamp the first video level to ground. The first op amp has a gain which is usually set around 20. The output of this op amp goes through another clamp active during the first video level. The second op amp is just a voltage follower. The analog switch used for clamping is a Maxim DG445. It has a low charge injection. The Analog to Digital converter is a Burr Brown part labeled DSP102, and is derived from the PCM1750, except its control is much simpler and its outputs are adapted to Digital Signal Processors (DSP). The DSP 102 is an 18 bits 5 microseconds conversion time and 14 bits linearity double converter. Input level must be between + and - 2.75 volts. We use it as a 16 bits converter. The clamp signals of the acquisition chain as well as the start convert signal are generated by the 87C750. There is an option called "cascading" which allows reading of both channels and emission of the converted values as a single 32 bit word. I use this mode to read the temperature through an AD592 analog device temperature sensor connected to channel B of the converter. The "cascade" pin which allows this mode is controlled by the 87C750. The DSP102 requires 8 and 16 Mhz clock to work at its maximum conversion rate. I chose to drive the 87C750 at 32 MHz and use a binary divider (

74HCT93) to generate the 16 and 8 MHz from the 32 MHz master clock. This clock is synchronous to all the cameras used in the focal plane. The A/D converter requires + and - 5 volts which are locally generated from the + and - 15 volts through low power 78L05 and 79L05 voltage regulators. The three outputs of the converter are emitted through fiber optics emitters made by Toshiba. These parts have the advantage of being directly TTL compatible. This board is actually routed using DIL circuits, but could be rerouted in a much more compact size using Surface Mount Technology versions of the different circuits, except for the microcontroller and the DSP102, which would be purchased in 24 pins flat packages, and the fiber optics drivers which are not available in any other packages than those actually used. It is also likely that if a faster converter becomes available (but in the same price and quality range than the DSP102), it will replace the existing one. Analog Device is said to sell sometimes in 1995 a 16 bits low cost 2.5 microseconds converter which will be called AD7882.

The connection of the controller board to the CCD is made through a regular DB25 connector. Through this connector, 8 wires (in fact 2x4) for two peltier coolers are brought to an 8 pin header.

<u>Connexion of the DB25 connector to the CCD head</u>			
1	+5va (AD592)	14	VINB (AD592)
2	GNDa	15	Video Out
3	Peltier1	16	A1
4	Peltier1	17	A2
5	Peltier2	18	A3
6	Peltier2	19	OSG
7	Peltier3	20	RG
8	Peltier3	21	H1
9	Peltier4	22	H2
10	Peltier4	23	H3
11	VATG+	24	VOD
12	VRT	25	VRD
13	VOG		

Notes :

- Pin 1 and 14 connect to the temperature probe attached to the CCD.

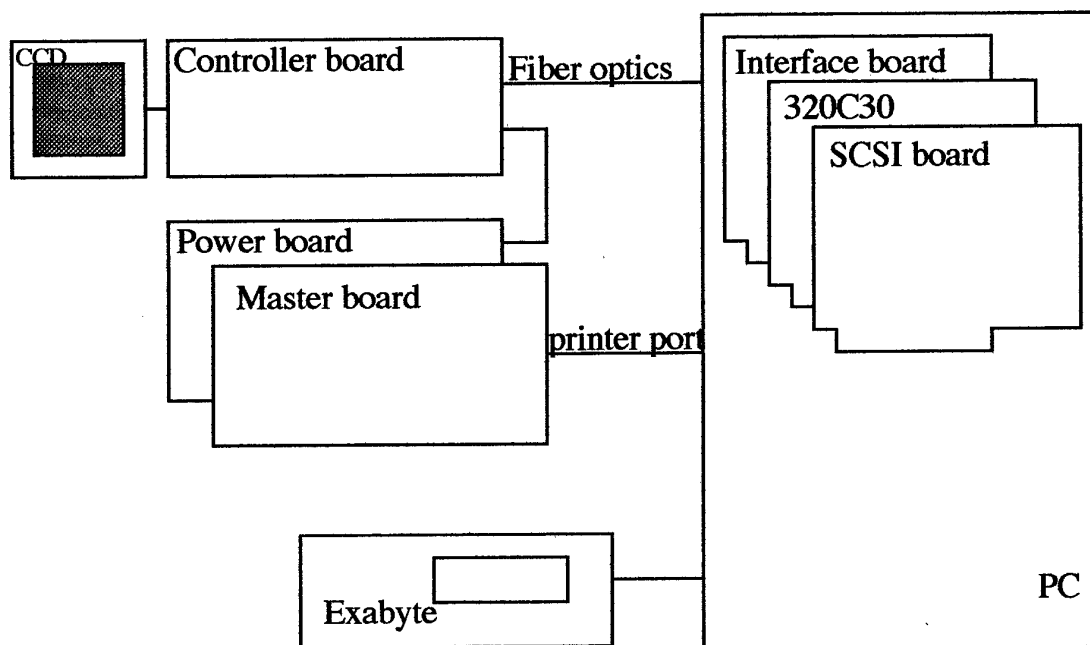
- Pin 3 to 10 are double pins connected to the two peltier cooling units. Two pins are paired together (3 with 4, 5 with 6, and so on) since high currents flow through them.
- Pin 16 to 23 are CCD clocks
- Pins 12, 13, 24 and 25 are CCD bias voltages.
- Pin 11 (VATG+) connects to the Array Transfert gate of the aplifier side which is not in use. This biasing prevents charges of going to that horizontal register. The unused amplifier is otherwise used the same way as the other one. ATG of the working amplifier is the same as A3 of the CCD.
- If all the other ones are working correctly, pin 15 should find some usefulness. :-)

CCD clocking :

It follows the manufacturer's documentation. Please refer to it in Appendix 2 for more information.

Implementation of a single CCD camera

Using this set of board, it is possible to build a single CCD camera. The schematics are given in the following figure. The master board is replaced by a simple clock generator and the control of the sequencer is done using a PC parallel port. The readout is organised around a DSP320C30 evaluation module to which a very simple PC board which is a fiber optics to TTL converter is added. This is the configuration under which this system has been tested. The software which controls the camera and performs the readout is integrated in a PC package called SPEEDI. This package was started in 1993 using GNU C++ which is a public domain C++ compiler having a DSO extender, and has been upgraded since to a multi window program which can directly control an Exabyte tape drive, NFS disks, run different cameras, and provide all the functions we wanted to have in such a package.



Implementation of a multi CCD camera

A system able to drive easily 15 CCDs can be implemented using the following design :

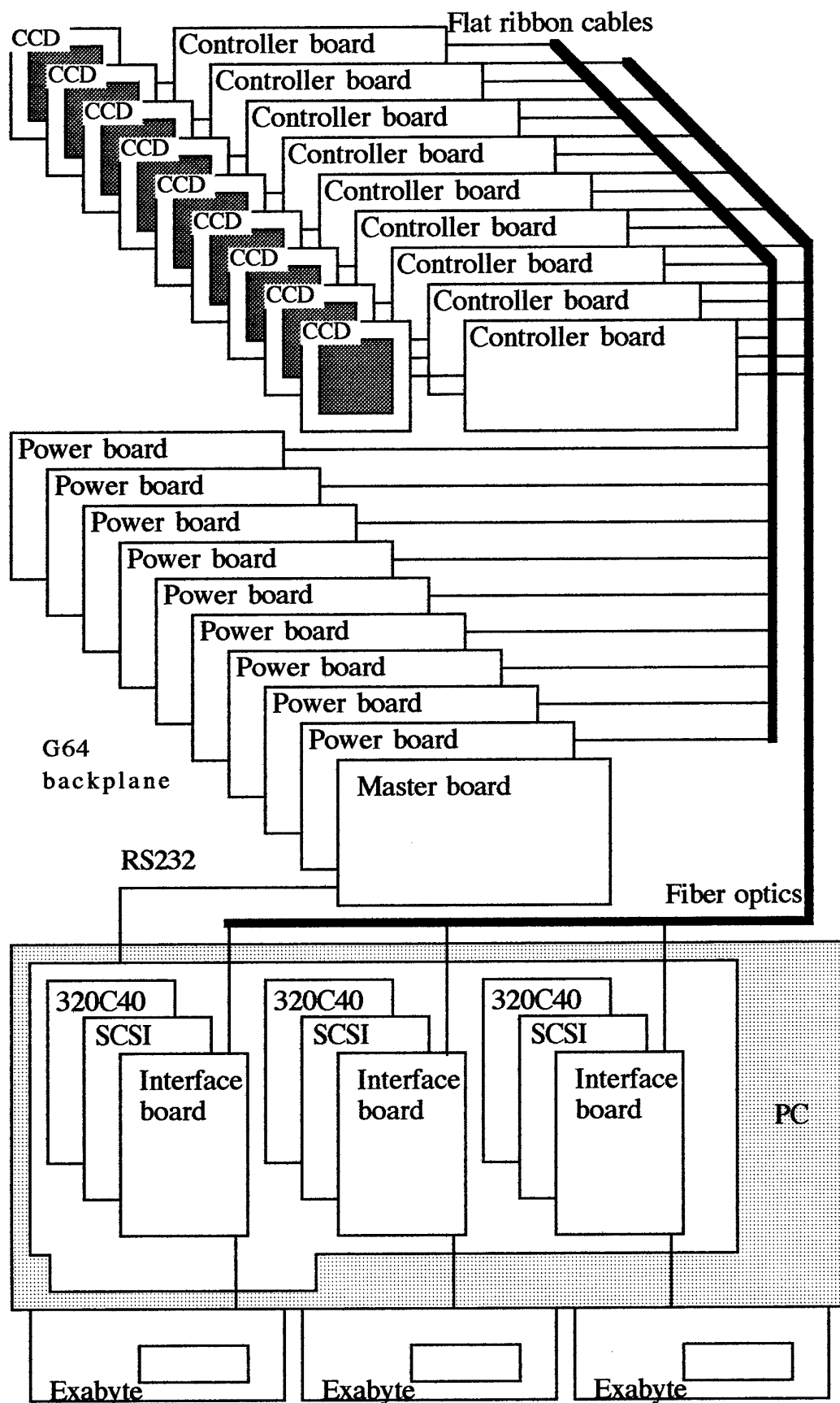
CCD control :

Each CCD has its power and controller board. The power boards all reside inside a eurorack using a G64 backplane. This rack is installed outside the telescope's tube. Flat ribbon cables are routed along one of the spider vane of the telescope. This backplane supplies each power board with +10, +20, -20 and +28 volts as well as ground. A master board, also on the backplane supplies a synchronous 32 MHz clock to each power board, as well as a general reset and serial interface to each module. It contains an 8051 microcontroller which is driven by an RS232C interface by the control PC. The connection of each module to a given backplane signal is made through wire wrapping connections. On the master board, the clock and reset signals are driven through TTL gates in order to get the required throughput. The clock line of the serial interface is synchronous to all the CCDs hooked to the system and generated by an I/O line of the microcontroller. The data line is generated using the other 15 I/O lines of the master board microcontroller. When the PC sends an order to this master board, it tells whether the order is relevant to a given CCD or to all the CCDs, the master board programs the CCDs adequately and waits for other orders. If a given order is to be sent to only a single camera, while the data clock runs for all camera modules, the data comm

pin of the other cameras transmit a zero which are converted to a no-operation order.

CCD readout :

Our idea is to use 320C40 parallel DSPs for readout. A single DSP module will readout three camera modules. An interface, very likely based on yet another 8051 will convert and tag 3 18 bits serial lines into a single parallel 8 bit port compatible to the 320C40 ports. A given pixel will be converted in a 24 bits word with its highest 4 bits being the number of the camera from which it comes. We have contracted a DSP board manufacturer in order to produce a SCSI interface for a TIM40 module (a module which contains a 320C40 as well as local memory). This interface also contains up to 32 megabytes of 0 wait states memory. We have already written code to read a given pixel, perform real time photometric correction, and cosmic ray removal. We were not able to port our real time stellar detection yet, but this should be achieved within a few months. When this system will be operationnal, it will be able to read pixels when they are available, use the DMA to store the data in the right buffer, perform a storage of the raw data to an Exabyte tape drive, perform real time photometric correction, cosmic ray removal, as well as detection. It will be possible to download data to the PC for test purpose only, and list of detected objects will be sent to our workstation for analysis.



Appendix 1 :
Electronic Schematics
of the O.C.A. CCD
camera controller

There are three Orcad based designs related to this controller :

SNGLPOW relates to the power supply board.

CCDANAL relates to the main board, i.e. sequencer and analogic chain.

CCDSUP relates to the CCD support, the small printed circuit board that supports the CCD inside its dewar.

Other schematics which are included in this design is

PELTIER, which is the schematics of the current regulated power supply we use for peltier modules. Since this handles relatively high current, it is soldered directly on a heat sink.

CONTROL, which is the board which runs a single CCD camera. We wire wrapped it, so there is no printed circuit board for it.

These, as well as information related to the programming of the sequencer will be available through anonymous ftp at [rossini.obs-nice.fr](ftp://rossini.obs-nice.fr) as a single file called OCACCD.zip. This file will be regularly updated as more functions are programmed into the sequencer, and if new version of these boards are manufactured (for example SMT components). It is highly recommended to ftp these files rather than to use the files included in this report. Also it is better to take contact with use ([maury @ ocar01.obs-azur.fr](mailto:maury@ocar01.obs-azur.fr)) since we have already have those boards fabricated and that the prices for these boards will be less expensive here since the tooling for their fabrication has already been paid for.

CCD POWER SUPPLY

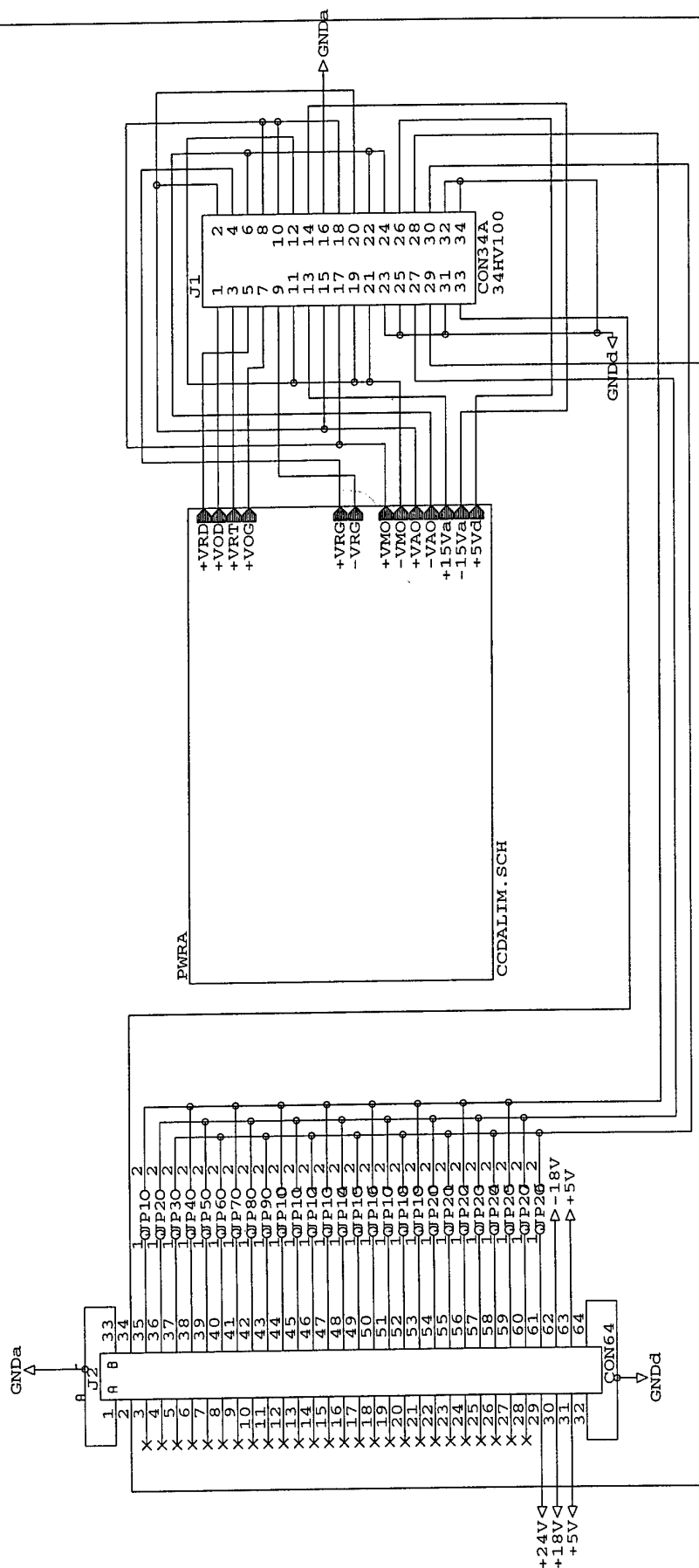
Revised: November 17, 1994

Revision: 1.1

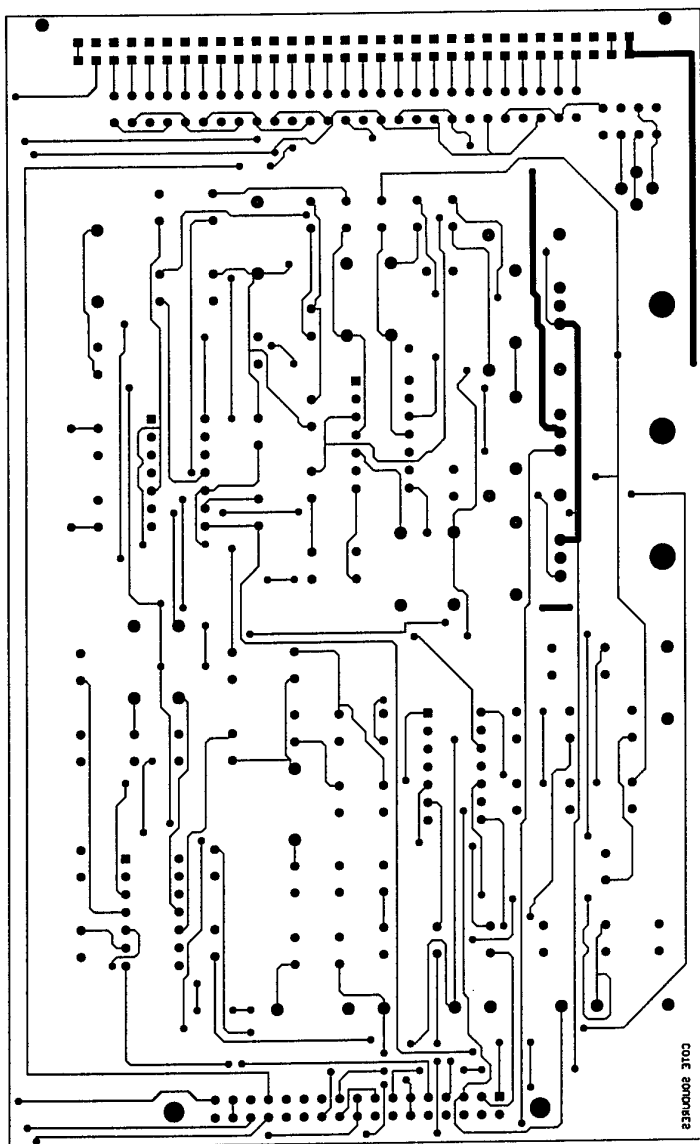
March 6, 1995

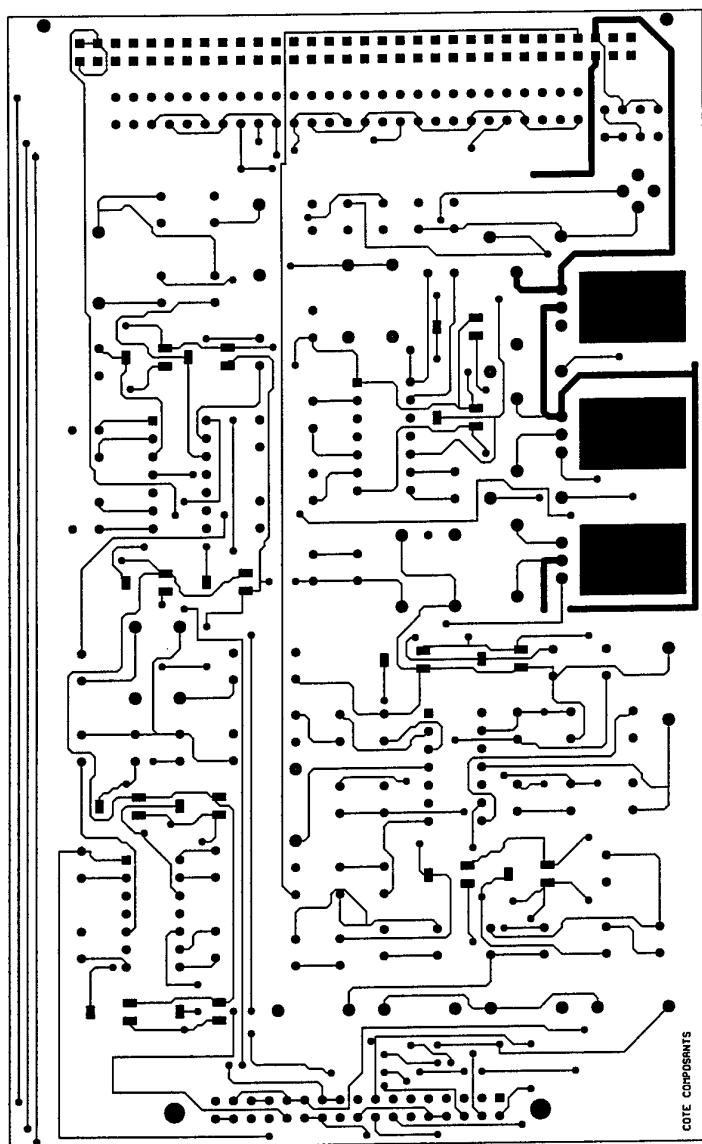
Bill Of Materials

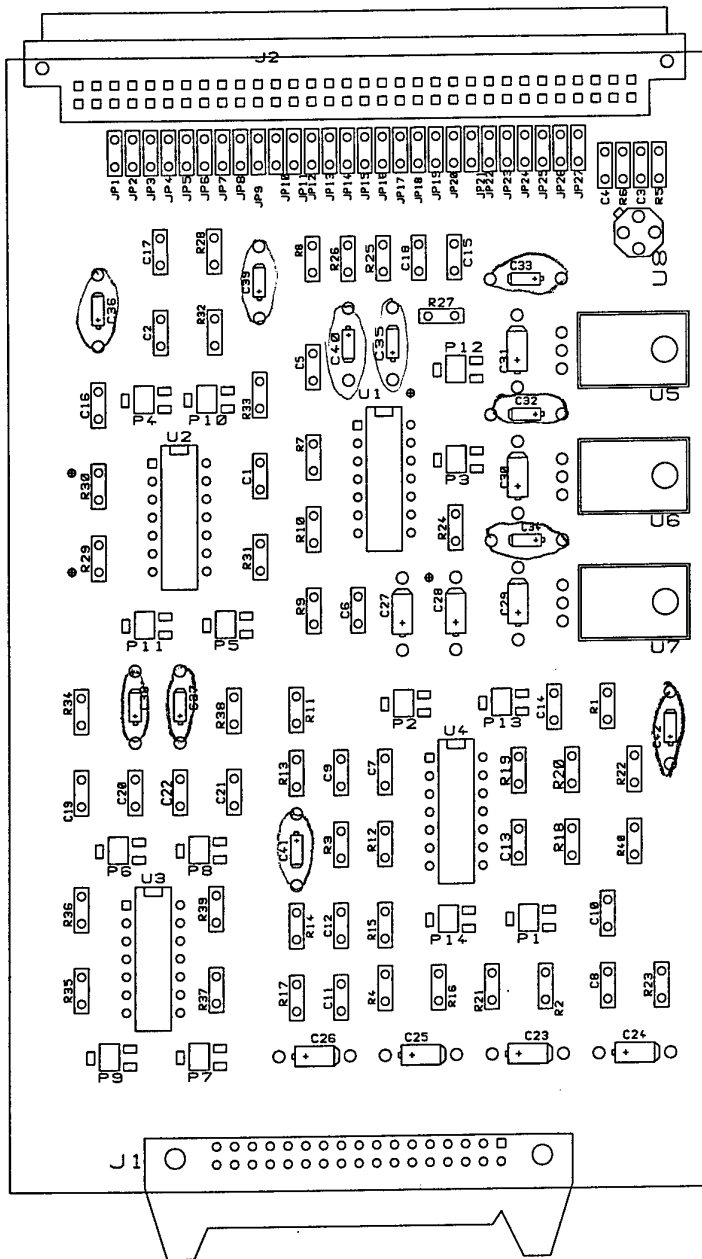
Item	Quantity	Reference	Part
1	18	C1,C2,C3,C4,C5,C6,C7,C8, C11,C14,C15,C16,C17,C18, C19,C20,C21,C22	1uF
2	4	C9,C10,C12,C13	10nF
3	9	C23,C24,C25,C26,C27,C28, C29,C30,C31	100uF
4	11	C32,C33,C34,C35,C36,C37, C38,C39,C40,C41,C42	10uF
5	27	JP1,JP2,JP3,JP4,JP5,JP6, JP7,JP8,JP9,JP10,JP11, JP12,JP13, JP14,JP15,JP16,JP17,JP18,JP19, JP20,JP21, JP22,JP23,JP24,JP25, JP26,JP27	SHORT
6	1	J1	CON34A
7	1	J2	CON64
8	14	P1,P2,P3,P4,P5,P6,P7,P8, P9,P10,P11,P12,P13,P14	100K
9	4	R1,R2,R3,R4	22
10	1	R5	10.0
11	1	R6	3.01K
12	5	R7,R11,R17,R20,R23	10.0K
13	1	R8	7.32K
14	1	R9	12.1K
15	1	R10	6.98K
16	4	R12,R16,R19,R21	20.0K
17	4	R13,R15,R18,R22	47k
18	8	R14,R25,R26,R28,R32,R34, R38,R40	10
19	10	R24,R27,R29,R30,R31,R33, R35,R36,R37,R39	1k
20	4	U1,U6,U7,U8	LF347
21	1	U2	7815
22	1	U3	UA7915
23	1	U4	7805
24	1	U5	LM399

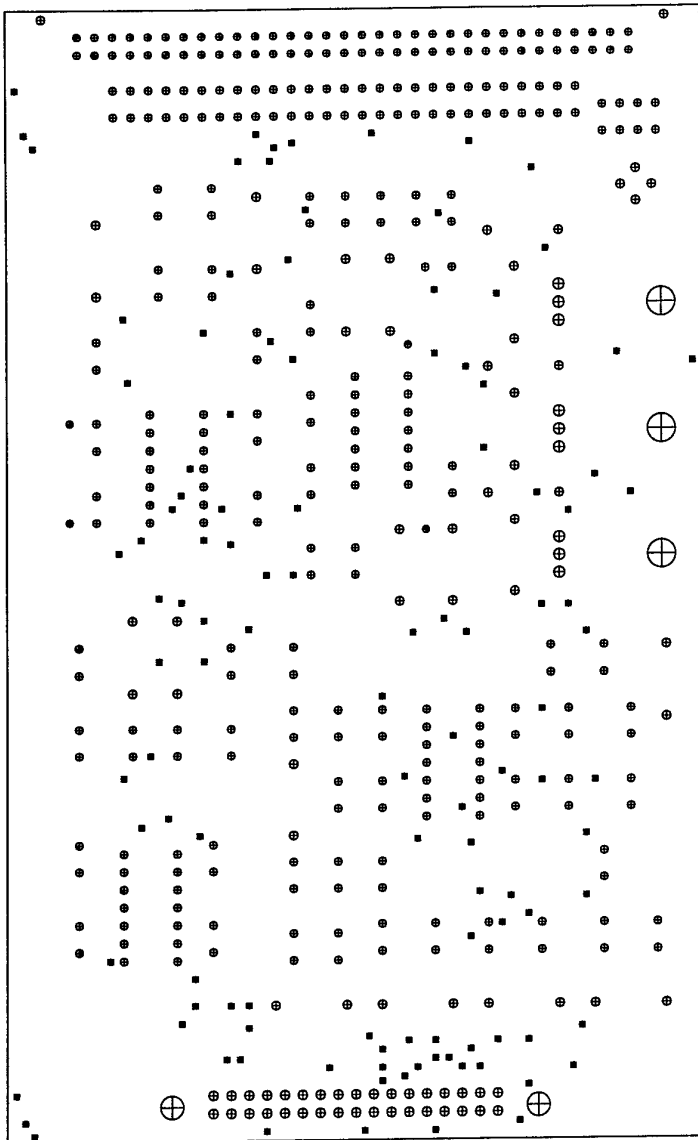


OBSERVATOIRE DE LA COTE D'AZUR			
Title		CCD POWER SUPPLY	
Size	Document Number	REV	
A4		1.1	
Date:	November 17, 1994	Sheet	1 of 2









ANALOG BOARD

Revised: March 2, 1995

Revision: V1.0

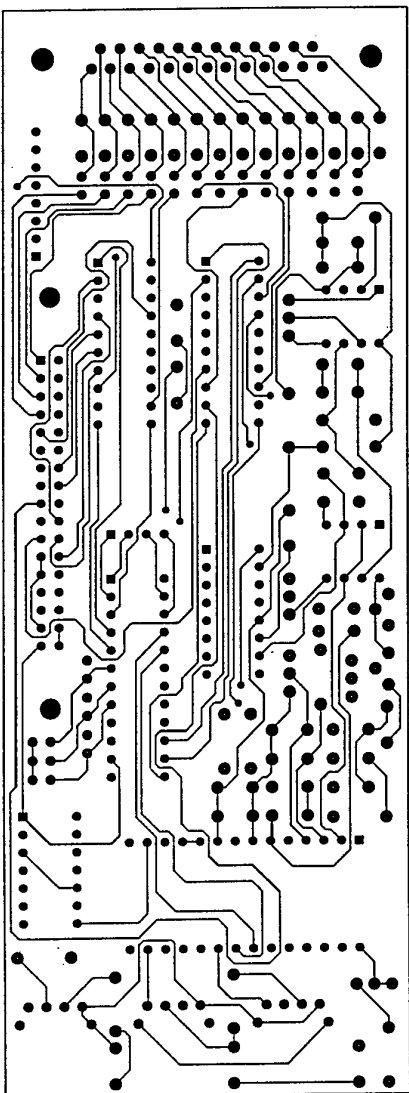
Bill Of Materials

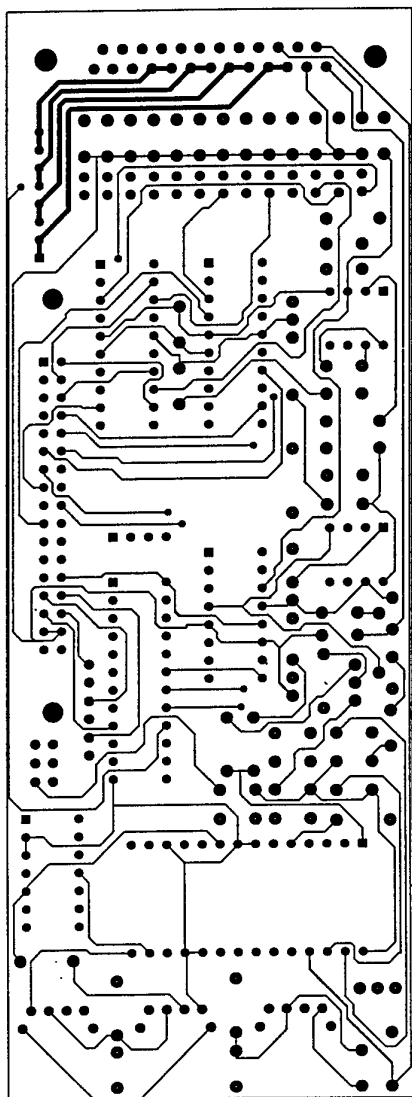
March 6, 1995

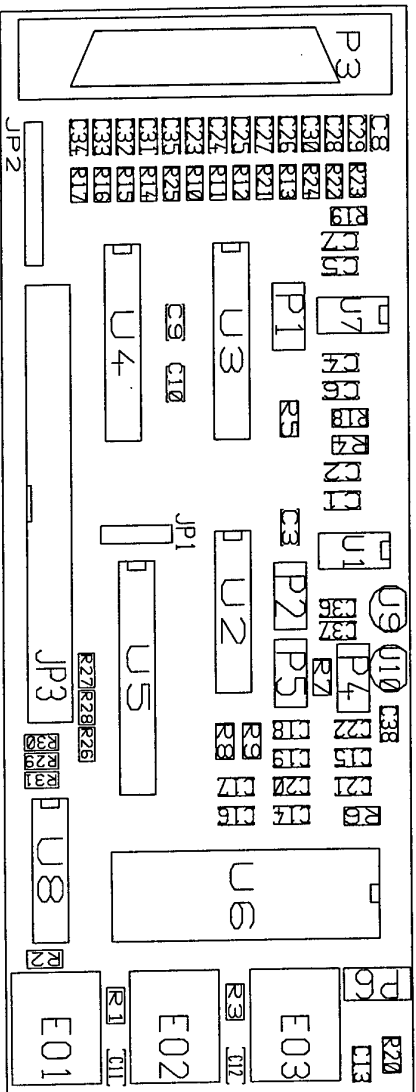
Item	Quantity	Reference	Part
1	12	C1,C3,C6,C7,C9,C10,C14, C15,C16,C18,C19,C20	10uF
2	3	C2,C4,C5	10nF
3	1	C8	2nF
4	5	C11,C12,C13,C21,C36	10 uF
5	1	C17	0.01uF
6	16	C22,C23,C24,C25,C26,C27, C28,C29,C30,C31,C32,C33, C34,C35,C37,C38	0.1uF
7	3	EO1,EO2,EO3	TOTX195
8	1	JP1	HEADER 4
9	1	JP2	HEADER 8
10	1	JP3	HEAD 34
11	5	P1,P2,R29,R30,R31	100K
12	1	P3	DB25
13	2	P4,P5	25K
14	4	P6,R26,R27,R28	1K
15	3	R1,R2,R3	6.2K
16	1	R4	100
17	1	R5	6.8K
18	1	R6	10
19	2	R7,R9	47K
20	1	R8	150K
21	13	R10,R11,R12,R13,R14,R15, R16,R17,R21,R22,R23,R24,R25	50
22	1	R18	3.3K
23	1	R19	330
24	1	R20	10k
25	2	U1,U7	OPA627
26	1	U2	DG445
27	2	U3,U4	MAX333A
28	1	U5	87C750
29	1	U6	DSP102
30	1	U8	74HCT93

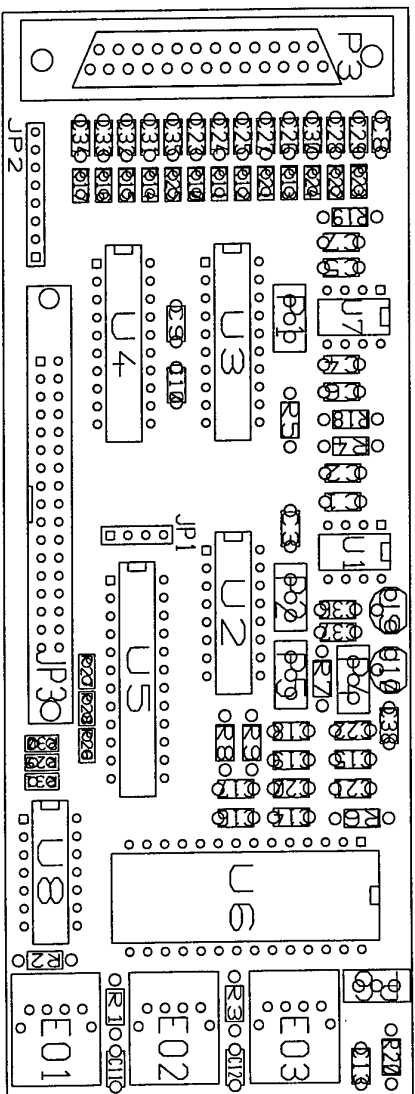
31	1	U9
32	1	U10

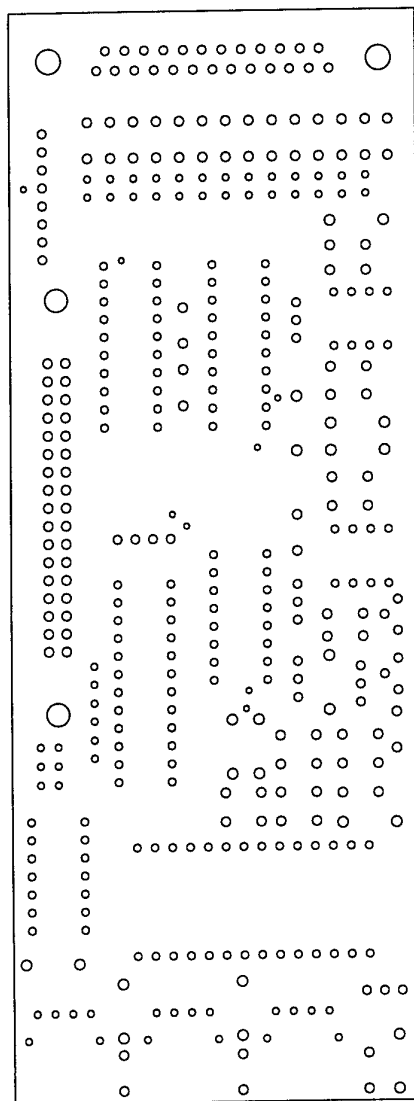
78L05
79L05











CCD HEAD

Revised: January 31, 1995

Revision: V1.0

Bill Of Materials

January 31, 1995

<u>Item</u>	<u>Quantity</u>	<u>Reference</u>	<u>Part</u>
1	1	Peltier1	Duratec DT1089
2	1	Peltier2	Duratec DT1069
3	1	P1	DB25
4	2	R1,R2	20k
5	1	U1	AD592
6	1	U2	CCD442A

DSP320C30 INTERFACE BOARD

Revised: March 4, 1995

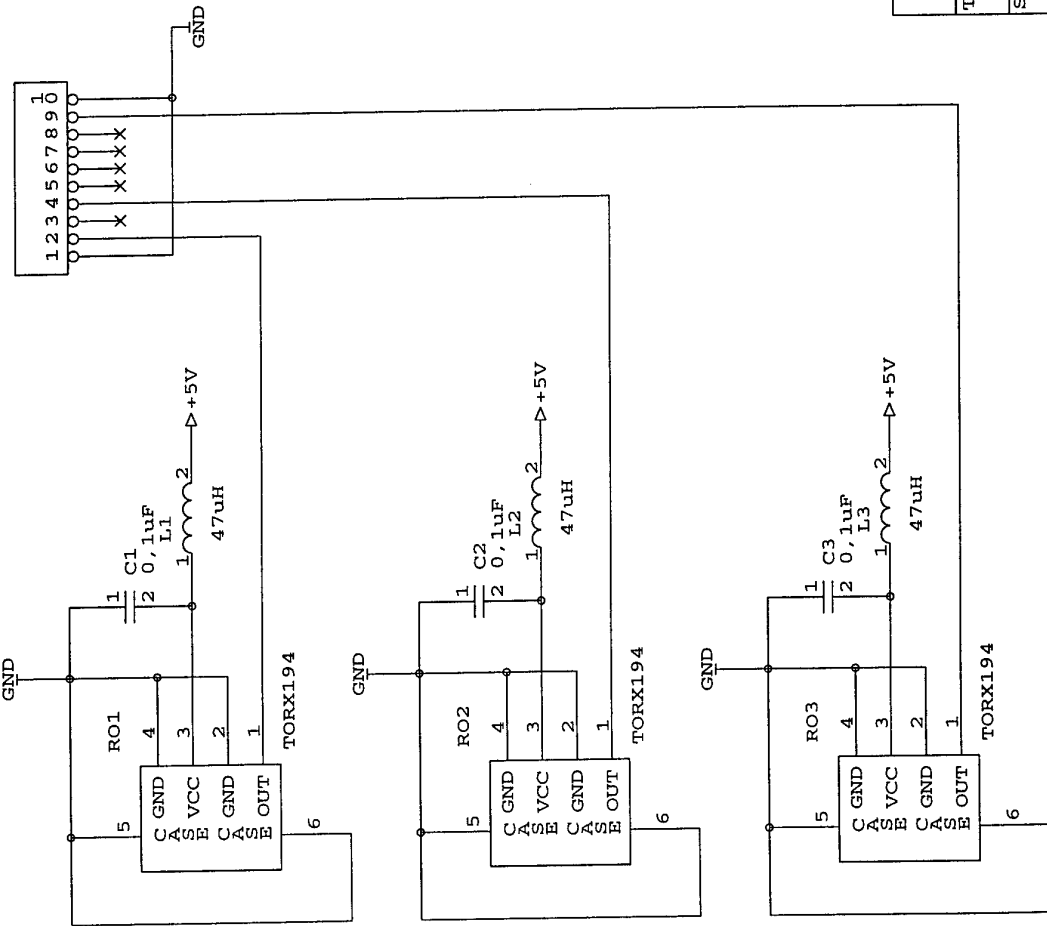
Revision: V1.0

March 6, 1995

Bill Of Materials

<u>Item</u>	<u>Quantity</u>	<u>Reference</u>	<u>Part</u>
1	3	C1,C2,C3	0,1uF
2	1	C4	10uF
3	1	JP1	HEAD 10
4	1	J1	IBMPC
5	3	L1,L2,L3	47uC
6	3	RO1,RO2,RO3	TORX194

JP1
HEAD 10



J1

X	A31	B31	X
X	A30	B30	X
X	A29	B29	X
X	A28	B28	X
X	A27	B27	X
X	A26	B26	X
X	A25	B25	X
X	A24	B24	X
X	A23	B23	X
X	A22	B22	X
X	A21	B21	X
X	A20	B20	X
X	A19	B19	X
X	A18	B18	X
X	A17	B17	X
X	A16	B16	X
X	A15	B15	X
X	A14	B14	X
X	A13	B13	X
X	A12	B12	X
X	A11	B11	X
X	A10	B10	X
X	A9	B9	X
X	A8	B8	X
X	A7	B7	X
X	A6	B6	X
X	A5	B5	X
X	A4	B4	X
X	A3	B3	X
X	A2	B2	X
X	A1	B1	X

IBMPC

OBSERVATOIRE DE LA COTE D'AZUR
06460 CAUSSOLS FRANCE

Title

DSP320C30 interface board

Size

Document Number

A4

REV

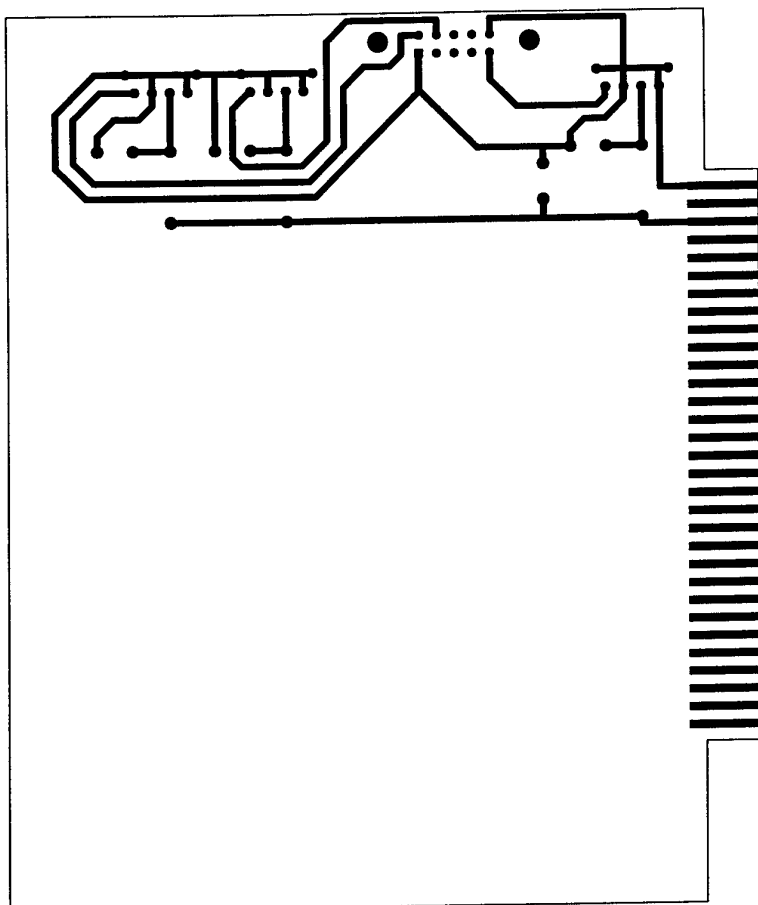
1.0

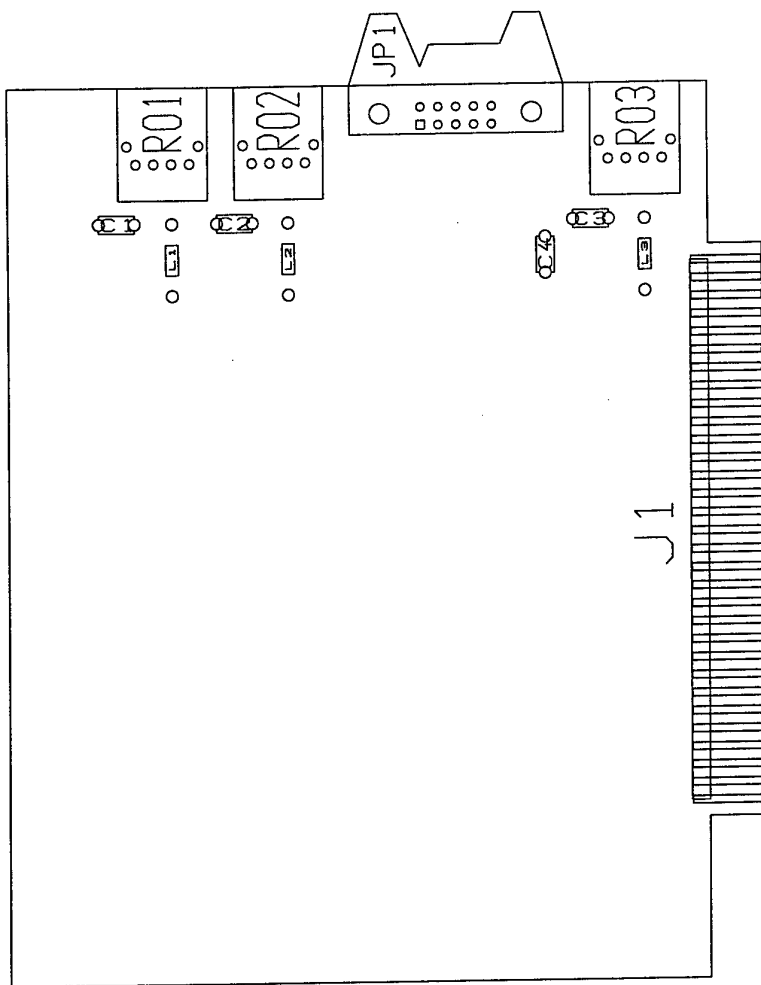
Date:

March 4, 1995

Sheet

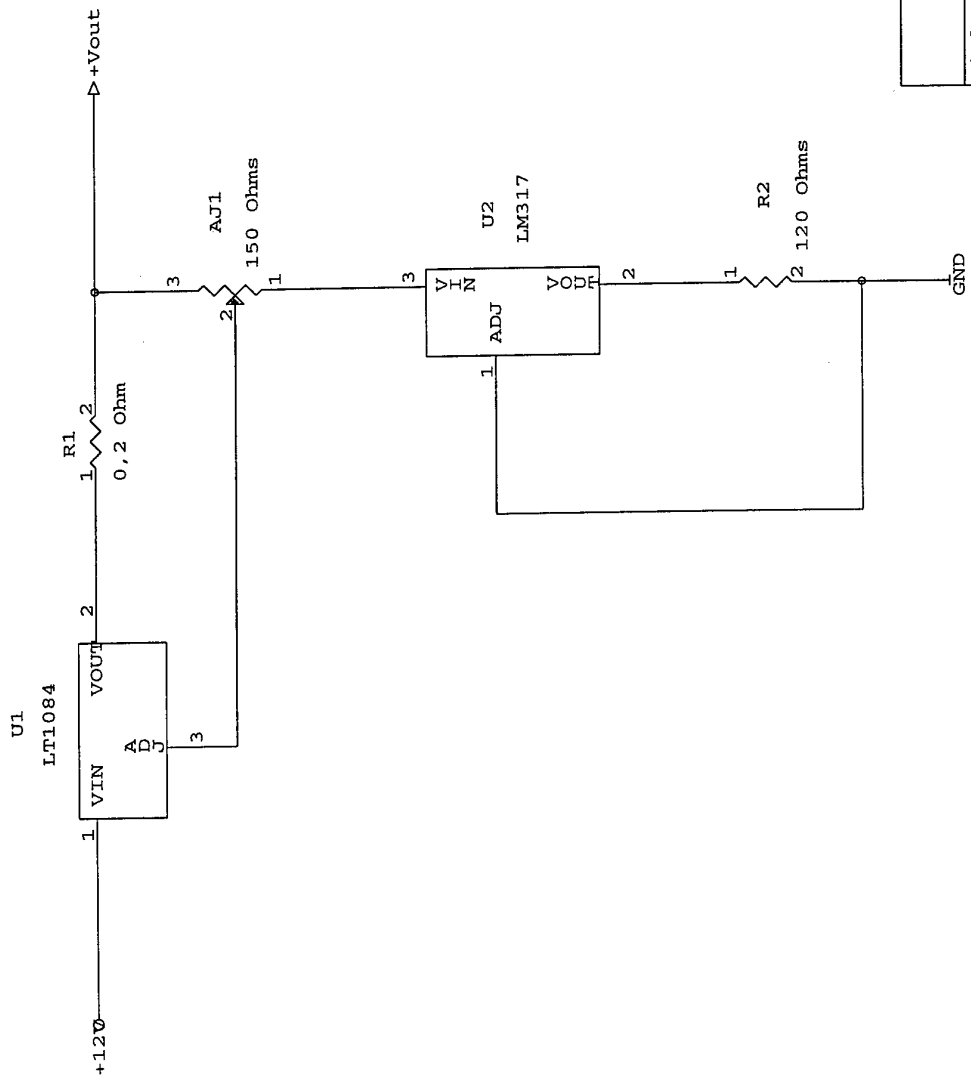
1 of 1





PELTIER MODULE POWER SUPPLY Revised: March 10, 1995
Revision: V1.0
Bill Of Materials March 10, 1995

<u>Item</u>	<u>Quantity</u>	<u>Reference</u>	<u>Part</u>
1	1	AJ1	150 Ohm
2	1	R1	0,2 Ohm
3	1	R2	120 Ohms
4	1	U1	LT1084
5	1	U2	LM317



OBSERVATOIRE DE LA COTE D'AZUR		
06460 CAUSSOLS FRANCE		
Title		
PELTIER POWER SUPPLY		
Size Document Number		
A4		REV
Date:	March 11, 1995	Sheet 1 of 1

CONTROL BOARD

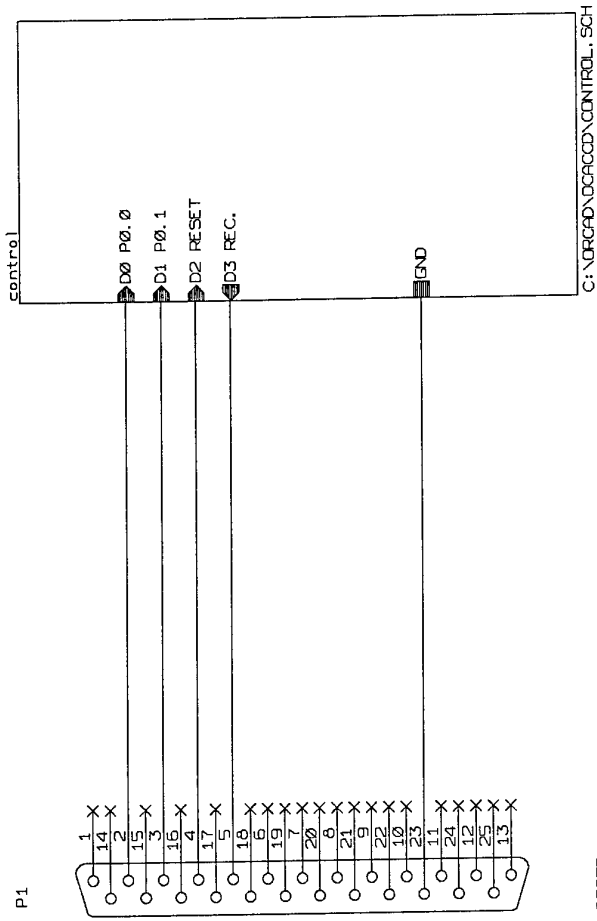
Revised: March 14, 1995

Revision: 1.0

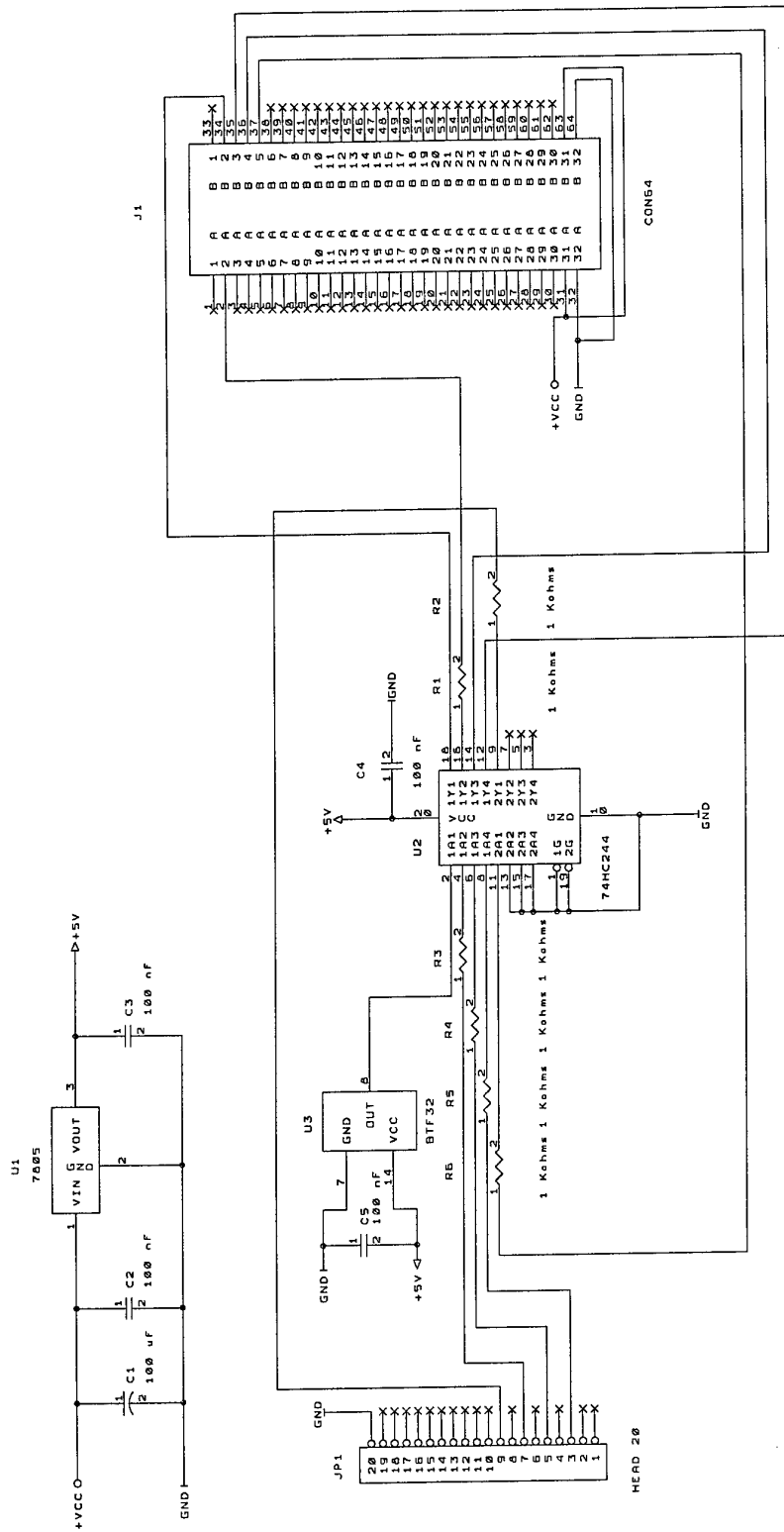
Bill Of Materials

March 14, 1995

<u>Item</u>	<u>Quantity</u>	<u>Reference</u>	<u>Part</u>
1	1	C1	100 uF
2	4	C2,C3,C4,C5	100 nF
3	1	JP1	HEAD 20
4	1	J1	CON64
5	6	R1,R2,R3,R4,R5,R6	1 Kohms
6	1	U1	7805
7	1	U2	74HC244
8	1	U3	BTF32



Size	Document Number	REV
A4		
Date:	March 14, 1995	Sheet 1 of 2



OBSERVATOIRE DE LA COTE D'AZUR	
06460 CAUSSOLS FRANCE	
Titre	
CAMERA CCD LORAL 442A	
Série Document Number	
REV	
Date	
March 14, 1995 Sheet 1 of 1	

93-4007

Appendix 2 :
Data sheets of the
main components of
the O.C.A. controller

This section contains the data sheets of the components used in this controller. It does omit passive parts, as well as other usual parts as 78xx voltage regulators and the like.

- CCD442A
- AD592
- 87C750
- MAX333A
- DG445
- OPA627
- DSP102
- 74HCT93
- TOTX195
- TORX194
- LM399
- OPA470

This information is only available of course in the printed form of this report.

FEATURES

Very Low Noise $5\text{nV}/\sqrt{\text{Hz}}$ @ 1kHz Max
Excellent Input Offset Voltage 0.4mV Max
Low Offset Voltage Drift $2\mu\text{V}/^\circ\text{C}$ Max
Very High Gain 1000V/mV Min
Outstanding CMR 110dB Min
Slew Rate $2\text{V}/\mu\text{s}$ Typ
Gain-Bandwidth Product 6MHz Typ
Industry Standard Quad Pinouts
Available in Die Form

ORDERING INFORMATION [†]

V _{OS} MAX (μV)	PACKAGE			OPERATING TEMPERATURE RANGE
	CERDIP 14-PIN	PLASTIC	LCC*	
400	—	—	OP470ARC/883	MIL
400	OP470AY*	—	OP470ATC/883	MIL
400	OP470EY	—	—	IND
800	OP470FY	—	—	IND
1000	—	OP470GP	—	XIND
1000	—	OP470GS**	—	XIND

For devices processed in total compliance to MIL-STD-883, add /883 after part number. Consult factory for 883 data sheet.

Burn-in is available on commercial and industrial temperature range parts in CerDIP, plastic DIP, and TO-can packages.

For availability and burn-in information on SO and PLOC packages, contact your local sales office.

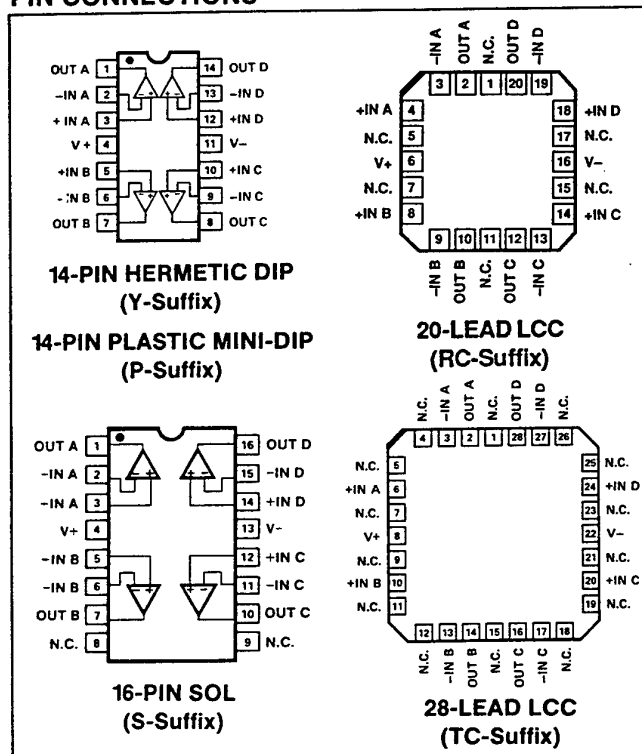
GENERAL DESCRIPTION

The OP-470 is a high-performance monolithic quad operational amplifier with exceptionally low voltage noise, $5\text{nV}/\sqrt{\text{Hz}}$ at 1kHz Max, offering comparable performance to the industry standard OP-27.

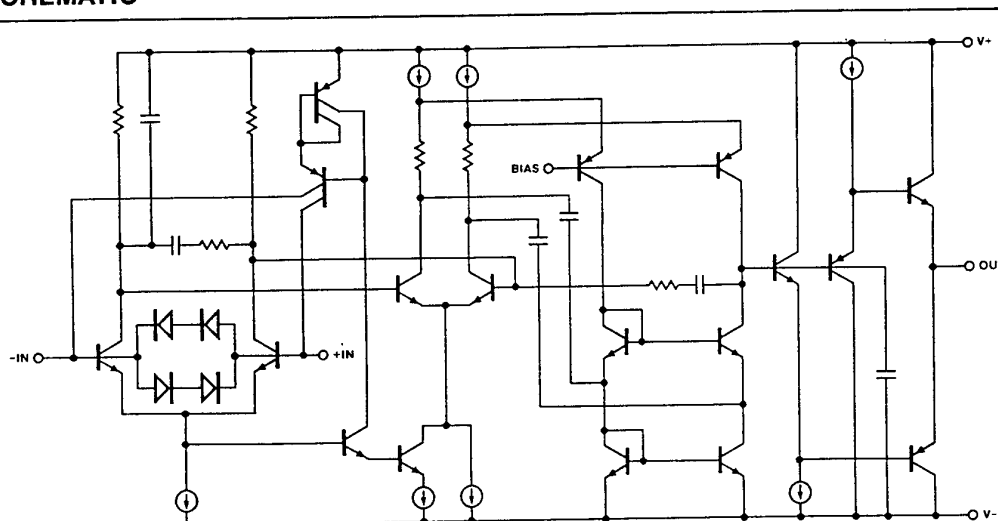
The OP-470 features an input offset voltage below 0.4mV, excellent for a quad op amp, and an offset drift under $2\mu\text{V}/^\circ\text{C}$, guaranteed over the full military temperature range. Open-loop gain of the OP-470 is over 1,000,000 into a 10k Ω load

insuring excellent gain accuracy and linearity, even in high-gain applications. Input bias current is under 25nA which reduces errors due to signal source resistance. The OP-470's CMR of over 110dB and PSRR of less than $1.8\mu\text{V}/\text{V}$ significantly reduce errors due to ground noise and power supply fluctuations. Power consumption of the quad OP-470 is half that of four OP-27s, a significant advantage for power con-

PIN CONNECTIONS



AMPLIFIED SCHEMATIC



OP-470

scious applications. The OP-470 is unity-gain stable with a gain-bandwidth product of 6MHz and a slew rate of 2V/ μ s.

The OP-470 offers excellent amplifier matching which is important for applications such as multiple gain blocks, low-noise instrumentation amplifiers, quad buffers, and low-noise active filters.

The OP-470 conforms to the industry standard 14-pin DIP pinout. It is pin compatible with the OP-11, LM148/149, HA4741, HA5104, and RM4156 quad op amps and can be used to upgrade systems using these devices.

For higher speed applications the OP-471, with a slew rate of 3V/ μ s, is recommended.

ABSOLUTE MAXIMUM RATINGS (Note 1)

Supply Voltage	$\pm 18V$
Differential Input Voltage (Note 2)	$\pm 1.0V$
Differential Input Current (Note 2)	$\pm 25mA$
Input Voltage	Supply Voltage
Output Short-Circuit Duration	Continuous
Storage Temperature Range	
P, TC, Y-Package	$-65^{\circ}C$ to $+150^{\circ}C$

Lead Temperature Range (Soldering, 60 sec)	$300^{\circ}C$
Junction Temperature (T_J)	$-65^{\circ}C$ to $+150^{\circ}C$
Operating Temperature Range	
OP-470A	$-55^{\circ}C$ to $+125^{\circ}C$
OP-470E, OP-470F	$-25^{\circ}C$ to $+85^{\circ}C$
OP-470G	$-40^{\circ}C$ to $+85^{\circ}C$

PACKAGE TYPE	θ_{JA} (Note 3)	θ_{JC}	UNITS
14-Pin Hermetic DIP (Y)	94	10	$^{\circ}C/W$
14-Pin Plastic DIP (P)	76	33	$^{\circ}C/W$
20-Contact LCC (RC)	78	30	$^{\circ}C/W$
28-Contact LCC (TC)	70	28	$^{\circ}C/W$
16-Pin SOL (S)	88	23	$^{\circ}C/W$

NOTES:

1. Absolute maximum ratings apply to both DICE and packaged parts, unless otherwise noted.
2. The OP-470's inputs are protected by back-to-back diodes. Current limiting resistors are not used in order to achieve low noise performance. If differential voltage exceeds $\pm 1.0V$, the input current should be limited to $\pm 25mA$.
3. θ_{JA} is specified for worst case mounting conditions, i.e., θ_{JA} is specified for device in socket for TO, CerDIP, P-DIP, and LCC packages; θ_{JA} is specified for device soldered to printed circuit board for SO and PLCC packages.

ELECTRICAL CHARACTERISTICS at $V_S = \pm 15V$, $T_A = 25^{\circ}C$, unless otherwise noted.

PARAMETER	SYMBOL	CONDITIONS	OP-470A/E			OP-470F			OP-470G			UNITS
			MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX	
Input Offset Voltage	V_{OS}		—	0.1	0.4	—	0.2	0.8	—	0.4	1.0	mV
Input Offset Current	I_{OS}	$V_{CM} = 0V$	—	3	10	—	6	20	—	12	30	nA
Input Bias Current	I_B	$V_{CM} = 0V$	—	6	25	—	15	50	—	25	60	nA
Input Noise Voltage	e_{np-p}	0.1Hz to 10Hz (Note 1)	—	80	200	—	80	200	—	80	200	nV _{p-p}
Input Noise Voltage Density	e_n	$f_O = 10Hz$	—	3.8	6.5	—	3.8	6.5	—	3.8	6.5	nV/ $\sqrt{Hz}$
		$f_O = 100Hz$	—	3.3	5.5	—	3.3	5.5	—	3.3	5.5	
		$f_O = 1kHz$	—	3.2	5.0	—	3.2	5.0	—	3.2	5.0	
		(Note 2)										
Input Noise Current Density	i_n	$f_O = 10Hz$	—	1.7	—	—	1.7	—	—	1.7	—	pA/ $\sqrt{Hz}$
		$f_O = 100Hz$	—	0.7	—	—	0.7	—	—	0.7	—	
		$f_O = 1kHz$	—	0.4	—	—	0.4	—	—	0.4	—	
Large-Signal Voltage Gain	A_{VO}	$V_O = \pm 10V$										V/mV
		$R_L = 10k\Omega$ $R_L = 2k\Omega$	1000 500	2300 1200	—	800 400	1700 900	—	800 400	1700 900	—	
Input Voltage Range	IVR	(Note 3)	± 11	± 12	—	± 11	± 12	—	± 11	± 12	—	V
Output Voltage Swing	V_O	$R_L \geq 2k\Omega$	± 12	± 13	—	± 12	± 13	—	± 12	± 13	—	V
Common-Mode Rejection CMR		$V_{CM} = \pm 11V$	110	125	—	100	120	—	100	120	—	dB
Power Supply Rejection Ratio	PSRR	$V_S = \pm 4.5V$ to $\pm 18V$	—	0.56	1.8	—	1.0	5.6	—	1.0	5.6	$\mu V/V$
Slew Rate	SR		1.4	2	—	1.4	2	—	1.4	2	—	V/ μs

ELECTRICAL CHARACTERISTICS at $V_S = \pm 15V$, $T_A = 25^\circ C$, unless otherwise noted. (Continued)

PARAMETER	SYMBOL	CONDITIONS	OP-470A/E			OP-470F			OP-470G			UNITS
			MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX	
Supply Current (Amplifiers)	I_{SY}	No Load	—	9	11	—	9	11	—	9	11	mA
Gain Bandwidth Product	GBW	$A_V = +10$	—	6	—	—	6	—	—	6	—	MHz
Common-Mode Rejection Separation	CS	$V_O = 20V_{p-p}$ $f_O = 10Hz$ (Note 1)	125	155	—	125	155	—	125	155	—	dB
Input Capacitance	C_{IN}		—	2	—	—	2	—	—	2	—	pF
Input Resistance Differential-Mode	R_{IN}		—	0.4	—	—	0.4	—	—	0.4	—	M Ω
Input Resistance Common-Mode	R_{INCM}		—	11	—	—	11	—	—	11	—	G Ω
Slew Rate	t_s	$A_V = +1$	—	5.5	—	—	5.5	—	—	5.5	—	μs
		to 0.1% to 0.01%	—	6.0	—	—	6.0	—	—	6.0	—	

2

NOTES:

1. Guaranteed but not 100% tested.

2. Sample tested.

3. Guaranteed by CMR test.

ELECTRICAL CHARACTERISTICS at $V_S = \pm 15V$, $-55^\circ C \leq T_A \leq 125^\circ C$ for OP-470A, unless otherwise noted.

PARAMETER	SYMBOL	CONDITIONS	OP-470A			UNITS
			MIN	TYP	MAX	
Input Offset Voltage	V_{OS}		—	0.14	0.6	mV
Average Input Offset Voltage Drift	TCV_{OS}		—	0.4	2	$\mu V/^\circ C$
Input Offset Current	I_{OS}	$V_{CM} = 0V$	—	5	20	nA
Input Bias Current	I_B	$V_{CM} = 0V$	—	15	50	nA
Open-Loop Voltage Gain	A_{VO}	$V_O = \pm 10V$	750	1600	—	V/mV
		$R_L = 10k\Omega$ $R_L = 2k\Omega$	400	800	—	
Output Voltage Range	IVR	(Note 1)	± 11	± 12	—	V
Output Voltage Swing	V_O	$R_L \geq 2k\Omega$	± 12	± 13	—	V
Common-Mode Rejection	CMR	$V_{CM} = \pm 11V$	100	120	—	dB
Power Supply Rejection Ratio	PSRR	$V_S = \pm 4.5V$ to $\pm 18V$	—	1.0	5.6	$\mu V/V$
Supply Current (Amplifiers)	I_{SY}	No Load	—	9.2	11	mA

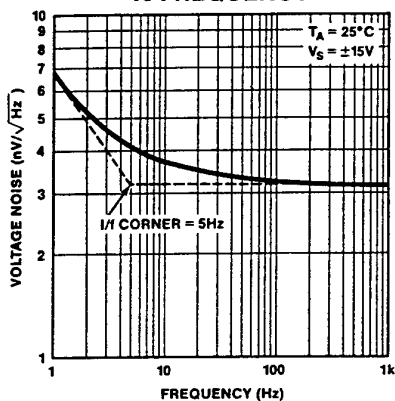
NOTE:

1. Guaranteed by CMR test.

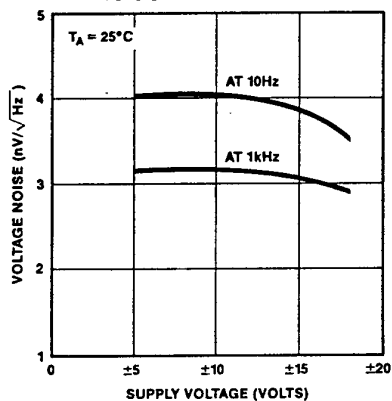
OP-470

TYPICAL PERFORMANCE CHARACTERISTICS

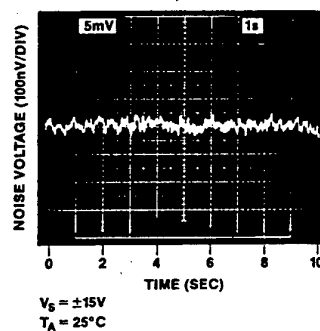
VOLTAGE NOISE DENSITY vs FREQUENCY



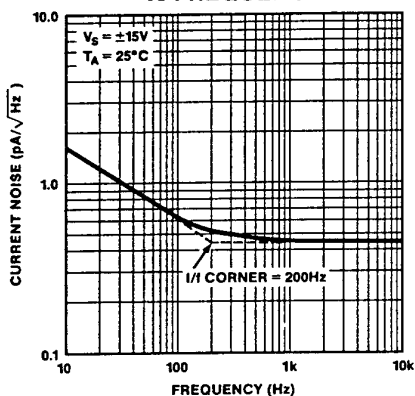
VOLTAGE NOISE DENSITY vs SUPPLY VOLTAGE



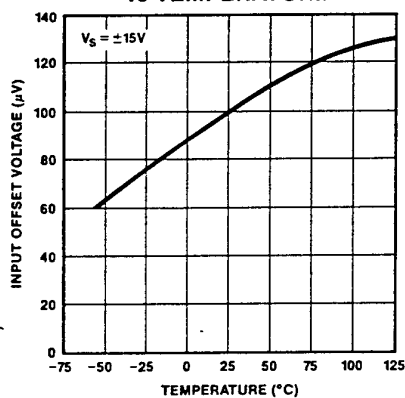
0.1Hz TO 10Hz NOISE



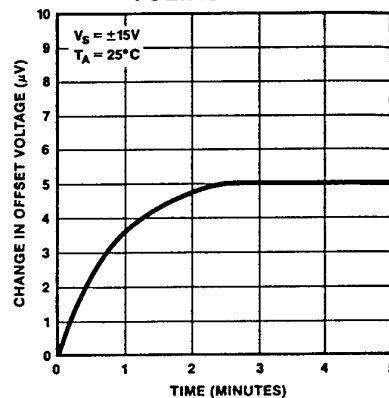
CURRENT NOISE DENSITY vs FREQUENCY



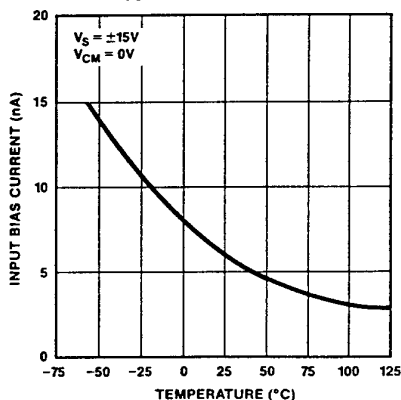
INPUT OFFSET VOLTAGE vs TEMPERATURE



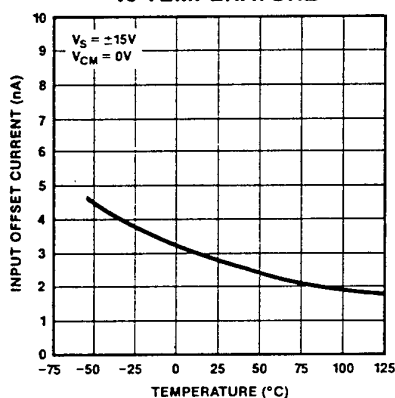
WARM-UP OFFSET VOLTAGE DRIFT



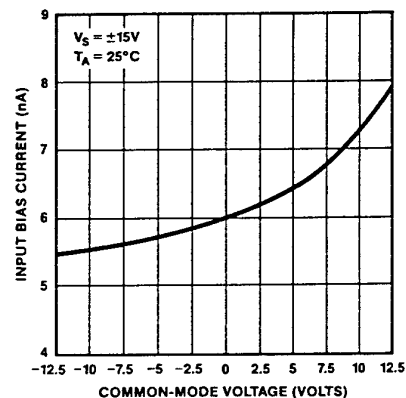
INPUT BIAS CURRENT vs TEMPERATURE



INPUT OFFSET CURRENT vs TEMPERATURE



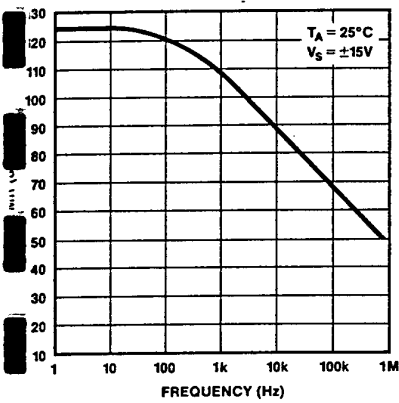
INPUT BIAS CURRENT vs COMMON-MODE VOLTAGE



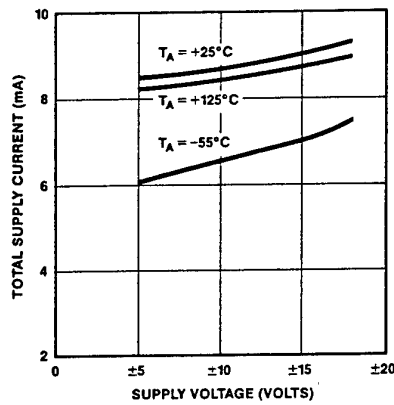
TYPICAL PERFORMANCE CHARACTERISTICS

2

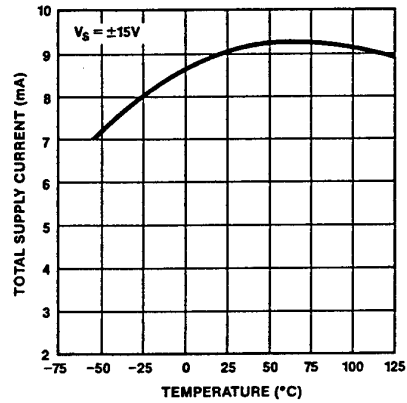
CMR vs FREQUENCY



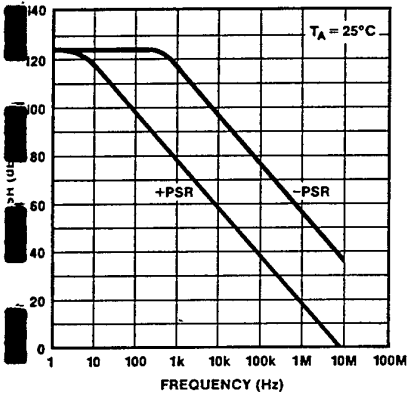
TOTAL SUPPLY CURRENT vs SUPPLY VOLTAGE



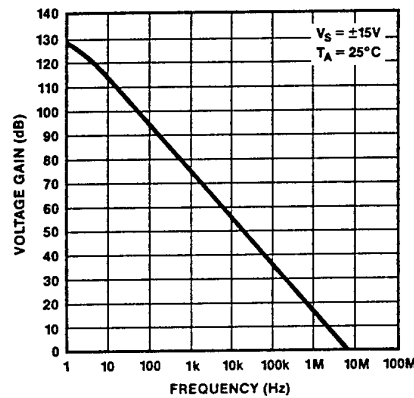
TOTAL SUPPLY CURRENT vs TEMPERATURE



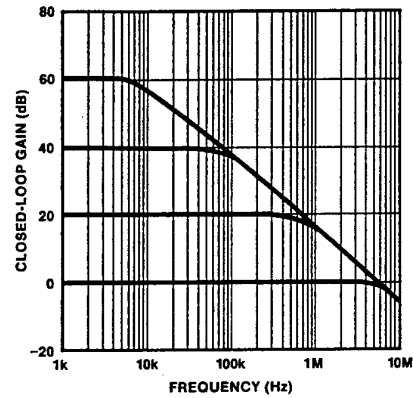
PSR vs FREQUENCY



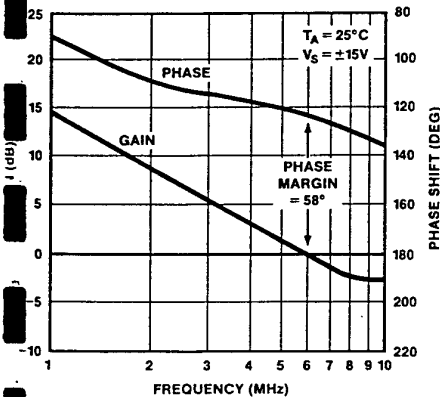
OPEN-LOOP GAIN vs FREQUENCY



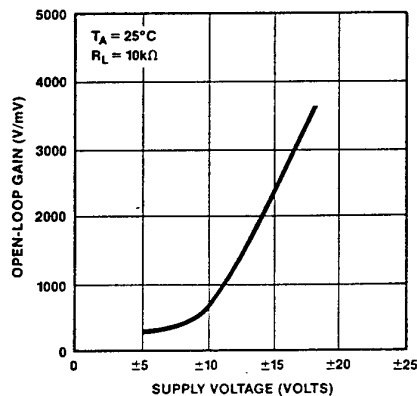
CLOSED-LOOP GAIN vs FREQUENCY



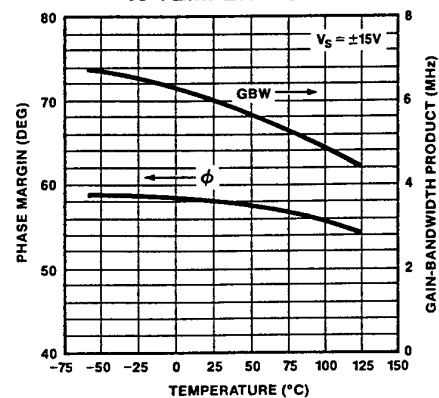
OPEN-LOOP GAIN, PHASE SHIFT vs FREQUENCY



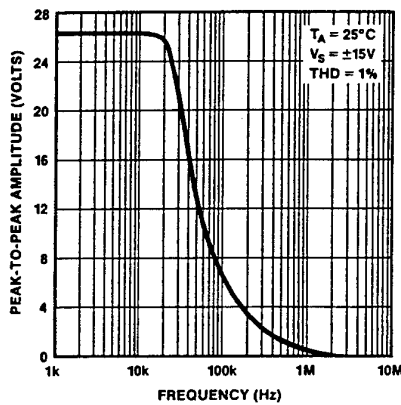
OPEN-LOOP GAIN vs SUPPLY VOLTAGE



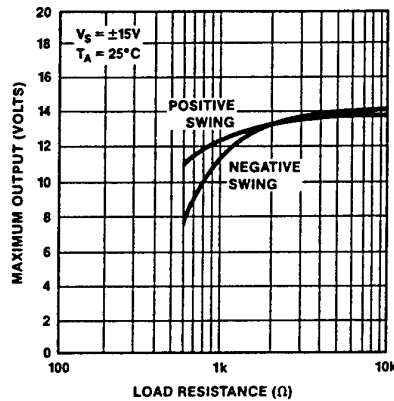
GAIN-BANDWIDTH PRODUCT, PHASE MARGIN vs TEMPERATURE



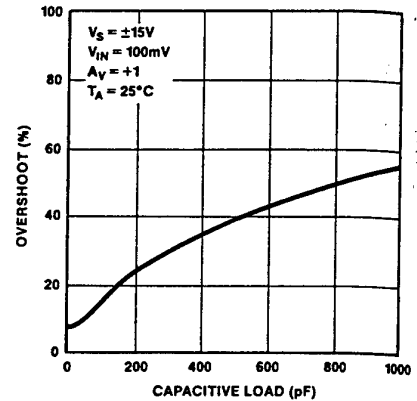
MAXIMUM OUTPUT SWING vs FREQUENCY



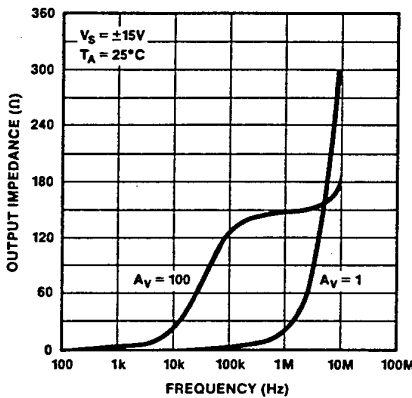
MAXIMUM OUTPUT VOLTAGE vs LOAD RESISTANCE



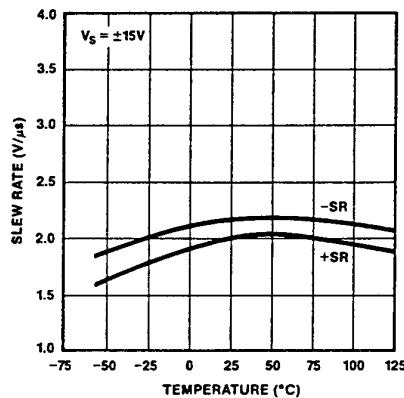
SMALL-SIGNAL OVERSHOOT vs CAPACITIVE LOAD



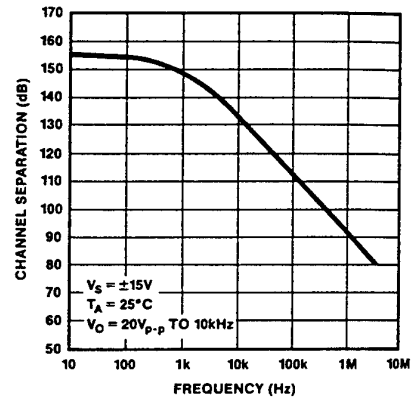
OUTPUT IMPEDANCE vs FREQUENCY



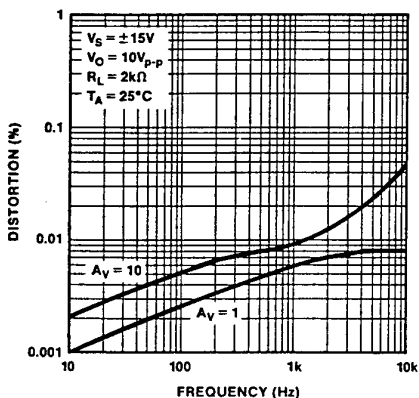
SLEW RATE vs TEMPERATURE



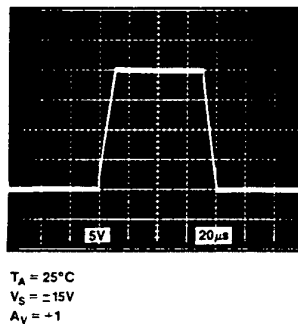
CHANNEL SEPARATION vs FREQUENCY



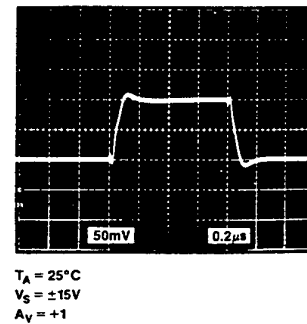
TOTAL HARMONIC DISTORTION vs FREQUENCY



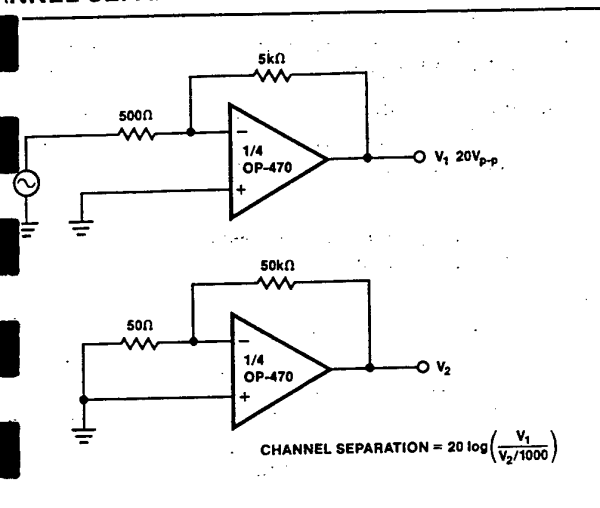
LARGE-SIGNAL TRANSIENT RESPONSE



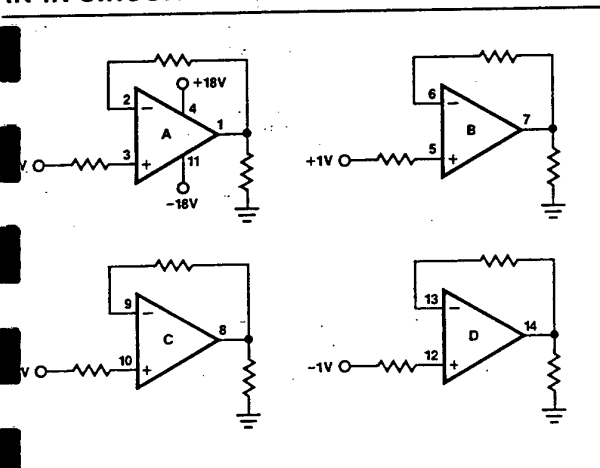
SMALL-SIGNAL TRANSIENT RESPONSE



CHANNEL SEPARATION TEST CIRCUIT



IN-IN CIRCUIT



APPLICATIONS INFORMATION

VOLTAGE AND CURRENT NOISE

The OP-470 is a very low-noise quad op amp, exhibiting a total voltage noise of only $3.2 \text{ nV}/\sqrt{\text{Hz}}$ @ 1kHz. The exceptionally low noise characteristics of the OP-470 is in part achieved by operating the input transistors at high collector currents since the voltage noise is inversely proportional to the square root of the collector current. Current noise, however, is directly proportional to the square root of collector current. As a result, the outstanding voltage noise performance of the OP-470 is gained at the expense of current noise performance, which is typical for low noise amplifiers.

To obtain the best noise performance in a circuit it is vital to understand the relationship between voltage noise (e_n), current noise (i_n), and resistor noise (e_t).

TOTAL NOISE AND SOURCE RESISTANCE

The total noise of an op amp can be calculated by:

$$E_n = \sqrt{(e_n)^2 + (i_n R_S)^2 + (e_t)^2}$$

where:

E_n = total input referred noise

e_n = op amp voltage noise

i_n = op amp current noise

e_t = source resistance thermal noise

R_S = source resistance

The total noise is referred to the input and at the output would be amplified by the circuit gain.

Figure 1 shows the relationship between total noise at 1kHz and source resistance. For $R_S < 1 \text{ k}\Omega$ the total noise is dominated by the voltage noise of the OP-470. As R_S rises above

FIGURE 1: Total Noise vs Source Resistance (Including Resistor Noise) at 1kHz

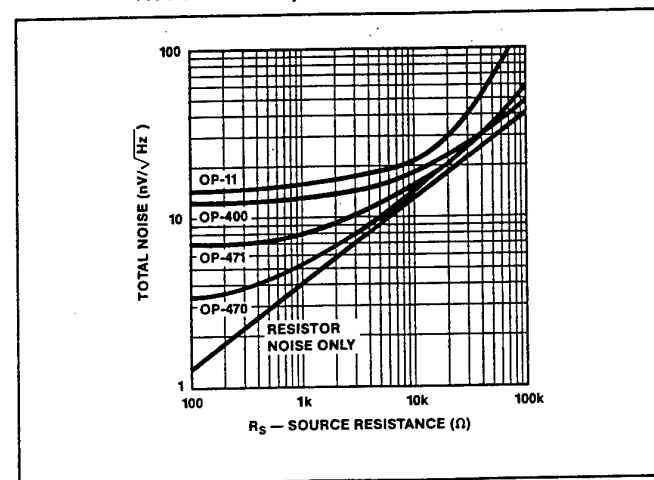
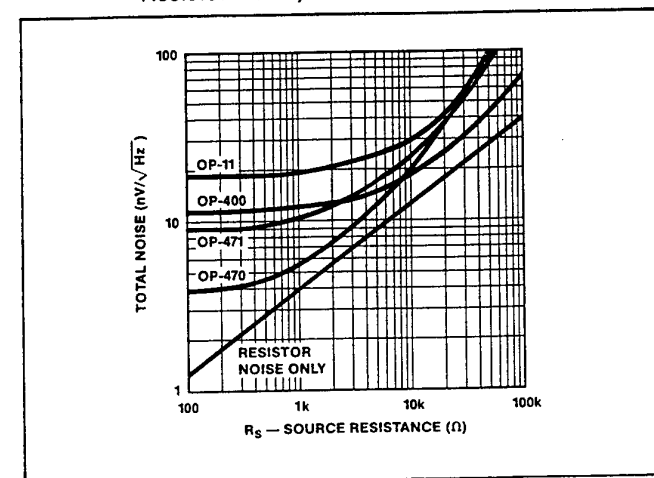


FIGURE 2: Total Noise vs Source Resistance (Including Resistor Noise) at 10Hz



OP-470

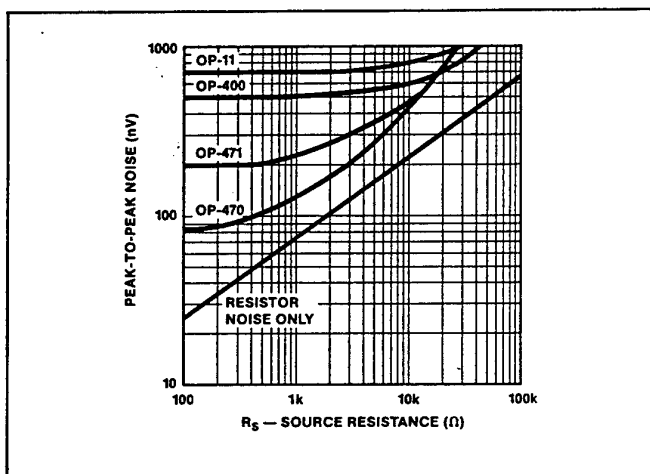
1k Ω , total noise increases and is dominated by resistor noise rather than by voltage or current noise of the OP-470. When R_S exceeds 20k Ω , current noise of the OP-470 becomes the major contributor to total noise.

Figure 2 also shows the relationship between total noise and source resistance, but at 10Hz. Total noise increases more quickly than shown in Figure 1 because current noise is inversely proportional to the square root of frequency. In Figure 2, current noise of the OP-470 dominates the total noise when $R_S > 5k\Omega$.

From Figures 1 and 2 it can be seen that to reduce total noise, source resistance must be kept to a minimum. In applications with a high source resistance, the OP-400, with lower current noise than the OP-470, will provide lower total noise.

Figure 3 shows peak-to-peak noise versus source resistance over the 0.1Hz to 10Hz range. Once again, at low values of R_S ,

FIGURE 3: Peak-To-Peak Noise (0.1Hz To 10Hz) vs Source Resistance (Includes Resistor Noise)



the voltage noise of the OP-470 is the major contributor to peak-to-peak noise with current noise the major contributor as R_S increases. The crossover point between the OP-470 and the OP-400 for peak-to-peak noise is at $R_S = 17k\Omega$.

The OP-471 is a higher speed version of the OP-470, with a slew rate of 8V/ μ s. Noise of the OP-471 is only slightly higher than the OP-470. Like the OP-470, the OP-471 is unity-gain stable.

For reference, typical source resistances of some signal sources are listed in Table I.

TABLE I

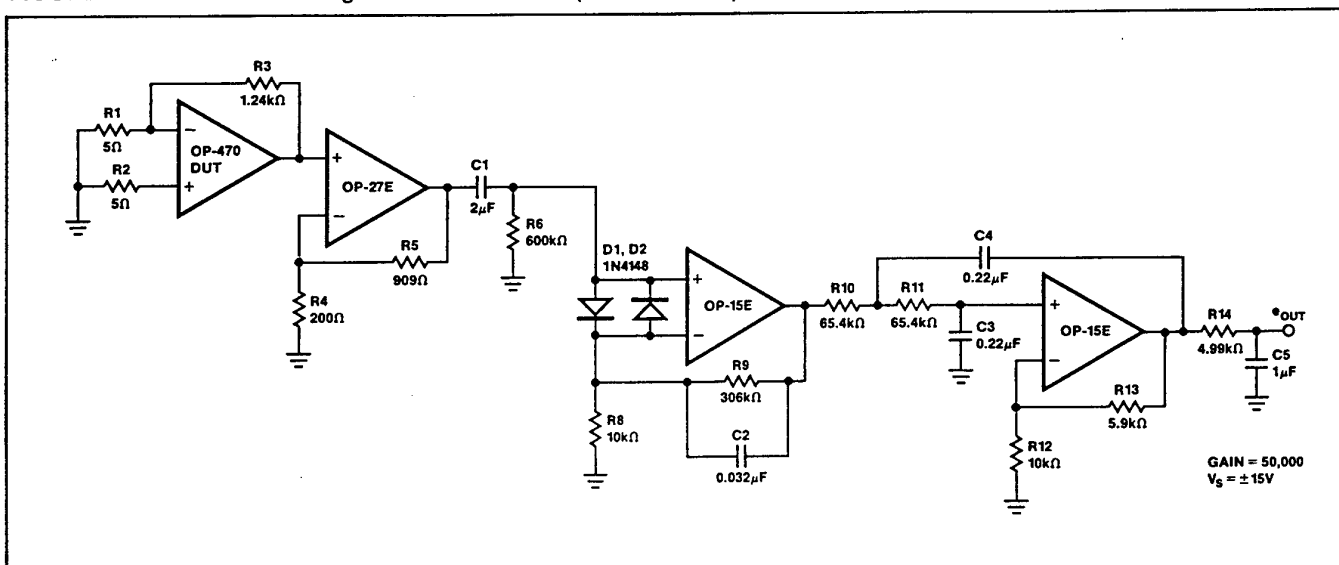
DEVICE	SOURCE IMPEDANCE	COMMENTS
Strain gauge	<500 Ω	Typically used in low-frequency applications.
Magnetic tapehead	<1500 Ω	Low I_B very important to reduce self-magnetization problems when direct coupling is used. OP-470 I_B can be neglected.
Magnetic phonograph cartridges	<1500 Ω	Similar need for low I_B in direct coupled applications. OP-470 will not introduce any self-magnetization problem.
Linear variable differential transformer	<1500 Ω	Used in rugged servo-feedback applications. Bandwidth of interest is 400Hz to 5kHz.

For further information regarding noise calculations, see "Minimization of Noise in Op-Amp Applications," Application Note AN-15.

NOISE MEASUREMENTS — PEAK-TO-PEAK VOLTAGE NOISE

The circuit of Figure 4 is a test setup for measuring peak-to-peak voltage noise. To measure the 200nV peak-to-peak

FIGURE 4: Peak-To-Peak Voltage Noise Test Circuit (0.1Hz To 10Hz)



ise specification of the OP-470 in the 0.1Hz to 10Hz range, following precautions must be observed:

The device has to be warmed-up for at least five minutes. As shown in the warm-up drift curve, the offset voltage typically changes $5\mu\text{V}$ due to increasing chip temperature after power-up. In the 10-second measurement interval, these temperature-induced effects can exceed tens-of-nanovolts.

For similar reasons, the device has to be well-shielded from air currents. Shielding also minimizes thermocouple effects.

Sudden motion in the vicinity of the device can also "feed-through" to increase the observed noise.

FIGURE 5: 0.1Hz To 10Hz Peak-To-Peak Voltage Noise Test Circuit Frequency Response

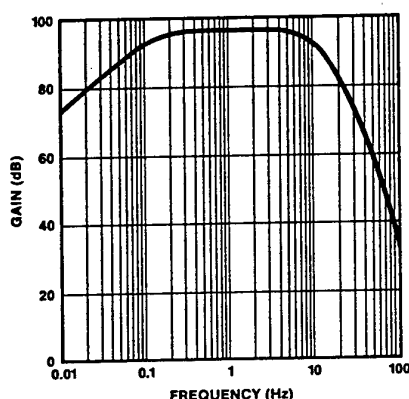
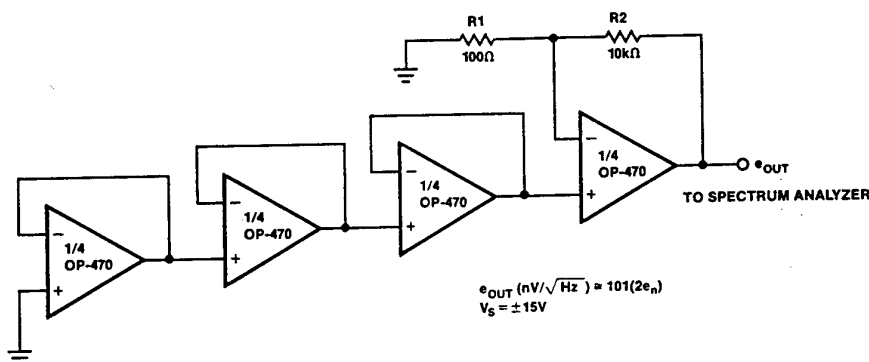


FIGURE 6: Noise Voltage Density Test Circuit



4. The test time to measure 0.1Hz-to-10Hz noise should not exceed 10 seconds. As shown in the noise-tester frequency-response curve of Figure 5, the 0.1Hz corner is defined by only one pole. The test time of 10 seconds acts as an additional pole to eliminate noise contribution from the frequency band below 0.1Hz.
5. A noise-voltage-density test is recommended when measuring noise on a large number of units. A 10Hz noise-voltage-density measurement will correlate well with a 0.1Hz-to-10Hz peak-to-peak noise reading, since both results are determined by the white noise and the location of the $1/f$ corner frequency.
6. Power should be supplied to the test circuit by well bypassed low-noise supplies, e.g. batteries. These will minimize output noise introduced via the amplifier supply pins.

NOISE MEASUREMENT — NOISE VOLTAGE DENSITY

The circuit of Figure 6 shows a quick and reliable method of measuring the noise voltage density of quad op amps. Each individual amplifier is series-connected and is in unity-gain, save the final amplifier which is in a noninverting gain of 101. Since the ac noise voltages of each amplifier are uncorrelated, they add in rms fashion to yield:

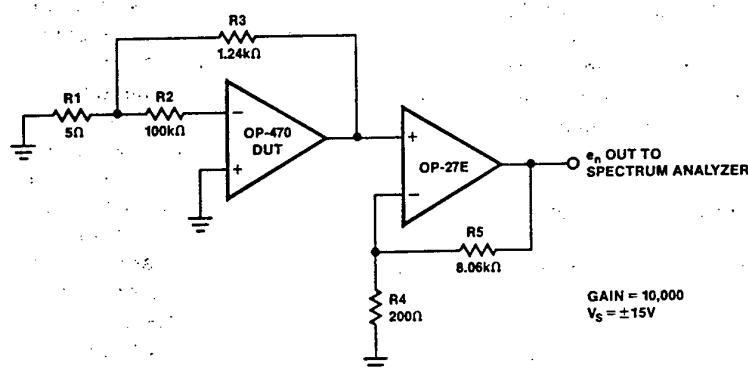
$$e_{OUT} = 101 \left(\sqrt{e_{nA}^2 + e_{nB}^2 + e_{nC}^2 + e_{nD}^2} \right)$$

The OP-470 is a monolithic device with four identical amplifiers. The noise voltage density of each individual amplifier will match, giving:

$$e_{OUT} = 101 \left(\sqrt{4e_n^2} \right) = 101 (2e_n)$$

OP-470

FIGURE 7: Current Noise Density Test Circuit



NOISE MEASUREMENT — CURRENT NOISE DENSITY

The test circuit shown in Figure 7 can be used to measure current noise density. The formula relating the voltage output to current noise density is:

$$i_n = \frac{\sqrt{\left(\frac{e_{nOUT}}{G}\right)^2 - \left(40\text{nV}/\sqrt{\text{Hz}}\right)^2}}{R_S}$$

where:

G = gain of 10000

R_S = 100kΩ source resistance

CAPACITIVE LOAD DRIVING AND POWER SUPPLY CONSIDERATIONS

The OP-470 is unity-gain stable and is capable of driving large capacitive loads without oscillating. Nonetheless, good supply bypassing is highly recommended. Proper supply bypassing reduces problems caused by supply line noise and improves the capacitive load driving capability of the OP-470.

In the standard feedback amplifier, the op amp's output resistance combines with the load capacitance to form a low-pass filter that adds phase shift in the feedback network and reduces stability. A simple circuit to eliminate this effect is shown in Figure 8. The added components, C1 and R3, decouple the amplifier from the load capacitance and provide additional stability. The values of C1 and R3 shown in Figure 8 are for a load capacitance of up to 1000pF when used with the OP-470.

In applications where the OP-470's inverting or noninverting inputs are driven by a low source impedance (under 100Ω) or connected to ground, if V₊ is applied before V₋, or when V₋ is disconnected, excessive parasitic currents will flow. Most

FIGURE 8: Driving Large Capacitive Loads

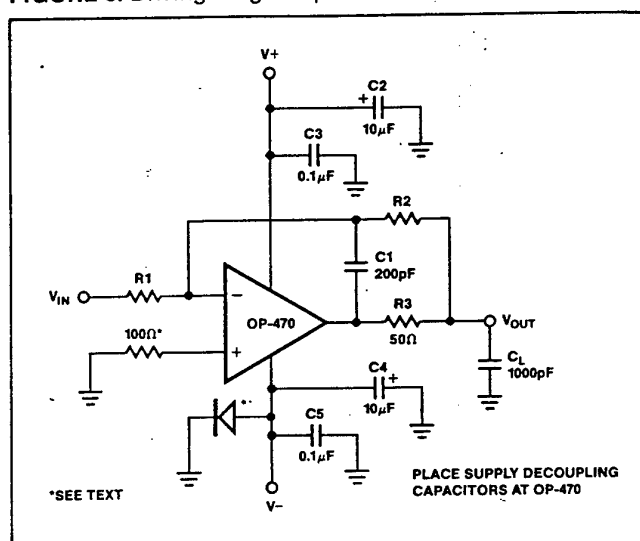
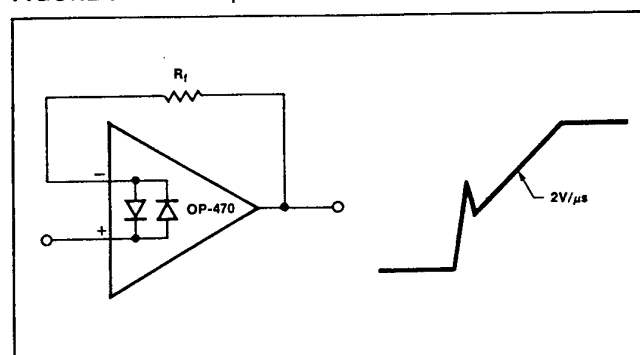


FIGURE 9: Pulsed Operation



Applications use dual tracking supplies and with the device supply pins properly bypassed, power-up will not present a problem. A source resistance of at least 100Ω in series with all outputs (Figure 8) will limit the parasitic currents to a safe level. V_{-} is disconnected. It should be noted that any source resistance, even 100Ω , adds noise to the circuit. Where noise is required to be kept at a minimum, a germanium or Schottky diode can be used to clamp the V_{-} pin and eliminate the parasitic current flow instead of using series limiting resistors. In most applications, only one diode clamp is required per board or system.

UNITY-GAIN BUFFER APPLICATIONS

When $R_f \leq 100\Omega$ and the input is driven with a fast, large-amplitude pulse ($>1V$), the output waveform will look as shown in Figure 9.

During the fast feedthrough-like portion of the output, the output protection diodes effectively short the output to the input, and a current, limited only by the output short-circuit current, will be drawn by the signal generator. With $R_f \geq 100\Omega$, the output is capable of handling the current requirements ($I_L \leq 20mA$ at $10V$); the amplifier will stay in its active mode and a smooth transition will occur.

When $R_f > 3k\Omega$, a pole created by R_f and the amplifier's input capacitance ($2pF$) creates additional phase shift and reduces the phase margin. A small capacitor (20 to $50pF$) in parallel with R_f helps eliminate this problem.

APPLICATIONS

LOW NOISE AMPLIFIER

A simple method of reducing amplifier noise by paralleling amplifiers is shown in Figure 10. Amplifier noise, depicted in Figure 11, is around $2nV/\sqrt{Hz}$ @ $1kHz$ (R.T.I.). Gain for each paralleled amplifier and the entire circuit is 1000. The 200Ω resistors limit circulating currents and provide an effective input resistance of 50Ω . The amplifier is stable with a $10nF$ capacitive load and can supply up to $30mA$ of output drive.

DIGITAL PANNING CONTROL

Figure 12 uses a DAC-8408, a quad 8-bit DAC, to pan a signal between two channels. The complementary DAC current outputs of two of the DAC-8408's four DACs drive current-to-voltage converters built from a single quad OP-470. The amplifiers have complementary outputs with the amplitudes dependent upon the digital code applied to the DAC. Figure 13 shows the complementary outputs for a $1kHz$ input signal and a digital ramp applied to the DAC data inputs. Distortion of the digital panning control is less than 0.01% .

An error due to the mismatching between the internal DAC feedback resistors and the current-to-voltage feedback resis-

tors is eliminated by using feedback resistors internal to the DAC. Of the four DACs available in the DAC-8408, only two, DACs A and C, actually pass a signal. DACs B and D are used to provide the additional feedback resistors needed in the circuit. If the V_{REFB} and V_{REFD} inputs remain unconnected the current-to-voltage converters using R_{FBB} and R_{FBD} are unaffected by digital data reaching DACs B and D.

FIGURE 10: Low Noise Amplifier

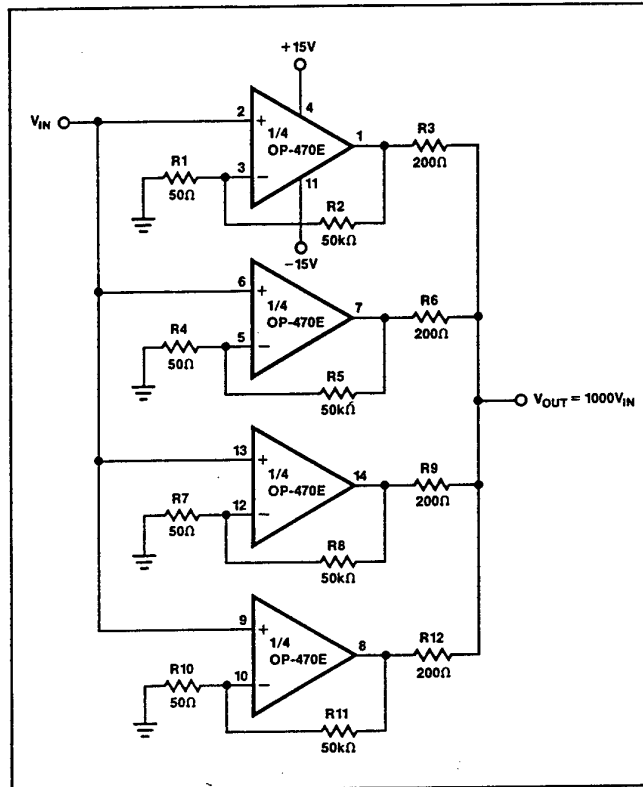
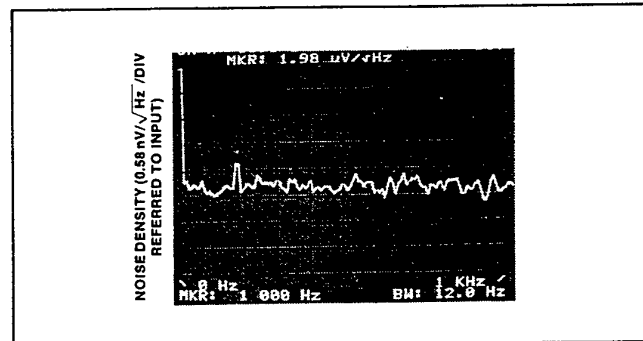


FIGURE 11: Noise Density of Low Noise Amplifier, $G = 1000$



OP-470

FIGURE 12: Digital Panning Control Circuit

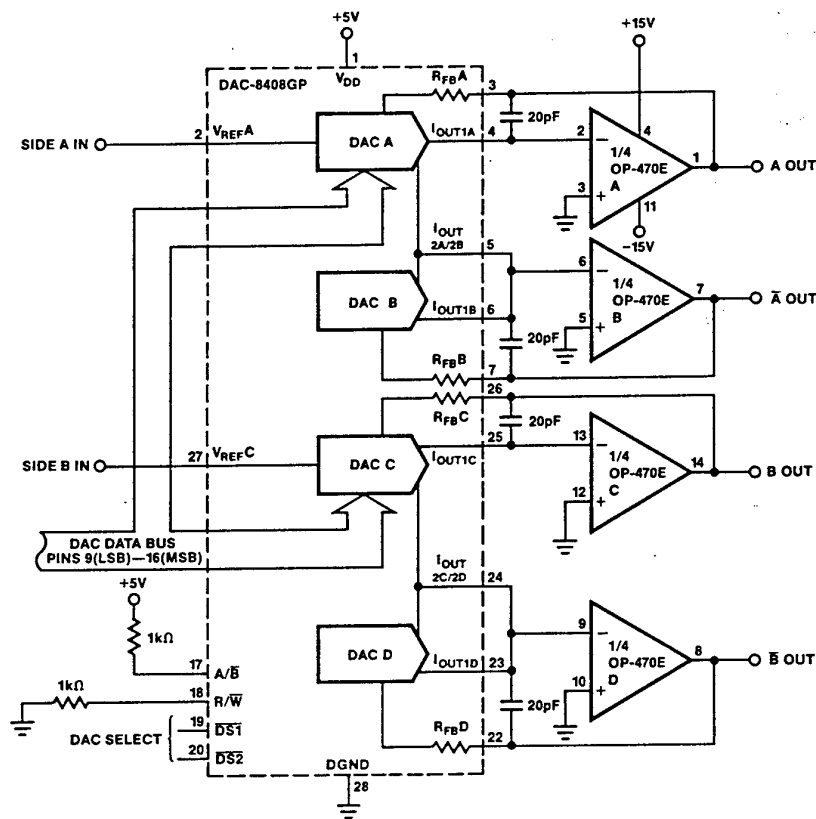
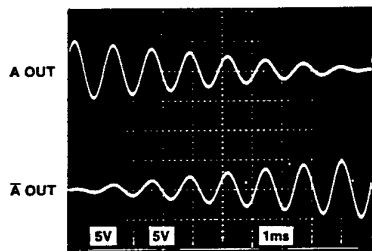


FIGURE 13: Digital Panning Control Output

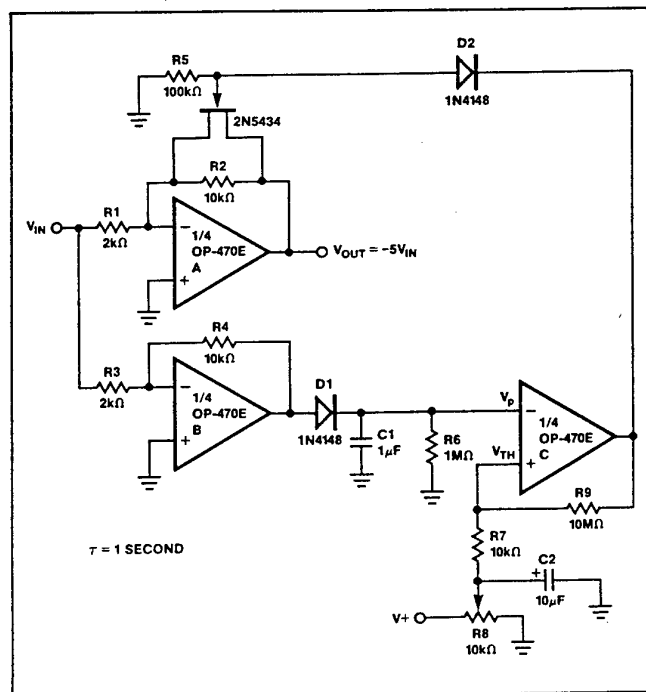


SQUELCH AMPLIFIER

The circuit of Figure 14 is a simple squelch amplifier that uses a FET switch to cut off the output when the input signal falls below a preset limit.

The input signal is sampled by a peak detector with a time constant set by C1 and R6. When the output of the peak detector, V_p , falls below the threshold voltage, V_{TH} , set by R8, the comparator formed by op amp C switches from V^- to V^+ . This drives the gate of the N-channel FET high, turning it ON, reducing the gain of the inverting amplifier formed by op amp A to zero.

FIGURE 14: Squelch Amplifier

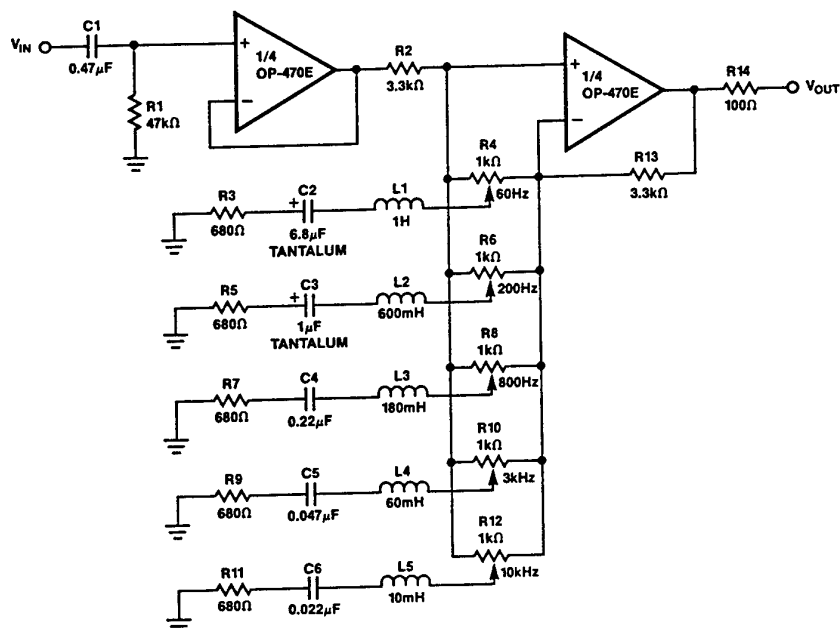


F-BAND LOW NOISE STEREO GRAPHIC EQUALIZER

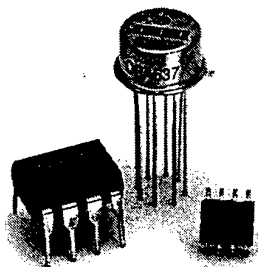
graphic equalizer circuit shown in Figure 15 provides of boost or cut over a 5-band range. Signal-to-noise

ratio over a 20kHz bandwidth is better than 100dB referred to a 3V rms input. Larger inductors can be replaced by active inductors but this reduces the signal-to-noise ratio.

FIGURE 15: 5-Band Low Noise Graphic Equalizer



For Immediate Assistance, Contact Your Local Salesperson



OPA627
OPA637

AVAILABLE IN DIE

Precision High-Speed *Difet*® OPERATIONAL AMPLIFIERS

FEATURES

- VERY LOW NOISE: $4.5\text{nV}/\sqrt{\text{Hz}}$ at 10kHz
- FAST SETTLING TIME:
OPA627—550ns to 0.01%
OPA637—450ns to 0.01%
- LOW V_{OS} : 100 μV max
- LOW DRIFT: 0.8 $\mu\text{V}/^\circ\text{C}$ max
- LOW I_B : 5pA max
- OPA627: Unity-Gain Stable
- OPA637: Stable in Gain ≥ 5

DESCRIPTION

The OPA627 and OPA637 *Difet* operational amplifiers provide a new level of performance in a precision FET op amp. When compared to the popular OPA111 op amp, the OPA627/637 has lower noise, lower offset voltage, and much higher speed. It is useful in a broad range of precision and high speed analog circuitry.

The OPA627/637 is fabricated on a high-speed, dielectrically-isolated complementary NPN/PNP process. It operates over a wide range of power supply voltage— $\pm 4.5\text{V}$ to $\pm 18\text{V}$. Laser-trimmed *Difet* input circuitry provides high accuracy and low-noise performance comparable with the best bipolar-input op amps.

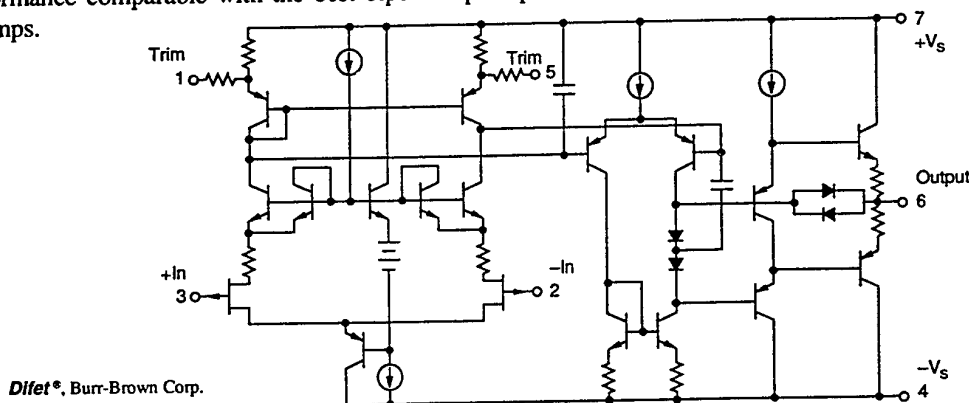
APPLICATIONS

- PRECISION INSTRUMENTATION
- FAST DATA ACQUISITION
- DAC OUTPUT AMPLIFIER
- OPTOELECTRONICS
- SONAR, ULTRASOUND
- HIGH-IMPEDANCE SENSOR AMPS
- HIGH-PERFORMANCE AUDIO CIRCUITRY
- ACTIVE FILTERS

High frequency complementary transistors allow increased circuit bandwidth, attaining dynamic performance not possible with previous precision FET op amps. The OPA627 is unity-gain stable. The OPA637 is stable in gains equal to or greater than five.

Difet fabrication achieves extremely low input bias currents without compromising input voltage noise performance. Low input bias current is maintained over a wide input common-mode voltage range with unique cascode circuitry.

The OPA627/637 is available in plastic DIP, SOIC and metal TO-99 packages. Industrial and military temperature range models are available.



Difet®, Burr-Brown Corp.

International Airport Industrial Park • Mailing Address: PO Box 11400 • Tucson, AZ 85734 • Street Address: 6730 S. Tucson Blvd. • Tucson, AZ 85706
Tel: (602) 746-1111 • Twx: 910-952-1111 • Cable: BBRCORP • Telex: 066-6491 • FAX: (602) 889-1510 • Immediate Product Info: (800) 548-6132

SPECIFICATIONS

ELECTRICAL

 $T_A = +25^\circ\text{C}$, $V_S = \pm 15\text{V}$ unless otherwise noted.

PARAMETER	CONDITIONS	OPA627BM/BP/SM OPA637BM/BP/SM			OPA627AM/AP/AU OPA637AM/AP/AU			UNITS
		MIN	TYP	MAX	MIN	TYP	MAX	
OFFSET VOLTAGE ⁽¹⁾								
Input Offset Voltage			40	100		130	250	μV
AP, BP, AU Grades			100	250		280	500	μV
Average Drift			0.4	0.8		1.2	2	$\mu\text{V}/^\circ\text{C}$
AP, BP, AU Grades			0.8	2		2.5		$\mu\text{V}/^\circ\text{C}$
Power Supply Rejection	$V_S = \pm 4.5$ to $\pm 18\text{V}$	106	120		100	116		dB
INPUT BIAS CURRENT ⁽²⁾								
Input Bias Current	$V_{CM} = 0\text{V}$		1	5		2	10	pA
Over Specified Temperature	$V_{CM} = 0\text{V}$			1			2	nA
SM Grade	$V_{CM} = 0\text{V}$			50				nA
Over Common-Mode Voltage	$V_{CM} = \pm 10\text{V}$		1			2		pA
Input Offset Current	$V_{CM} = 0\text{V}$		0.5	5		1	10	pA
Over Specified Temperature	$V_{CM} = 0\text{V}$			1			2	nA
SM Grade	$V_{CM} = 0\text{V}$			50				nA
NOISE								
Input Voltage Noise								$\text{nV}/\sqrt{\text{Hz}}$
Noise Density: $f = 10\text{Hz}$			15	40		20		$\text{nV}/\sqrt{\text{Hz}}$
$f = 100\text{Hz}$			8	20		10		$\text{nV}/\sqrt{\text{Hz}}$
$f = 1\text{kHz}$			5.2	8		5.6		$\text{nV}/\sqrt{\text{Hz}}$
$f = 10\text{kHz}$			4.5	6		4.8		$\text{nV}/\sqrt{\text{Hz}}$
Voltage Noise, BW = 0.1 to 10Hz			0.6	1.6		0.8		$\mu\text{Vp-p}$
Input Bias Current Noise								$\text{fA}/\sqrt{\text{Hz}}$
Noise Density, $f = 100\text{Hz}$			1.6	2.5		2.5		$\text{fA}/\sqrt{\text{Hz}}$
Current Noise, BW = 0.1 to 10Hz			30	60		48		fA-p
INPUT IMPEDANCE								
Differential			$10^{12} \parallel 8$			.		$\Omega \parallel \text{pF}$
Common-Mode			$10^{12} \parallel 7$			.		$\Omega \parallel \text{pF}$
INPUT VOLTAGE RANGE								
Common-Mode Input Range		± 11	± 11.5		.	.		V
Over Specified Temperature		± 10.5	± 11		.	.		V
Common-Mode Rejection	$V_{CM} = \pm 10.5\text{V}$	106	116		100	110		dB
OPEN-LOOP GAIN								
Open-Loop Voltage Gain	$V_O = \pm 10\text{V}$, $R_L = 1\text{k}\Omega$	112	120		106	116		dB
Over Specified Temperature	$V_O = \pm 10\text{V}$, $R_L = 1\text{k}\Omega$	106	117		100	110		dB
SM Grade	$V_O = \pm 10\text{V}$, $R_L = 1\text{k}\Omega$	100	114					dB
FREQUENCY RESPONSE								
Slew Rate: OPA627	$G = -1$, 10V Step	40	55		.	.		V/ μs
OPA637	$G = -4$, 10V Step	100	135		.	.		V/ μs
Settling Time: OPA627 0.01%	$G = -1$, 10V Step		550		.	.		ns
0.1%	$G = -1$, 10V Step		450		.	.		ns
OPA637 0.01%	$G = -4$, 10V Step		450		.	.		ns
0.1%	$G = -4$, 10V Step		300		.	.		ns
Gain-Bandwidth Product: OPA627	$G = 1$		16		.	.		MHz
OPA637	$G = 10$		80		.	.		MHz
Total Harmonic Distortion + Noise	$G = +1$, $f = 1\text{kHz}$		0.00003		.	.		%
POWER SUPPLY								
Specified Operating Voltage		± 4.5	± 15	± 18	.	.	.	V
Operating Voltage Range			± 7	± 7.5	.	.	.	V
Current					.	.	.	mA
OUTPUT								
Voltage Output	$R_L = 1\text{k}\Omega$	± 11.5	± 12.3		.	.		V
Over Specified Temperature		± 11	± 11.5		.	.		V
Current Output	$V_O = \pm 10\text{V}$		± 45		.	.		mA
Short Circuit Current		± 35	+70/-55	± 100	.	.	.	mA
Output Impedance, Open-Loop	1MHz		55		.	.		Ω
TEMPERATURE RANGE								
Specification: AP, BP, AM, BM, AU		-25		+85	.	.	.	$^\circ\text{C}$
SM		-55		+125	.	.	.	$^\circ\text{C}$
Storage: AM, BM, SM		-60		+150	.	.	.	$^\circ\text{C}$
AP, BP, AU		-40		+125	.	.	.	$^\circ\text{C}$
θ_{JA} : AM, BM, SM			200		.	.	.	$^\circ\text{C/W}$
AP, BP			100		.	.	.	$^\circ\text{C/W}$
AU			160		.	.	.	$^\circ\text{C/W}$

* Specifications same as "B" grade.

NOTES: (1) Offset voltage measured fully warmed-up. (2) High-speed test at $T_J = 25^\circ\text{C}$. See Typical Performance Curves for warmed-up performance.

The information provided herein is believed to be reliable; however, BURR-BROWN assumes no responsibility for inaccuracies or omissions. BURR-BROWN assumes no responsibility for the use of this information, and all use of such information shall be entirely at the user's own risk. Prices and specifications are subject to change without notice. No patent rights or licenses to any of the circuits described herein are implied or granted to any third party. BURR-BROWN does not authorize or warrant any BURR-BROWN product for use in life support devices and/or systems.

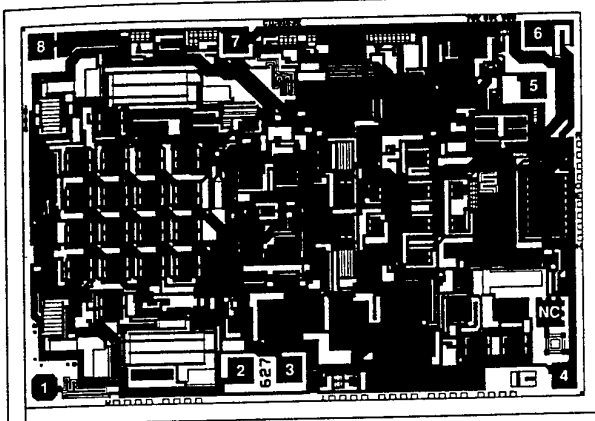
BURR-BROWN®



Burr-Brown IC Data Book—Linear Products

For Immediate Assistance, Contact Your Local Salesperson

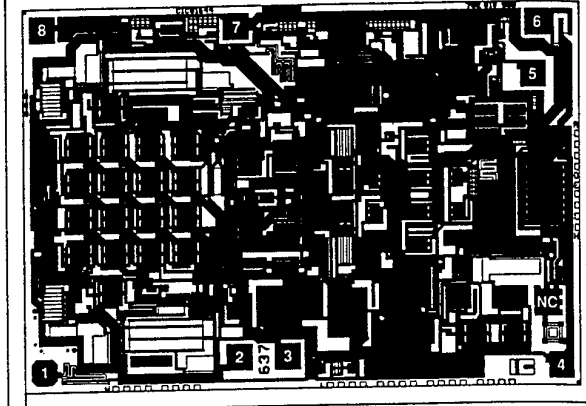
DICE INFORMATION



OPA627 DIE TOPOGRAPHY

PAD	FUNCTION	PAD	FUNCTION
1	Offset Trim	5	Offset Trim
2	-In	6	Output
3	+In	7	+V _S
4	-V _S	8	Substrate
		NC	No Connection

Substrate Bias: Dielectrically isolated. See data sheet for connection options.



OPA637 DIE TOPOGRAPHY

MECHANICAL INFORMATION

	MILS (0.001")	MILLIMETERS
Die Size	117 x 80 ±5	2.97 x 2.03 ±0.13
Die Thickness	20 ±3	0.51 ±0.08
Min. Pad Size	4 x 4	0.10 x 0.10
Transistor Count	46	
Backing:	None	

See "DICE PRODUCTS" Appendix C in Burr-Brown Data Book, or contact factory for current information.

ORDERING INFORMATION

MODEL	PACKAGE	TEMPERATURE RANGE
OPA627AP	Plastic DIP	-25°C to +85°C
OPA627BP	Plastic DIP	-25°C to +85°C
OPA627AU	SOIC	-25°C to +85°C
OPA627AM	TO-99 Metal	-25°C to +85°C
OPA627BM	TO-99 Metal	-25°C to +85°C
OPA627SM	TO-99 Metal	-55°C to +125°C
OPA637AP	Plastic DIP	-25°C to +85°C
OPA637BP	Plastic DIP	-25°C to +85°C
OPA637AU	SOIC	-25°C to +85°C
OPA637AM	TO-99 Metal	-25°C to +85°C
OPA637BM	TO-99 Metal	-25°C to +85°C
OPA637SM	TO-99 Metal	-55°C to +125°C

ABSOLUTE MAXIMUM RATINGS

Supply Voltage	±18V
Input Voltage Range	+V _S + 2V to -V _S - 2V
Differential Input Range	Total V _S + 4V
Power Dissipation	1000mW
Operating Temperature	
M Package	-55°C to +125°C
P, U Package	-40°C to +125°C
Storage Temperature	
M Package	-65°C to +150°C
P, U Package	-40°C to +125°C
Junction Temperature	
M Package	+175°C
P, U Package	+150°C
Lead Temperature (soldering, 10s)	+300°C
SOIC (soldering, 3s)	+260°C

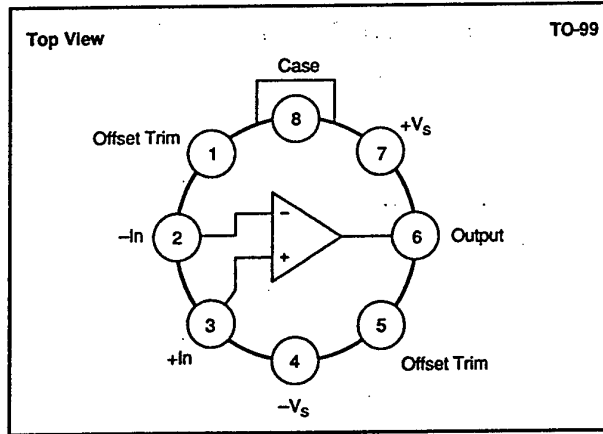
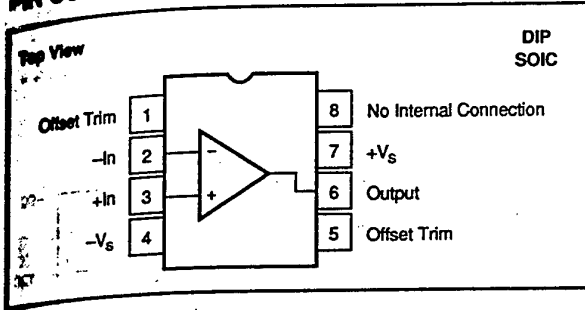
PACKAGE INFORMATION⁽¹⁾

MODEL	PACKAGE	PACKAGE DRAWING NUMBER
OPA627AP	Plastic DIP	006
OPA627BP	Plastic DIP	006
OPA627AU	SOIC	182
OPA627AM	TO-99 Metal	001
OPA627BM	TO-99 Metal	001
OPA627SM	TO-99 Metal	001
OPA637AP	Plastic DIP	006
OPA637BP	Plastic DIP	006
OPA637AU	SOIC	182
OPA637AM	TO-99 Metal	001
OPA637BM	TO-99 Metal	001
OPA637SM	TO-99 Metal	001

NOTE: (1) For detailed drawing and dimension table, please see end of data sheet, or Appendix D of Burr-Brown IC Data Book.

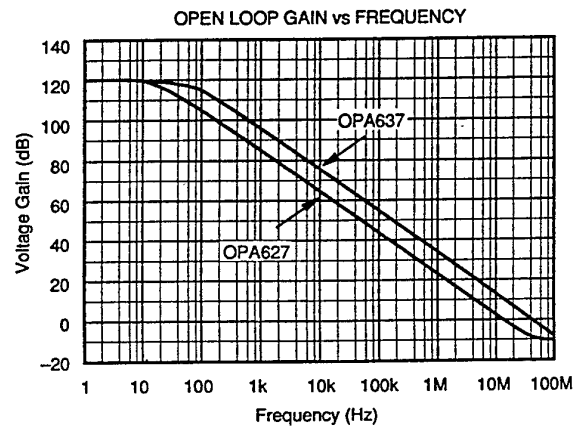
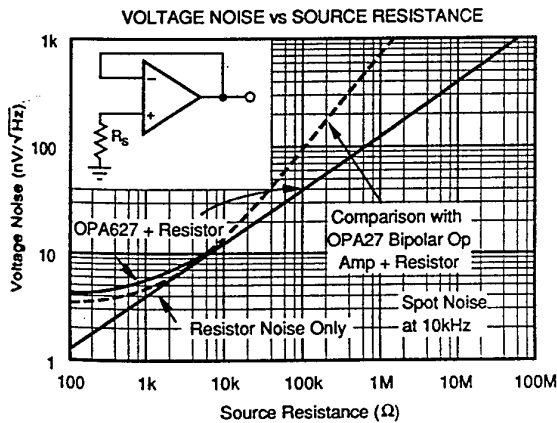
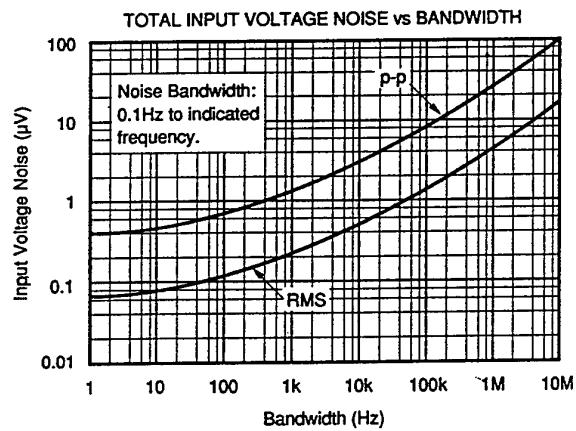
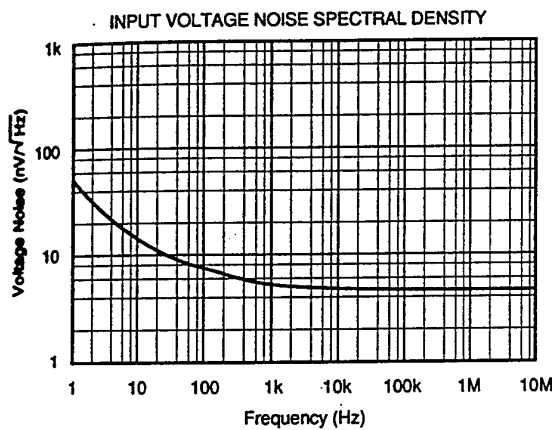
Or, Call Customer Service at 1-800-548-6132 (USA Only)

PIN CONFIGURATIONS



TYPICAL PERFORMANCE CURVES

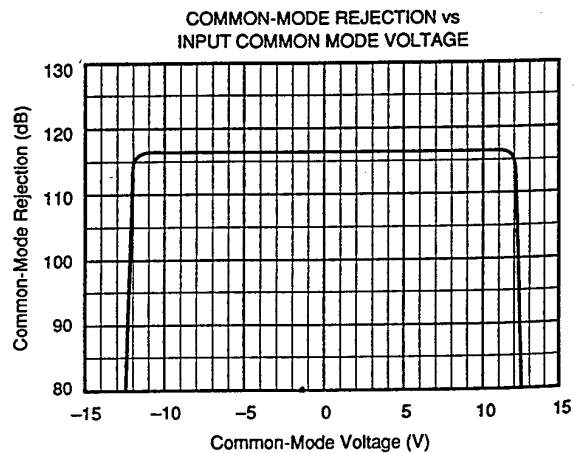
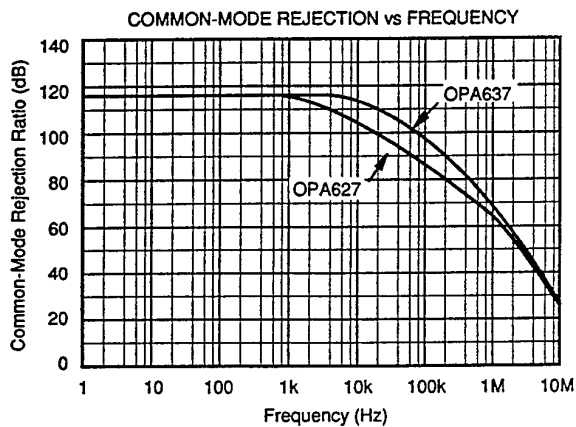
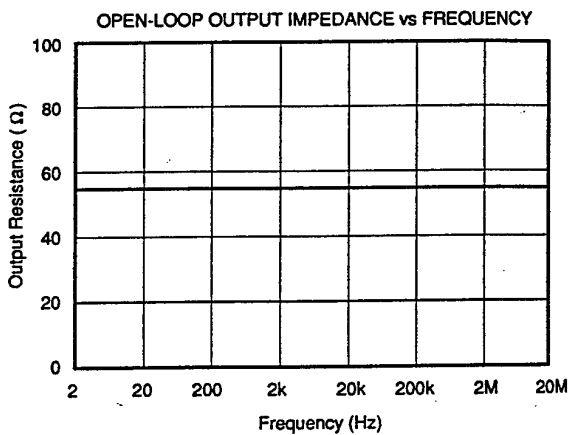
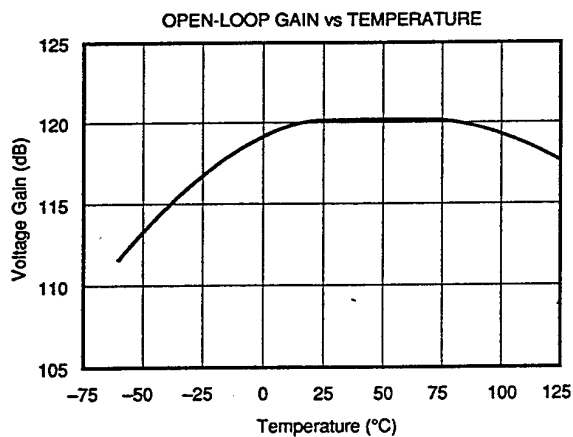
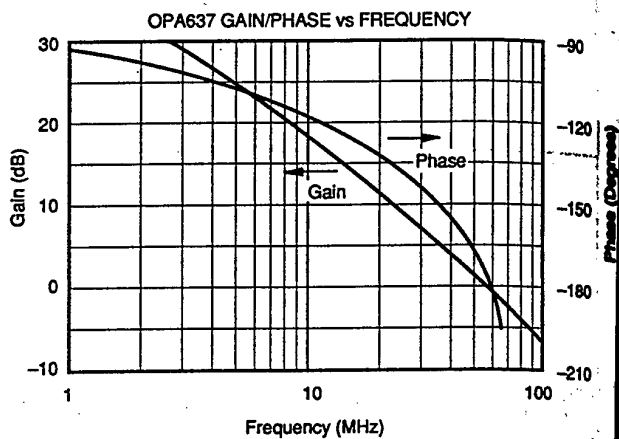
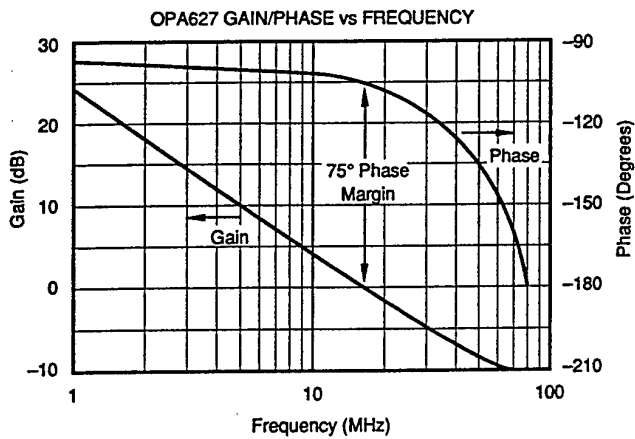
$T_A = +25^\circ\text{C}$, $V_S = \pm 15\text{V}$ unless otherwise noted.



For Immediate Assistance, Contact Your Local Salesperson

TYPICAL PERFORMANCE CURVES (CONT)

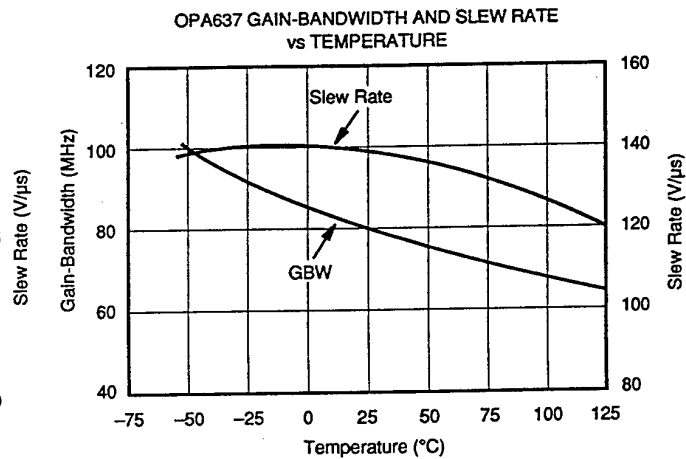
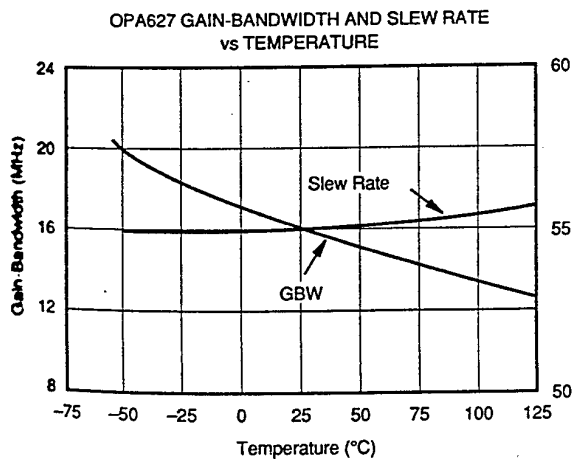
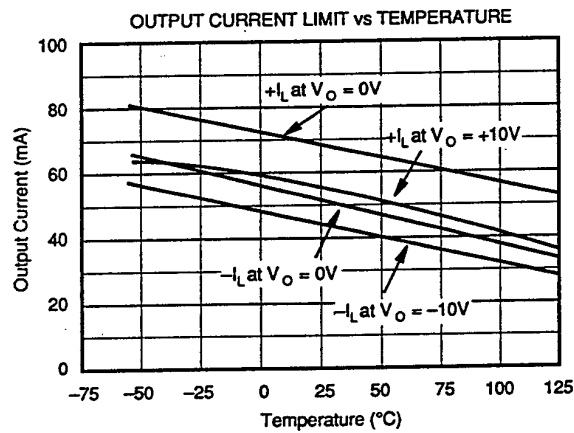
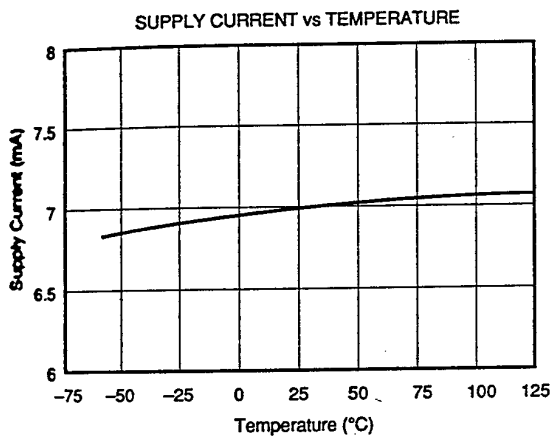
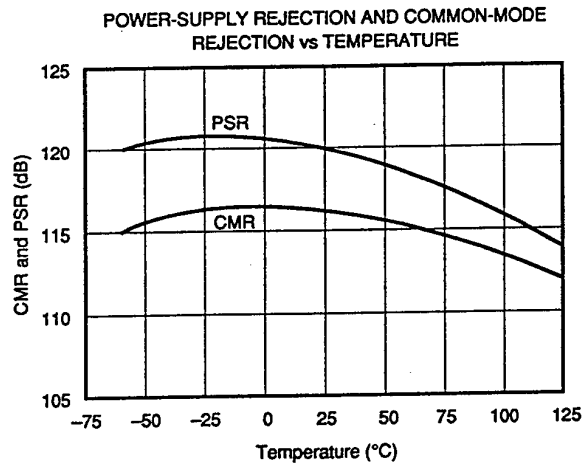
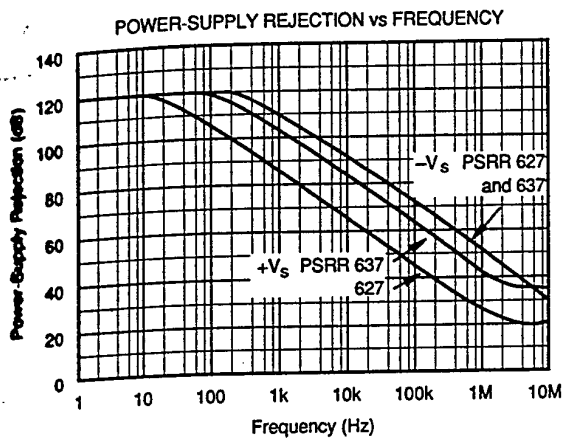
$T_A = +25^\circ\text{C}$, $V_S = \pm 15\text{V}$ unless otherwise noted.



Or, Call Customer Service at 1-800-548-6132 (USA Only)

TYPICAL PERFORMANCE CURVES (CONT)

$T_A = +25^\circ\text{C}$, $V_S = \pm 15\text{V}$ unless otherwise noted.

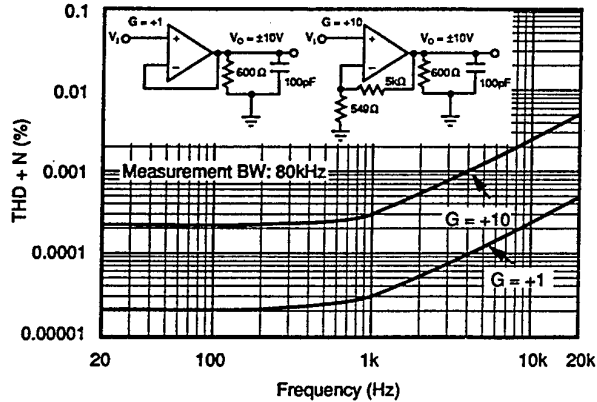


For Immediate Assistance, Contact Your Local Salesperson

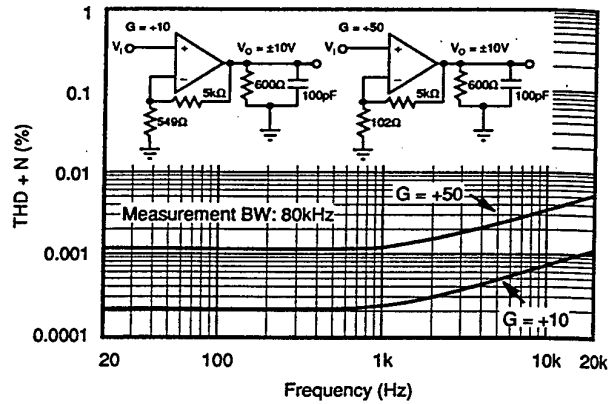
TYPICAL PERFORMANCE CURVES (CONT)

$T_A = +25^\circ\text{C}$, $V_S = \pm 15\text{V}$ unless otherwise noted.

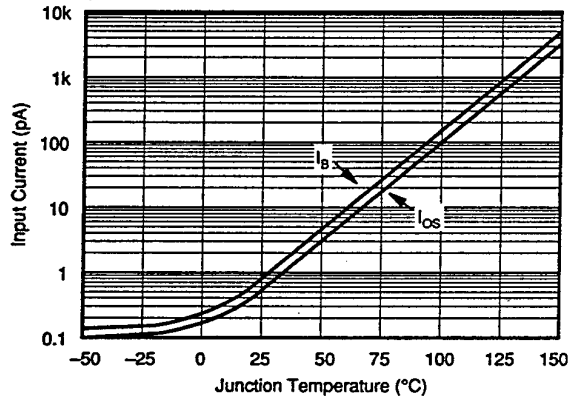
OPA627 TOTAL HARMONIC DISTORTION + NOISE
vs FREQUENCY



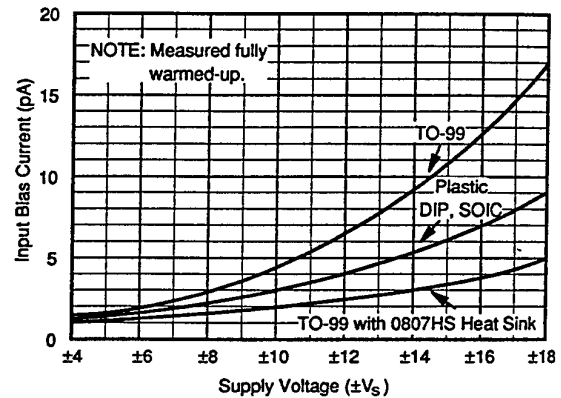
OPA637 TOTAL HARMONIC DISTORTION + NOISE
vs FREQUENCY



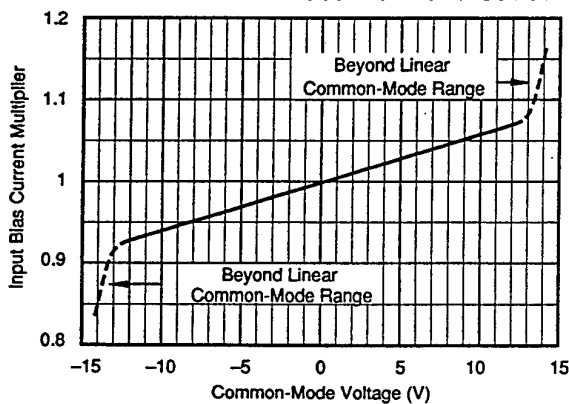
INPUT BIAS AND OFFSET CURRENT
vs JUNCTION TEMPERATURE



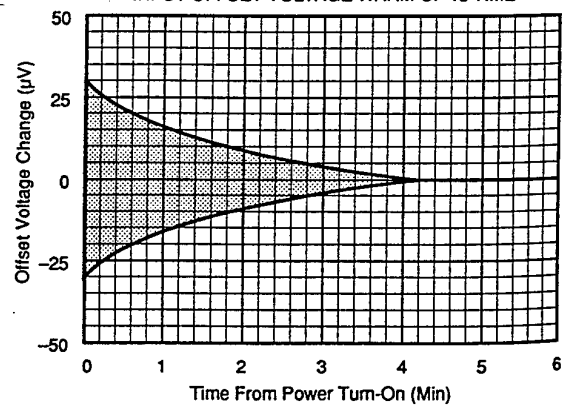
INPUT BIAS CURRENT
vs POWER SUPPLY VOLTAGE



INPUT BIAS CURRENT vs COMMON-MODE VOLTAGE

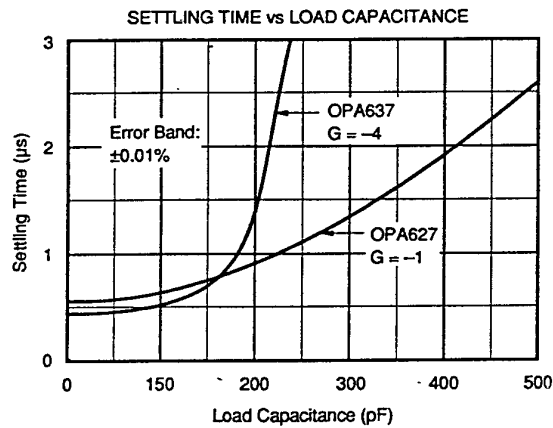
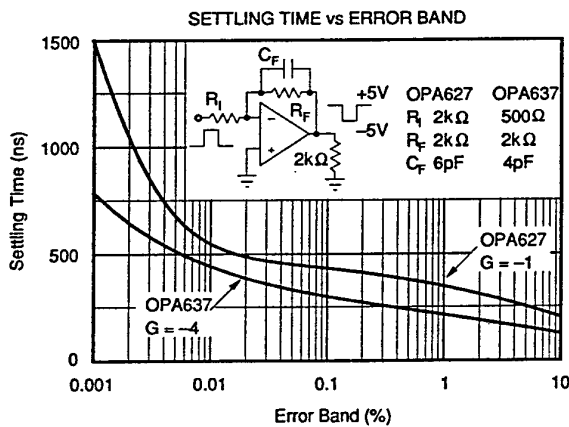
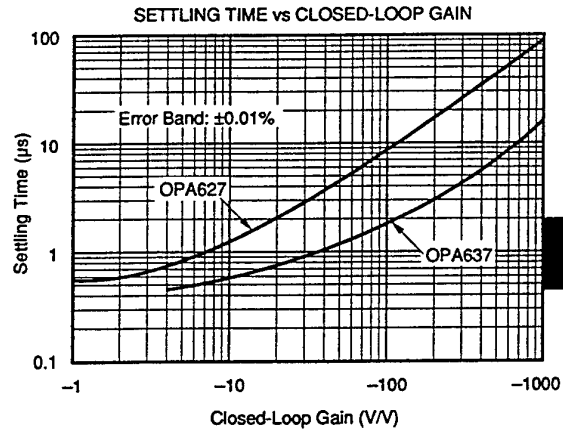
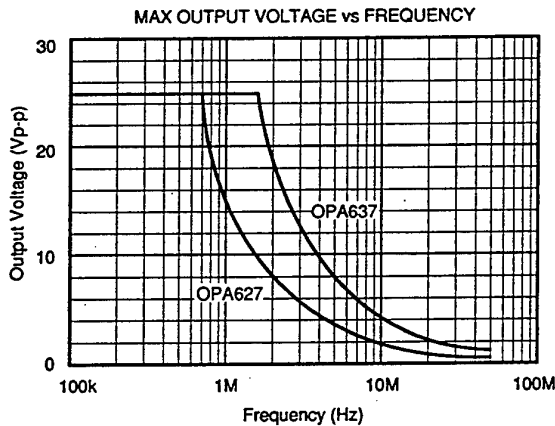


INPUT OFFSET VOLTAGE WARM-UP vs TIME



TYPICAL PERFORMANCE CURVES (CONT)

$T_A = +25^\circ\text{C}$, $V_E = \pm 15\text{V}$ unless otherwise noted.



APPLICATIONS INFORMATION

The OPA627 is unity-gain stable. The OPA637 may be used to achieve higher speed and bandwidth in circuits with noise gain greater than five. Noise gain refers to the closed-loop gain of a circuit as if the non-inverting op amp input were being driven. For example, the OPA637 may be used in a non-inverting amplifier with gain greater than five, or an inverting amplifier of gain greater than four.

When choosing between the OPA627 or OPA637, it is important to consider the high frequency noise gain of your circuit configuration. Circuits with a feedback capacitor (Figure 1) place the op amp in unity noise-gain at high frequency. These applications must use the OPA627 for proper stability. An exception is the circuit in Figure 2, where a small feedback capacitance is used to compensate for the input capacitance at the op amp's inverting input. In this case, the closed-loop noise gain remains constant with frequency, so if the closed-loop gain is equal to five or greater, the OPA637 may be used.

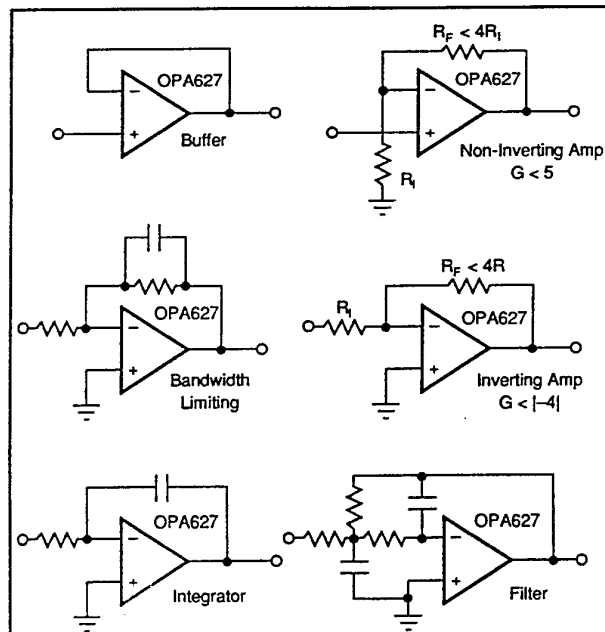


FIGURE 1. Circuits with Noise Gain Less than Five Require the OPA627 for Proper Stability.

For Immediate Assistance, Contact Your Local Salesperson

OFFSET VOLTAGE ADJUSTMENT

The OPA627/637 is laser-trimmed for low offset voltage and drift, so many circuits will not require external adjustment. Figure 3 shows the optional connection of an external potentiometer to adjust offset voltage. This adjustment should not be used to compensate for offsets created elsewhere in a system (such as in later amplification stages or in an A/D converter) because this could introduce excessive temperature drift. Generally, the offset drift will change by approximately $4\mu\text{V}/^\circ\text{C}$ for 1mV of change in the offset voltage due to an offset adjustment (as shown on Figure 3).

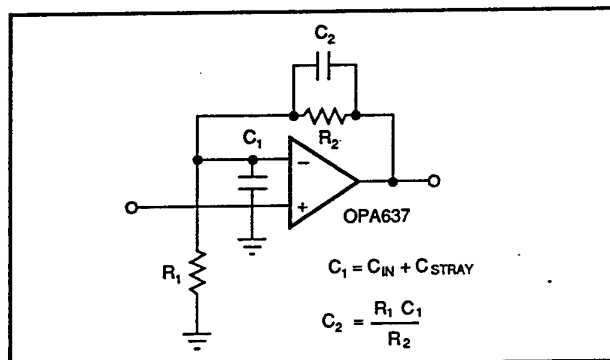


FIGURE 2. Circuits with Noise Gain Equal to or Greater than Five May Use the OPA637.

NOISE PERFORMANCE

Some bipolar op amps may provide lower voltage noise performance, but both voltage noise and bias current noise contribute to the total noise of a system. The OPA627/637 is unique in providing very low voltage noise and very low current noise. This provides optimum noise performance over a wide range of sources, including reactive source impedances. This can be seen in the performance curve showing the noise of a source resistor combined with the

noise of an OPA627. Above a $2\text{k}\Omega$ source resistance, the op amp contributes little additional noise. Below $1\text{k}\Omega$, op amp noise dominates over the resistor noise, but compares favorably with precision bipolar op amps.

CIRCUIT LAYOUT

As with any high speed, wide bandwidth circuit, careful layout will ensure best performance. Make short, direct interconnections and avoid stray wiring capacitance—especially at the input pins and feedback circuitry.

The case connection (pin 8 of TO-99 metal package only) should be connected to an AC ground for lowest possible pickup of external fields. While DC ground would be the most likely choice, pin 8 could also be connected to either power supply. (The case is not internally connected to the negative power supply as it is with most common op amps.) For lowest possible input bias current, the case may be driven as a guard—see Input Bias Current section. Pin 8 of the plastic DIP and SOIC versions has no internal connection.

Power supply connections should be bypassed with good high frequency capacitors positioned close to the op amp

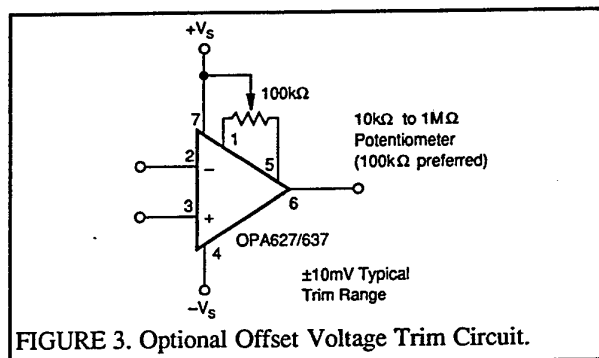


FIGURE 3. Optional Offset Voltage Trim Circuit.

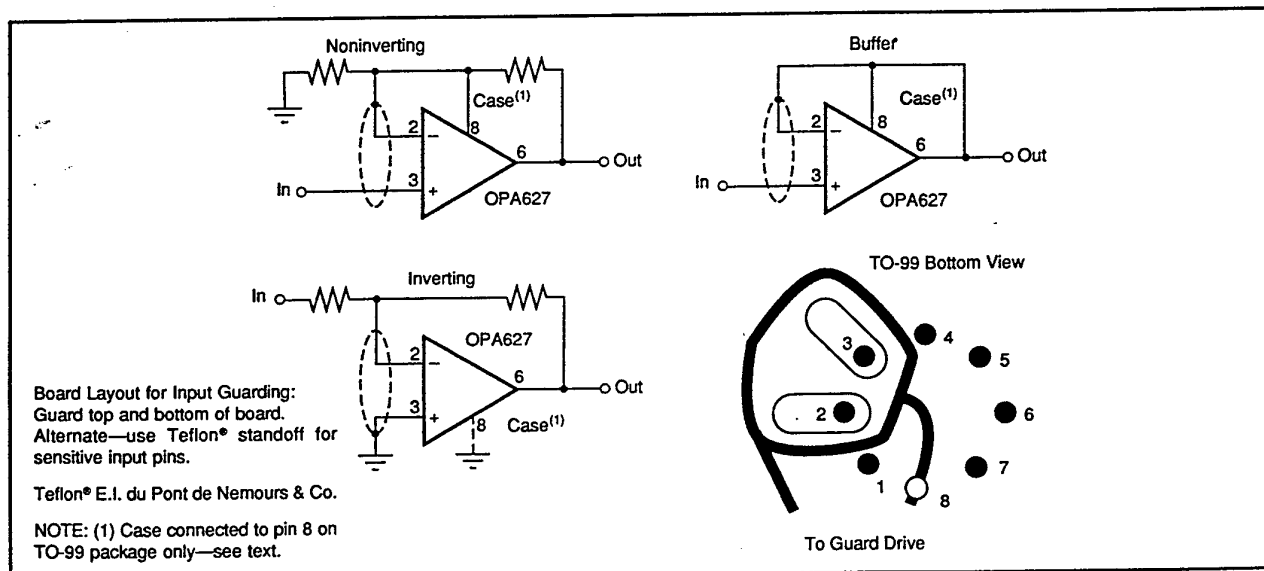


FIGURE 4. Connection of Input Guard for Lowest I_B .

pins. In most cases 0.1μF ceramic capacitors are adequate. The OPA627/637 is capable of high output current (in excess of 45mA). Applications with low impedance loads or capacitive loads with fast transient signals demand large currents from the power supplies. Larger bypass capacitors such as 1μF solid tantalum capacitors may improve dynamic performance in these applications.

INPUT BIAS CURRENT

Difet fabrication of the OPA627/637 provides very low input bias current. Since the gate current of a FET doubles approximately every 10°C, to achieve lowest input bias current, the die temperature should be kept as low as possible. The high speed and therefore higher quiescent current of the OPA627/637 can lead to higher chip temperature. A simple press-on heat sink such as the Burr-Brown model 807HS (TO-99 metal package) can reduce chip temperature by approximately 15°C, lowering the I_B to one-third its warmed-up value. The 807HS heat sink can also reduce low-frequency voltage noise caused by air currents and thermoelectric effects. See the data sheet on the 807HS for details.

Temperature rise in the plastic DIP and SOIC packages can be minimized by soldering the device to the circuit board. Wide copper traces will also help dissipate heat.

The OPA627/637 may also be operated at reduced power supply voltage to minimize power dissipation and temperature rise. Using $\pm 5\text{V}$ power supplies reduces power dissipation to one-third of that at $\pm 15\text{V}$. This reduces the I_B of TO-99 metal package devices to approximately one-fourth the value at $\pm 15\text{V}$.

Leakage currents between printed circuit board traces can easily exceed the input bias current of the OPA627/637. A circuit board “guard” pattern (Figure 4) reduces leakage effects. By surrounding critical high impedance input circuitry with a low impedance circuit connection at the same potential, leakage current will flow harmlessly to the low-impedance node. The case connection (TO-99 metal pack-

age only) may also be driven at guard potential to minimize any leakage which might occur from the input pins to the case. The case is not internally connected to $-V_s$.

Input bias current may also be degraded by improper handling or cleaning. Contamination from handling parts and circuit boards may be removed with cleaning solvents and deionized water. Each rinsing operation should be followed by a 30-minute bake at 85°C.

Many FET-input op amps exhibit large changes in input bias current with changes in input voltage. Input stage cascode circuitry makes the input bias current of the OPA627/637 virtually constant with wide common-mode voltage changes. This is ideal for accurate high input-impedance buffer applications.

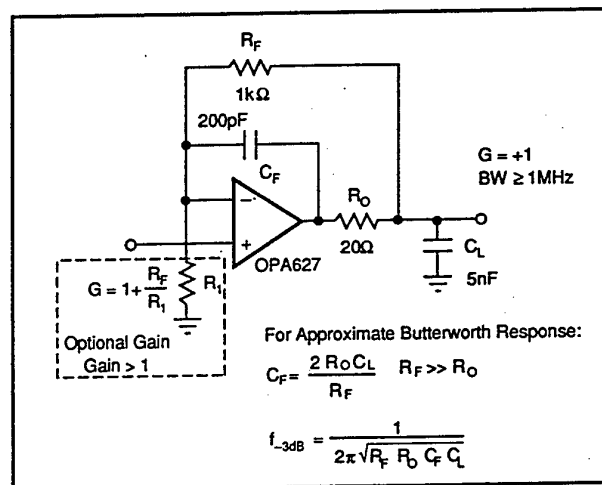


FIGURE 6. Driving Large Capacitive Loads.

PHASE-REVERSAL PROTECTION

The OPA627/637 has internal phase-reversal protection. Many FET-input op amps exhibit a phase reversal when the input is driven beyond its linear common-mode range. This is most often encountered in non-inverting circuits when the input is driven below -12V , causing the output to reverse into the positive rail. The input circuitry of the OPA627/637 does not induce phase reversal with excessive common-mode voltage, so the output limits into the appropriate rail.

OUTPUT OVERLOAD

When the inputs to the OPA627/637 are overdriven, the output voltage of the OPA627/637 smoothly limits at approximately 2.5V from the positive and negative power supplies. If driven to the negative swing limit, recovery takes approximately 500ns. When the output is driven into the positive limit, recovery takes approximately 6 μ s. Output recovery of the OPA627 can be improved using the output clamp circuit shown in Figure 5. Diodes at the inverting input prevent degradation of input bias current.

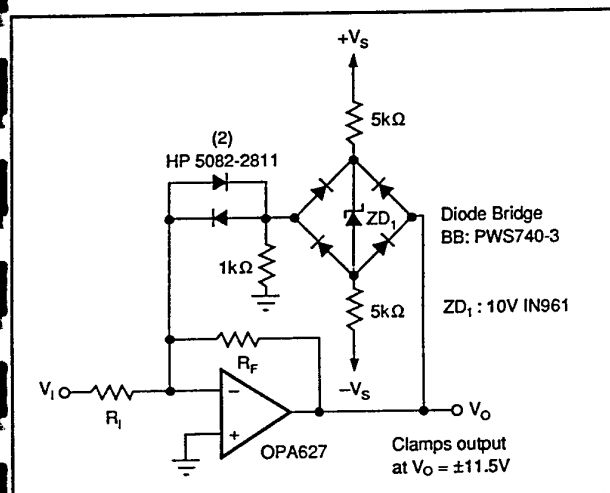


FIGURE 5. Clamp Circuit for Improved Overload Recovery.

For Immediate Assistance, Contact Your Local Salesperson

CAPACITIVE LOADS

As with any high-speed op amp, best dynamic performance can be achieved by minimizing the capacitive load. Since a load capacitance presents a decreasing impedance at higher frequency, a load capacitance which is easily driven by a slow op amp can cause a high-speed op amp to perform poorly. See the typical curves showing settling times as a function of capacitive load. The lower bandwidth of the OPA627 makes it the better choice for driving large capacitive loads. Figure 6 shows a circuit for driving very large load capacitance. This circuit's two-pole response can also be used to sharply limit system bandwidth. This is often useful in reducing the noise of systems which do not require the full bandwidth of the OPA627.

INPUT PROTECTION

The inputs of the OPA627/637 are protected for voltages between $+V_s + 2V$ and $-V_s - 2V$. If the input voltage can exceed these limits, the amplifier should be protected. The diode clamps shown in Figure 7a will prevent the input voltage from exceeding one forward diode voltage drop beyond the power supplies—well within the safe limits. If the input source can deliver current in excess of the maximum forward current of the protection diodes, use a series resistor, R_s , to limit the current. Be aware that adding resistance to the input will increase noise. The $4nV/\sqrt{Hz}$ theoretical thermal noise of a $1k\Omega$ resistor will add to the $4.5nV/\sqrt{Hz}$ noise of the OPA627/637 (by the square-root of the sum of the squares), producing a total noise of $6nV/\sqrt{Hz}$. Resistors below 100Ω add negligible noise.

Leakage current in the protection diodes can increase the total input bias current of the circuit. The specified maximum leakage current for commonly used diodes such as the 1N4148 is approximately 25nA—more than a thousand

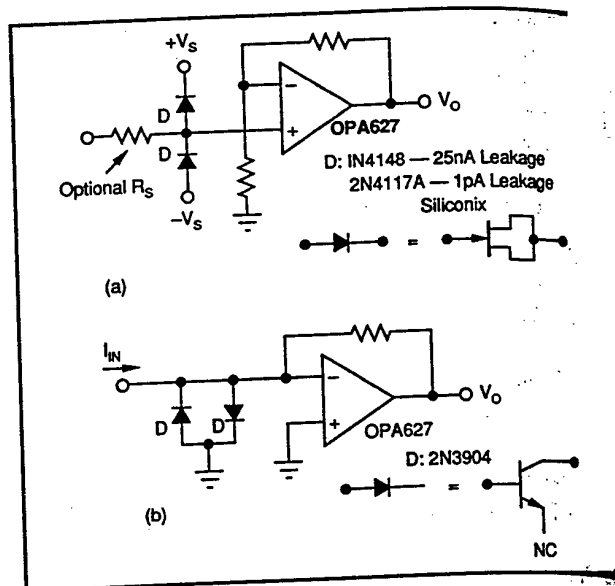


FIGURE 7. Input Protection Circuits.

times larger than the input bias current of the OPA627/637. Leakage current of these diodes is typically much lower and may be adequate in many applications. Light falling on the junction of the protection diodes can dramatically increase leakage current, so common glass-packaged diodes should be shielded from ambient light. Very low leakage can be achieved by using a diode-connected FET as shown. The 2N4117A is specified at 1pA and its metal case shields the junction from light.

Sometimes input protection is required on I/V converters or inverting amplifiers (Figure 7b). Although in normal operation, the voltage at the summing junction will be near zero (equal to the offset voltage of the amplifier), large input transients may cause this node to exceed 2V beyond

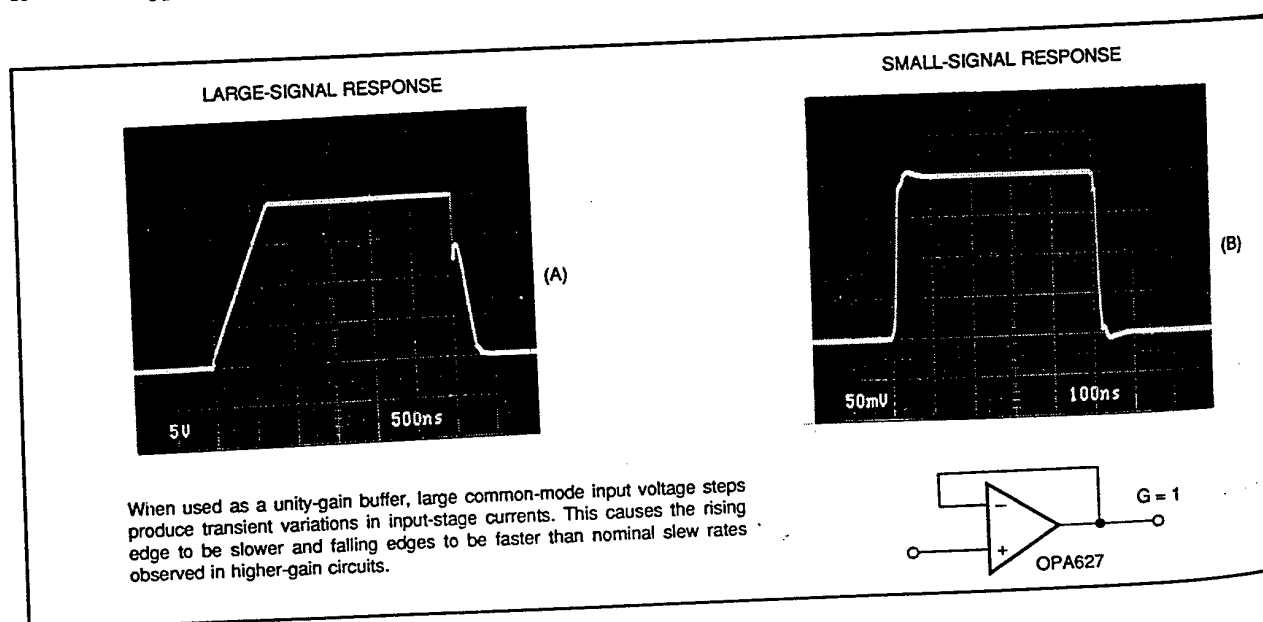


FIGURE 8. OPA627 Dynamic Performance, $G = +1$.

Or, Call Customer Service at 1-800-548-6132 (USA Only)

power supplies. In this case, the summing junction should be protected with diode clamps connected to ground. Even with the low voltage present at the summing junction, common signal diodes may have excessive leakage current.

Since the reverse voltage on these diodes is clamped, a diode-connected signal transistor can be used as an inexpensive low leakage diode (Figure 7b).

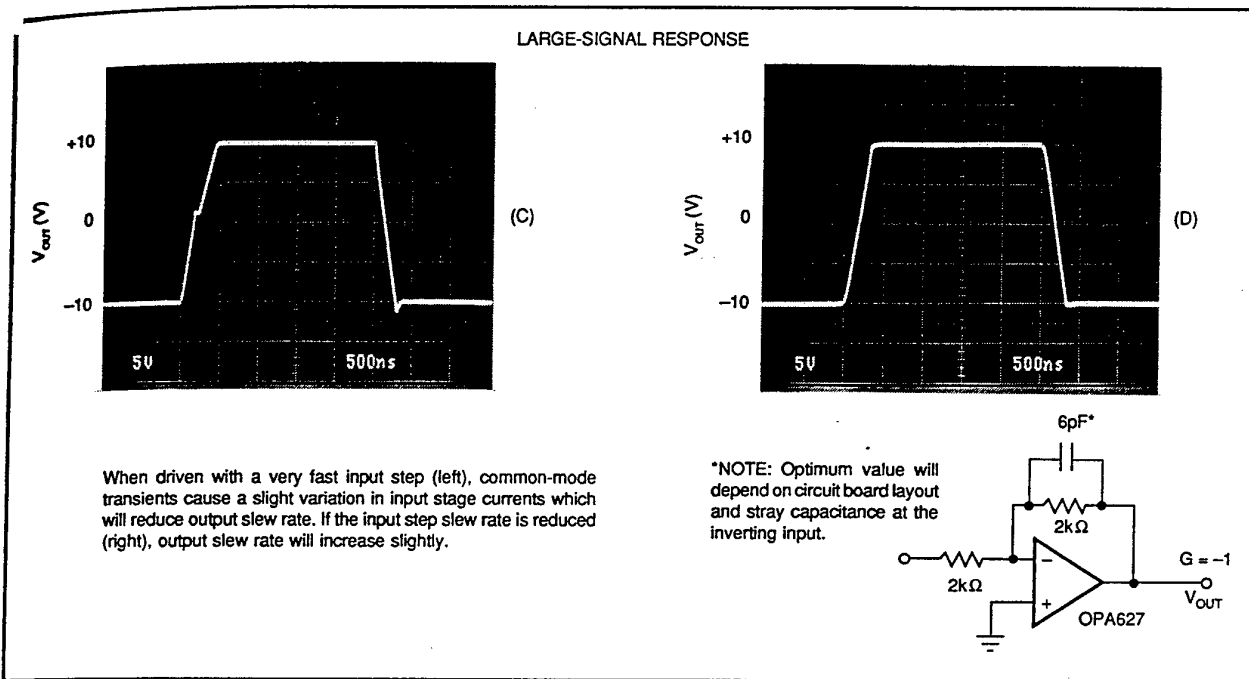


FIGURE 9. OPA627 Dynamic Performance, $G = -1$.

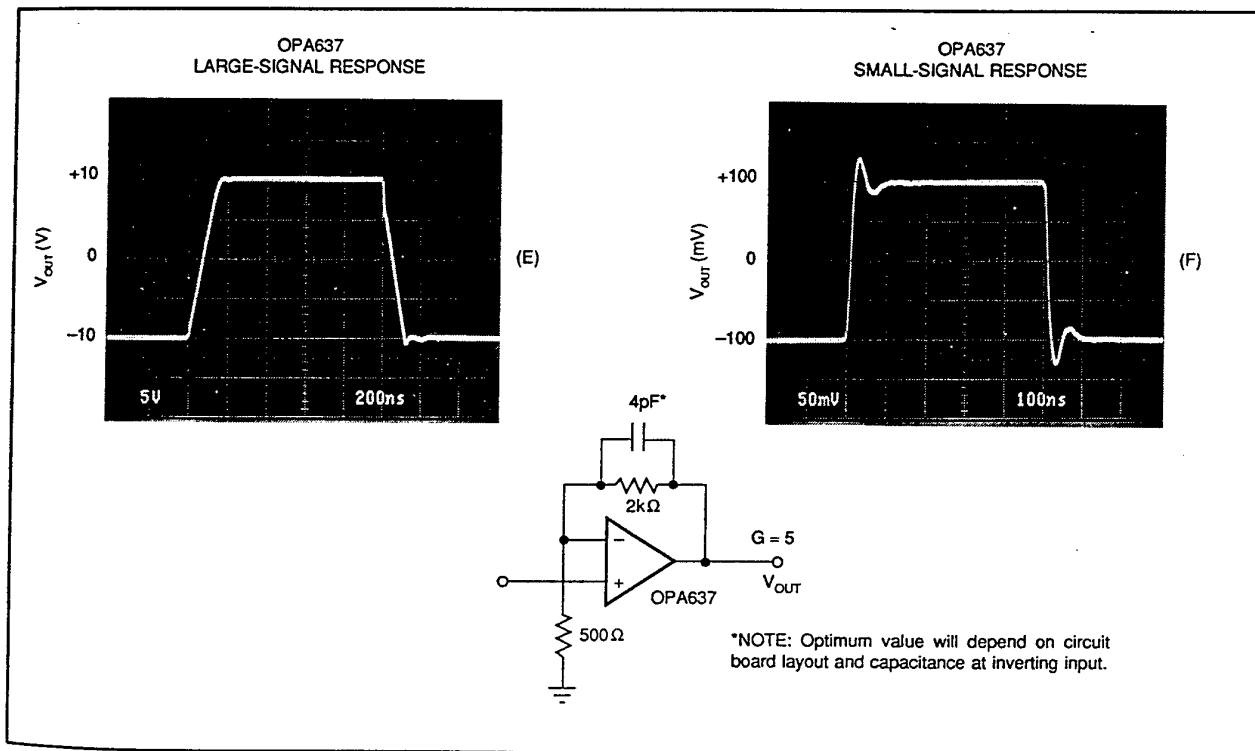


FIGURE 10. OPA637 Dynamic Response, $G = 5$.

For Immediate Assistance, Contact Your Local Salesperson

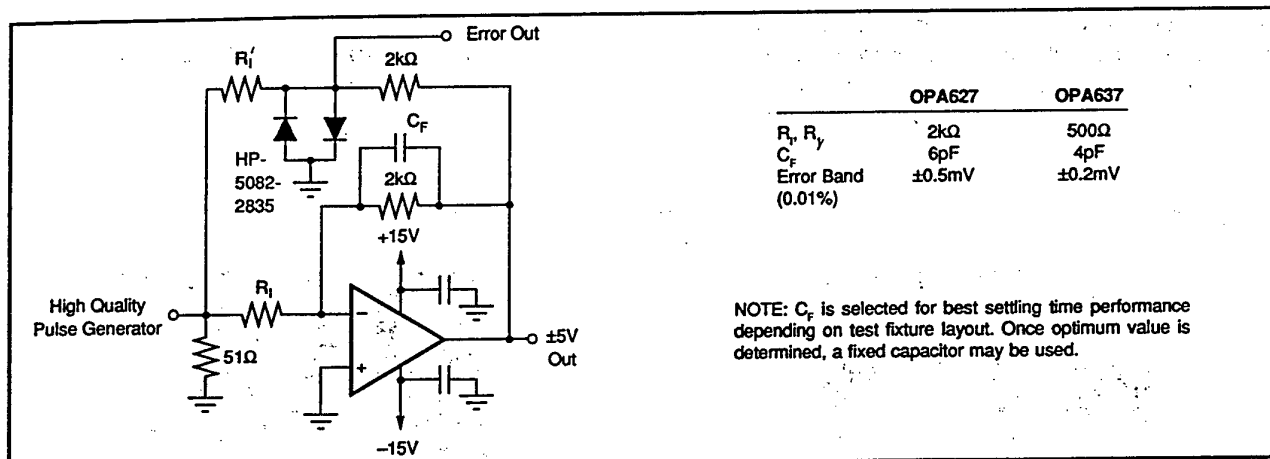


FIGURE 11. Settling Time and Slew Rate Test Circuit.

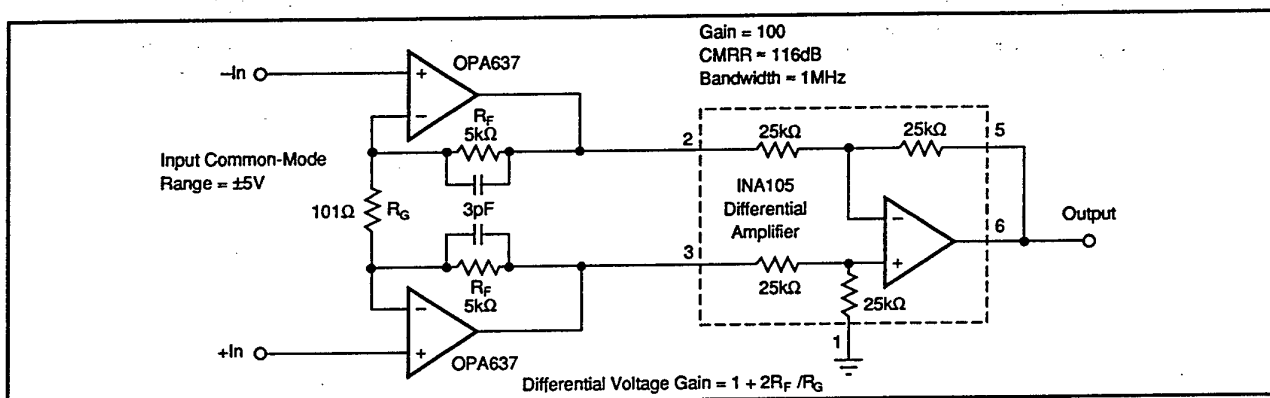


FIGURE 12. High Speed Instrumentation Amplifier, Gain = 100.

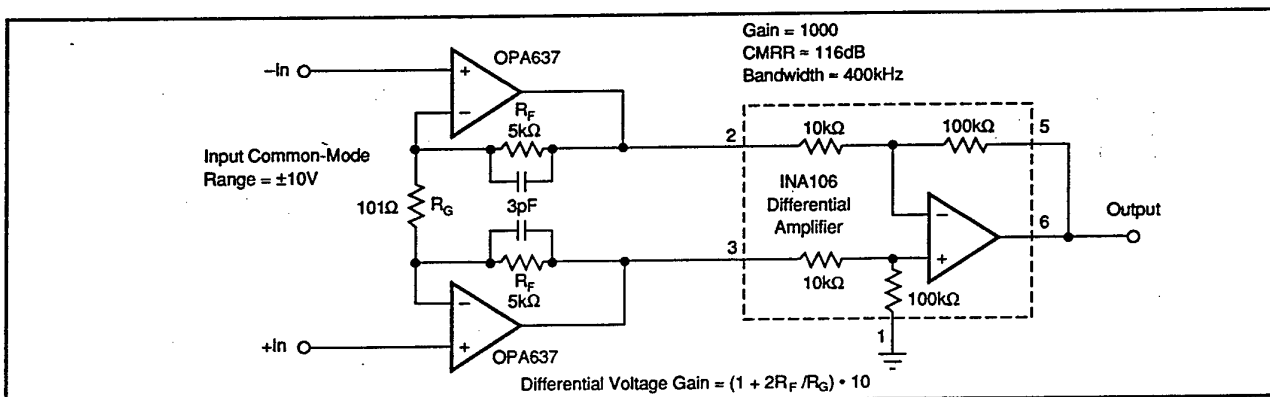


FIGURE 13. High Speed Instrumentation Amplifier, Gain = 1000.

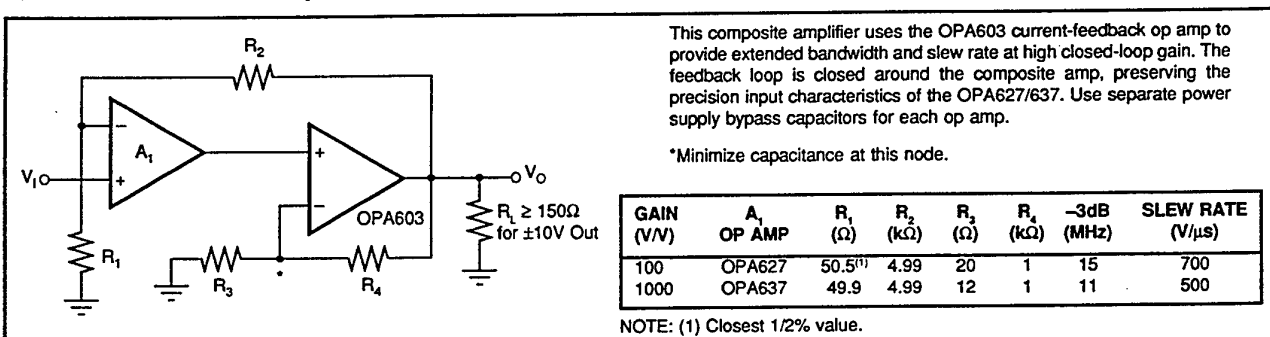
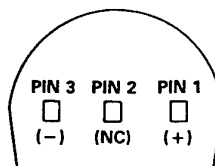


FIGURE 14. Composite Amplifier for Wide Bandwidth.

FEATURES

High Precalibrated Accuracy: 0.5°C max @ 25°C
Excellent Linearity: 0.15°C max (0 to +70°C)
Wide Operating Temperature Range: -25°C to +105°C
Single Supply Operation: +4V to +30V
Excellent Repeatability and Stability
High Level Output: 1μA/K
Two Terminal Monolithic IC: Temperature In/Current Out
Minimal Self-Heating Errors

CONNECTION DIAGRAM


*PIN 2 CAN BE EITHER ATTACHED OR UNCONNECTED

BOTTOM VIEW
PRODUCT DESCRIPTION

The AD592 is a two terminal monolithic integrated circuit temperature transducer that provides an output current proportional to absolute temperature. For a wide range of supply voltages the transducer acts as a high impedance temperature dependent current source of 1μA/K. Improved design and laser wafer trimming of the IC's thin film resistors allows the AD592 to achieve absolute accuracy levels and nonlinearity errors previously unattainable at a comparable price.

The AD592 can be employed in applications between -25°C and +105°C where conventional temperature sensors (i.e., thermistor, RTD, thermocouple, diode) are currently being used. The inherent low cost of a monolithic integrated circuit in a plastic package, combined with a low total parts count in any given application, make the AD592 the most cost effective temperature transducer currently available. Expensive linearization circuitry, precision voltage references, bridge components, resistance measuring circuitry and cold junction compensation are not required with the AD592.

Typical application areas include; appliance temperature sensing, automotive temperature measurement and control, HVAC (heating/ventilating/air conditioning) system monitoring, industrial temperature control, thermocouple cold junction compensation, board-level electronics temperature diagnostics, temperature readout options in instrumentation, and temperature correction circuitry for precision electronics. Particularly useful in remote sensing applications, the AD592 is immune to voltage drops and voltage noise over long lines due to its high impedance current output. AD592s can easily be multiplexed; the signal current can be switched by a CMOS multiplexer or the supply voltage can be enabled with a tri-state logic gate.

The AD592 is available in three performance grades; the AD592AN, AD592BN and AD592CN. All devices are packaged in a plastic TO-92 case rated from -45°C to +125°C. Performance is specified from -25°C to +105°C. AD592 chips are also available, contact the factory for details.

PRODUCT HIGHLIGHTS

1. With a single supply (4V to 30V) the AD592 offers 0.5°C temperature measurement accuracy.
2. A wide operating temperature range (-25°C to +105°C) and highly linear output make the AD592 an ideal substitute for older, more limited sensor technologies (i.e., thermistors, RTDs, diodes, thermocouples).
3. The AD592 is electrically rugged; supply irregularities and variations or reverse voltages up to 20V will not damage the device.
4. Because the AD592 is a temperature dependent current source, it is immune to voltage noise pickup and IR drops in the signal leads when used remotely.
5. The high output impedance of the AD592 provides greater than 0.5°C/V rejection of supply voltage drift and ripple.
6. Laser wafer trimming and temperature testing insures that AD592 units are easily interchangeable.
7. Initial system accuracy will not degrade significantly over time. The AD592 has proven long term performance and repeatability advantages inherent in integrated circuit design and construction.

AD592—SPECIFICATIONS (typical @ +25°C, V_S = +5V unless otherwise noted)

Model	AD592AN			AD592BN			AD592CN			Units
	Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	
ACCURACY										
Calibration Error @25°C ¹		1.5	2.5		0.7	1.0		0.3	0.5	°C
T _A = 0 to +70°C										°C
Error over Temperature		1.8	3.0		0.8	1.5		0.4	0.8	°C
Nonlinearity ²		0.15	0.35		0.1	0.25		0.05	0.15	°C
T _A = -25 to +105°C										°C
Error over Temperature ³		2.0	3.5		0.9	2.0		0.5	1.0	°C
Nonlinearity ²		0.25	0.5		0.2	0.4		0.1	0.35	°C
OUTPUT CHARACTERISTICS										
Nominal Current Output @25°C (298.2K)		298.2			298.2			298.2		μA
Temperature Coefficient		1			1			1		μA/°C
Repeatability ⁴			0.1			0.1			0.1	°C
Long Term Stability ⁵			0.1			0.1			0.1	°C/month
ABSOLUTE MAXIMUM RATINGS										
Operating Temperature	-25		+105	-25		+105	-25		+105	°C
Package Temperature ⁶	-45		+125	-45		+125	-45		+125	°C
Forward Voltage (+ to -)			44			44			44	V
Reverse Voltage (- to +)			20			20			20	V
Lead Temperature (Soldering 10 sec)			300			300			300	°C
POWER SUPPLY										
Operating Voltage Range	4		30	4		30	4		30	V
Power Supply Rejection										
+4V < V _S < +5V			0.5			0.5			0.5	°C/V
+5V < V _S < +15V			0.2			0.2			0.2	°C/V
+15V < V _S < +30V			0.1			0.1			0.1	°C/V

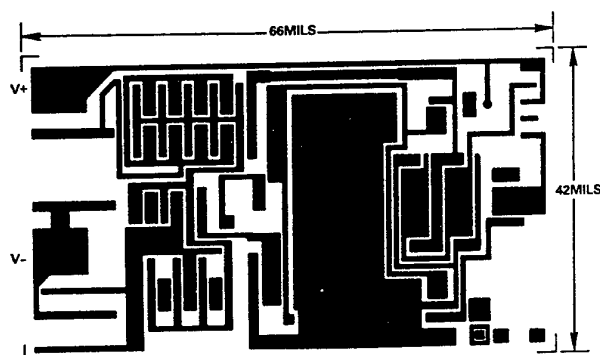
NOTES

- ¹An external calibration trim can be used to zero the error @25°C.
²Defined as the maximum deviation from a mathematically best fit line.
³Parameter tested on all production units at +105°C only. C grade at -25°C also.
⁴Maximum deviation between +25°C readings after a temperature cycle between -45°C and +125°C. Errors of this type are noncumulative.
⁵Operation @125°C, error over time is noncumulative.

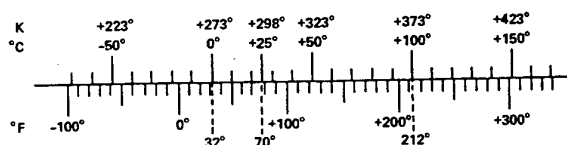
⁶Although performance is not specified beyond the operating temperature range, temperature excursions within the package temperature range will not damage the device. Specifications subject to change without notice.

Specifications shown in boldface are tested on all production units at final electrical test. Results from those tests are used to calculate outgoing quality levels. All min and max specifications are guaranteed, although only those shown in boldface are tested on all production units.

METALIZATION DIAGRAM



THE AD592 IS AVAILABLE IN LASER-TRIMMED CHIP FORM.



TEMPERATURE SCALE CONVERSION EQUATIONS

$$\begin{aligned} ^\circ\text{C} &= \frac{5}{9} (^{\circ}\text{F} - 32) & \text{K} &= ^\circ\text{C} + 273.15 \\ ^\circ\text{F} &= \frac{9}{5} ^\circ\text{C} + 32 & ^\circ\text{R} &= ^\circ\text{F} + 459.7 \end{aligned}$$

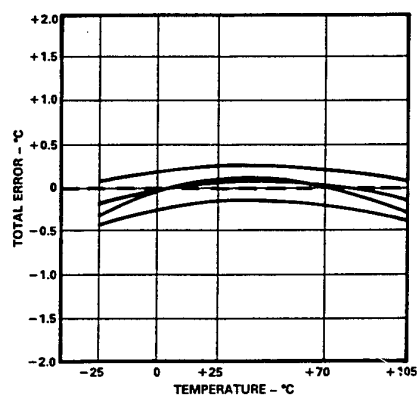
ORDERING GUIDE

Model	Max Cal Error @ 25°C	Max Error -25°C to +105°C	Max Nonlinearity -25°C to +105°C	Package Option*
AD592CN	0.5°C	1.0°C	0.35°C	TO-92
AD592BN	1.0°C	2.0°C	0.4°C	TO-92
AD592AN	2.5°C	3.5°C	0.5°C	TO-92

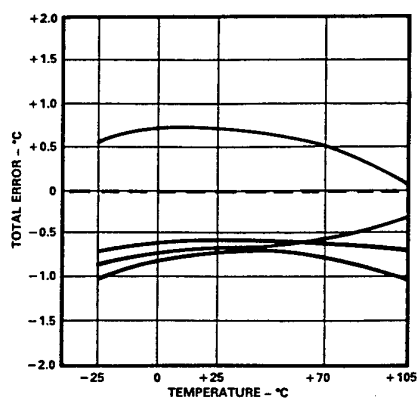
For outline information see Package Information section.

Typical Performance Curves—AD592

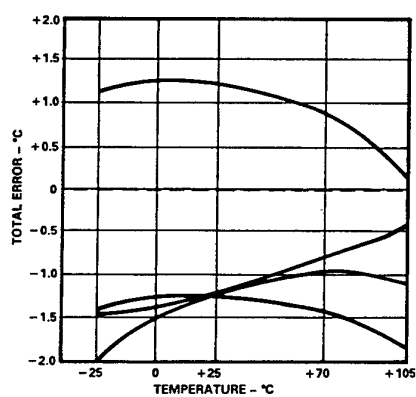
Typical @ $V_s = +5V$



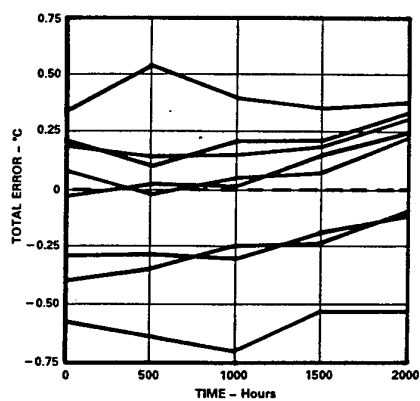
AD592CN Accuracy Over Temperature



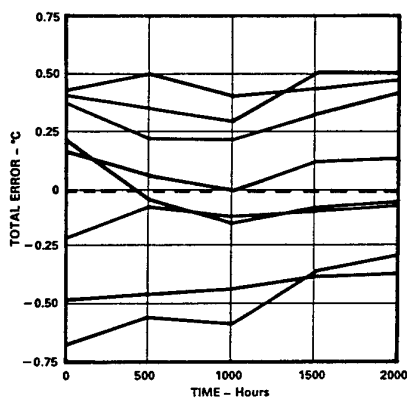
AD592BN Accuracy Over Temperature



AD592AN Accuracy Over Temperature



Long-Term Stability @ 85°C and 85% Relative Humidity



Long-Term Stability @ 125°C

AD592

THEORY OF OPERATION

The AD592 uses a fundamental property of silicon transistors to realize its temperature proportional output. If two identical transistors are operated at a constant ratio of collector current densities, r , then the difference in base-emitter voltages will be $(kT/q)(\ln r)$. Since both k , Boltzman's constant and q , the charge of an electron are constant, the resulting voltage is directly Proportional To Absolute Temperature (PTAT). In the AD592 this difference voltage is converted to a PTAT current by low temperature coefficient thin film resistors. This PTAT current is then used to force the total output current to be proportional to degrees Kelvin. The result is a current source with an output equal to a scale factor times the temperature (K) of the sensor. A typical V-I plot of the circuit at +25°C and the temperature extremes is shown in Figure 1.

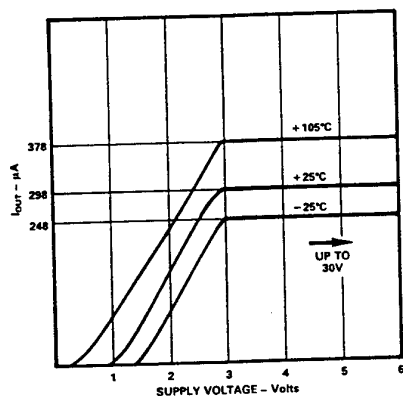


Figure 1. V-I Characteristics

Factory trimming of the scale factor to $1\mu\text{A/K}$ is accomplished at the wafer level by adjusting the AD592's temperature reading so it corresponds to the actual temperature. During laser trimming the IC is at a temperature within a few degrees of 25°C and is powered by a 5V supply. The device is then packaged and automatically temperature tested to specification.

FACTORS AFFECTING AD592 SYSTEM PRECISION

The accuracy limits given on the Specifications page for the AD592 makes it easy to apply in a variety of diverse applications. To calculate a total error budget in a given system it is important to correctly interpret the accuracy specifications, nonlinearity errors, the response of the circuit to supply voltage variations and the effect of the surrounding thermal environment. As with other electronic designs external component selection will have a major effect on accuracy.

CALIBRATION ERROR, ABSOLUTE ACCURACY AND NONLINEARITY SPECIFICATIONS

Three primary limits of error are given for the AD592 such that the correct grade for any given application can easily be chosen for the overall level of accuracy required. They are the calibration accuracy at 25°C, and the error over temperature from 0 to 70°C and -25°C to +105°C. These specifications correspond to the actual error the user would see if the current output of a AD592 were converted to a voltage with a precision resistor. Note that the maximum error at room temperature, over the commercial IC temperature range, or an extended range including the boiling point of water, can be directly read from the Specifications Table. All three error limits are a combination of initial error,

scale factor variation and nonlinearity deviation from the ideal $1\mu\text{A/K}$ output. Figure 2 graphically depicts the guaranteed limits of accuracy for an AD592CN.

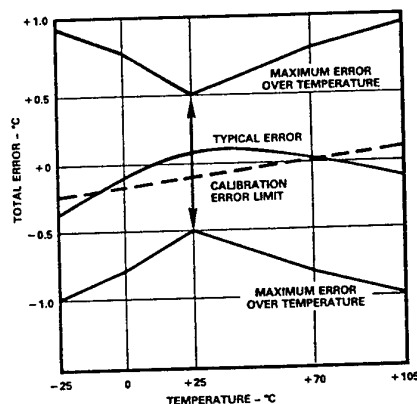


Figure 2. Error Specifications (AD592CN)

The AD592 has a highly linear output in comparison to older technology sensors (i.e., thermistors, RTDs and thermocouples), thus a nonlinearity error specification is separated from the absolute accuracy given over temperature. As a maximum deviation from a best-fit straight line this specification represents the only error which cannot be trimmed out. Figure 3 is a plot of typical AD592CN nonlinearity over the full rated temperature range.

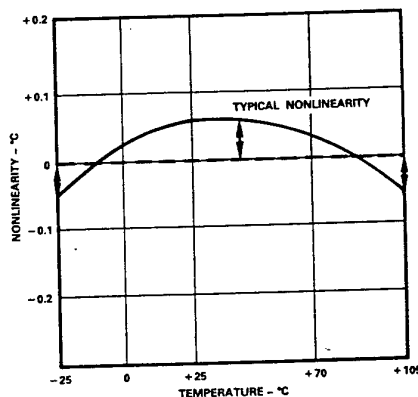


Figure 3. Nonlinearity Error (AD592CN)

TRIMMING FOR HIGHER ACCURACY

Calibration error at 25°C can be removed with a single temperature trim. Figure 4 shows how to adjust the AD592's scale factor in the basic voltage output circuit.

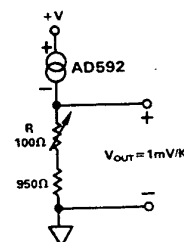


Figure 4. Basic Voltage Output (Single Temperature Trim)

To trim the circuit the temperature must be measured by a reference sensor and the value of R should be adjusted so the output (V_{OUT}) corresponds to 1mV/K . Note that the trim procedure should be implemented as close as possible to the temperature highest accuracy is desired for. In most applications if a single temperature trim is desired it can be implemented where the AD592 current-to-output voltage conversion takes place (e.g., output resistor, offset to an op amp). Figure 5 illustrates the effect on total error when using this technique.

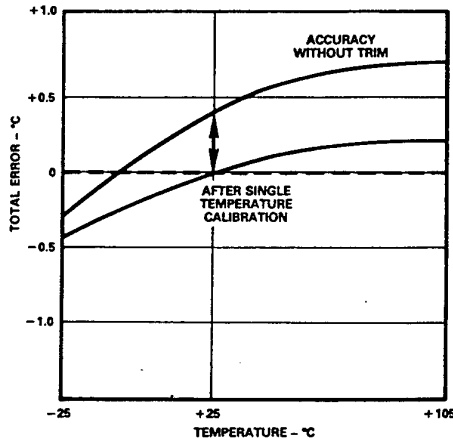


Figure 5. Effect of Scale Factor Trim on Accuracy

If greater accuracy is desired, initial calibration and scale factor errors can be removed by using the AD592 in the circuit of Figure 6.

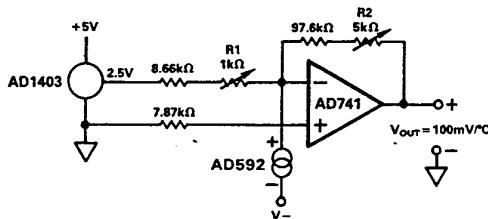


Figure 6. Two Temperature Trim Circuit

With the transducer at 0°C adjustment of R1 for a 0V output nulls the initial calibration error and shifts the output from K to $^\circ\text{C}$. Tweaking the gain of the circuit at an elevated temperature by adjusting R2 trims out scale factor error. The only error remaining over the temperature range being trimmed for is nonlinearity. A typical plot of two trim accuracy is given in Figure 7.

SUPPLY VOLTAGE AND THERMAL ENVIRONMENT EFFECTS

The power supply rejection characteristics of the AD592 minimizes errors due to voltage irregularity, ripple and noise. If a supply is used other than 5V (used in factory trimming), the power supply error can be removed with a single temperature trim. The PTAT nature of the AD592 will remain unchanged. The general insen-

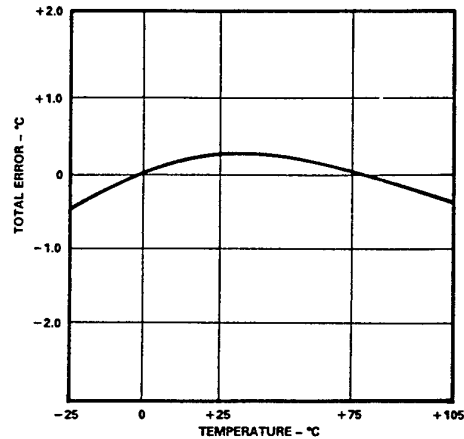


Figure 7. Typical Two Trim Accuracy

sitivity of the output allows the use of lower cost unregulated supplies and means that a series resistance of several hundred ohms (e.g., CMOS multiplexer, meter coil resistance) will not degrade the overall performance.

The thermal environment in which the AD592 is used determines two performance traits: the effect of self-heating on accuracy and the response time of the sensor to rapid changes in temperature. In the first case, a rise in the IC junction temperature above the ambient temperature is a function of two variables; the power consumption level of the circuit and the thermal resistance between the chip and the ambient environment (θ_{JA}). Self-heating error in $^\circ\text{C}$ can be derived by multiplying the power dissipation by θ_{JA} . Because errors of this type can vary widely for surroundings with different heat sinking capacities it is necessary to specify θ_{JA} under several conditions. Table I shows how the magnitude of self-heating error varies relative to the environment. In typical free air applications at 25°C with a 5V supply the magnitude of the error is 0.2°C or less. A common clip-on heat sink will reduce the error by 25% or more in critical high temperature, large supply voltage situations.

Medium	θ_{JA} ($^\circ\text{C}/\text{watt}$)	τ (sec)*
Still Air		
Without Heat Sink	175	60
With Heat Sink	130	55
Moving Air		
Without Heat Sink	60	12
With Heat Sink	40	10
Fluorinert Liquid	35	5
Aluminum Block**	30	2.4

* τ is an average of five time constants (99.3% of final value). In cases where the thermal response is not a simple exponential function, the actual thermal response may be better than indicated.

**With thermal grease.

Table I. Thermal Characteristics

AD592

Response of the AD592 output to abrupt changes in ambient temperature can be modeled by a single time constant τ exponential function. Figure 8 shows typical response time plots for several media of interest.

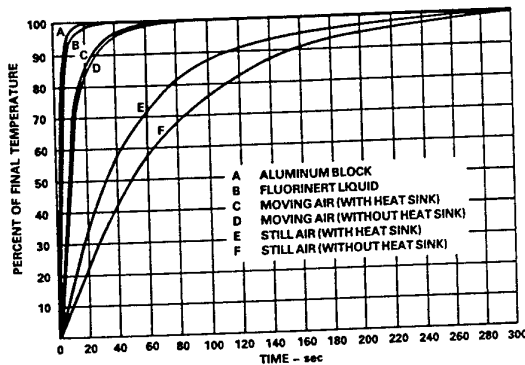


Figure 8. Thermal Response Curves

The time constant, τ , is dependent on θ_{JA} and the thermal capacities of the chip and the package. Table I lists the effective τ (time to reach 63.2% of the final value) for several different media. Copper printed circuit board connections were neglected in the analysis, however, they will sink or conduct heat directly through the AD592's solder dipped Kovar leads. When faster response is required a thermally conductive grease or glue between the AD592 and the surface temperature being measured should be used. In free air applications a clip-on heat sink will decrease output stabilization time by 10–20%.

MOUNTING CONSIDERATIONS

If the AD592 is thermally attached and properly protected, it can be used in any temperature measuring situation where the maximum range of temperatures encountered is between -25°C and $+105^{\circ}\text{C}$. Because plastic IC packaging technology is employed, excessive mechanical stress must be safeguarded against when fastening the device with a clamp or screw-on heat tab. Thermally conductive epoxy or glue is recommended under typical mounting conditions. In wet or corrosive environments any electrically isolated metal or ceramic well can be used to shield the AD592. Condensation at cold temperatures can cause leakage current related errors and should be avoided by sealing the device in nonconductive epoxy paint or dips.

APPLICATIONS

Connecting several AD592 devices in parallel adds the currents through them and produces a reading proportional to the average temperature. Series AD592s will indicate the lowest temperature

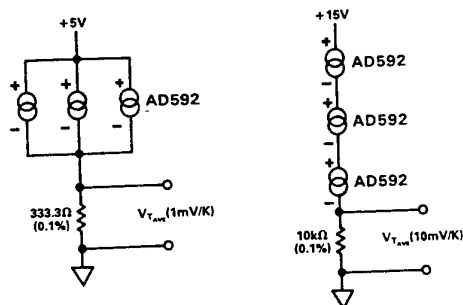


Figure 9. Average and Minimum Temperature Connections

because the coldest device limits the series current flowing through the sensors. Both of these circuits are depicted in Figure 9.

The circuit of Figure 10 demonstrates a method in which a voltage output can be derived in a differential temperature measurement.

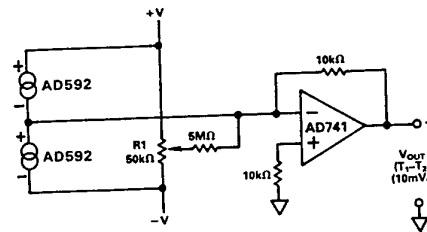


Figure 10. Differential Measurements

R_1 can be used to trim out the inherent offset between the two devices. By increasing the gain resistor ($10\text{k}\Omega$) temperature measurements can be made with higher resolution. If the magnitude of V_+ and V_- is not the same, the difference in power consumption between the two devices can cause a differential self-heating error.

Cold junction compensation (CJC) used in thermocouple signal conditioning can be implemented using an AD592 in the circuit configuration of Figure 11. Expensive simulated ice baths or hard to trim, inaccurate bridge circuits are no longer required.

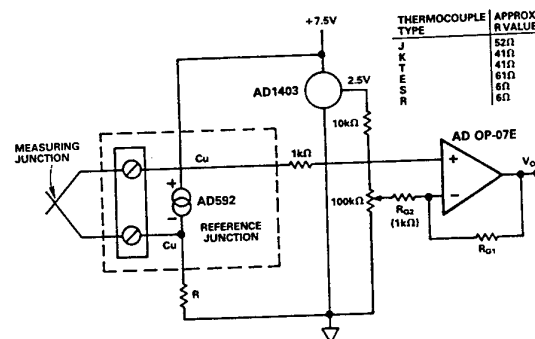


Figure 11. Thermocouple Cold Junction Compensation

The circuit shown can be optimized for any ambient temperature range or thermocouple type by simply selecting the correct value for the scaling resistor – R . The AD592 output ($1\mu\text{A/K}$) times R should approximate the line best fit to the thermocouple curve (slope in $\text{V}/^{\circ}\text{C}$) over the most likely ambient temperature range. Additionally, the output sensitivity can be chosen by selecting the resistors R_{G1} and R_{G2} for the desired noninverting gain. The offset adjustment shown simply references the AD592 to $^{\circ}\text{C}$. Note that the TC's of the reference and the resistors are the primary contributors to error. Temperature rejection of 40 to 1 can be easily achieved using the above technique.

Although the AD592 offers a noise immune current output, it is not compatible with process control/industrial automation current standards. Figure 12 is an example of a temperature to 4–20mA transmitter for use with 40V, $1\text{k}\Omega$ systems.

In this circuit the $1\mu\text{A/K}$ output of the AD592 is amplified to $1\text{mA}/^{\circ}\text{C}$ and offset so that 4mA is equivalent to 17°C and 20mA is equivalent to 33°C . R_t is trimmed for proper reading at an

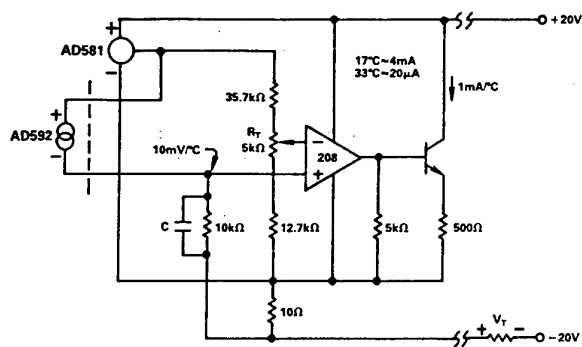


Figure 12. Temperature to 4-20mA Current Transmitter

intermediate reference temperature. With a suitable choice of resistors, any temperature range within the operating limits of the AD592 may be chosen.

Reading temperature with an AD592 in a microprocessor based system can be implemented with the circuit shown in Figure 13.

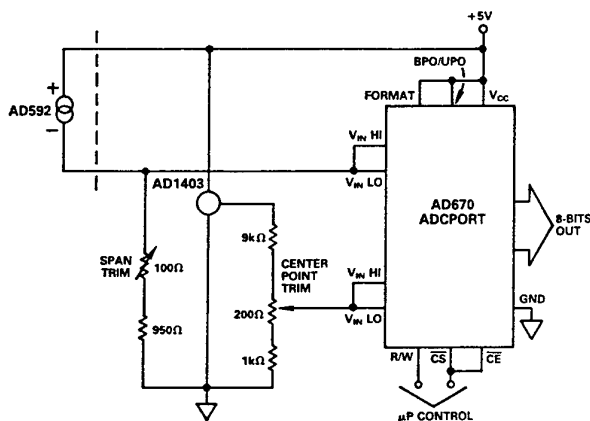


Figure 13. Temperature to Digital Output

By using a differential input A/D converter and choosing the current to voltage conversion resistor correctly any range of temperatures (up to the 130°C span the AD592 is rated for) centered at any point can be measured using a minimal number of components. In this configuration the system will resolve up to 1°C.

A variable temperature controlling thermostat can easily be built using the AD592 in the circuit of Figure 14.

R_{HIGH} and R_{LOW} determine the limits of temperature controlled by the potentiometer R_{SET} . The circuit shown operates over the full temperature range (-25°C to $+105^{\circ}\text{C}$) the AD592 is rated for. The reference maintains a constant set point voltage and insures that approximately 7V appears across the sensor. If it is necessary to guardband for extraneous noise hysteresis can be added by tying a resistor from the output to the ungrounded end of R_{LOW} .

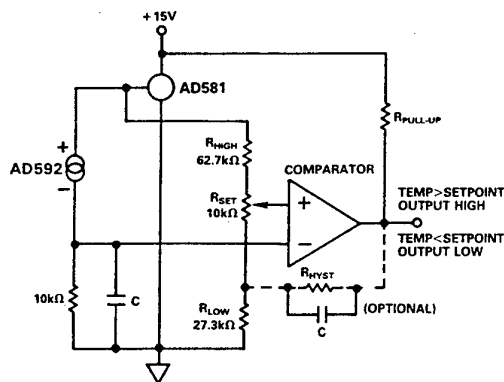


Figure 14. Variable Temperature Thermostat

Multiple remote temperatures can be measured using several AD592s with a CMOS multiplexer or a series of 5V logic gates because of the device's current-mode output and supply-voltage compliance range. The on-resistance of a FET switch or output impedance of a gate will not effect the accuracy, as long as 4V is maintained across the transducer. MUXs and logic driving circuits should be chosen to minimize leakage current related errors. Figure 15 illustrates a locally controlled MUX switching the signal current from several remote AD592s. CMOS or TTL gates can also be used to switch the AD592 supply voltages, with the multiplexed signal being transmitted over a single twisted pair to the load.

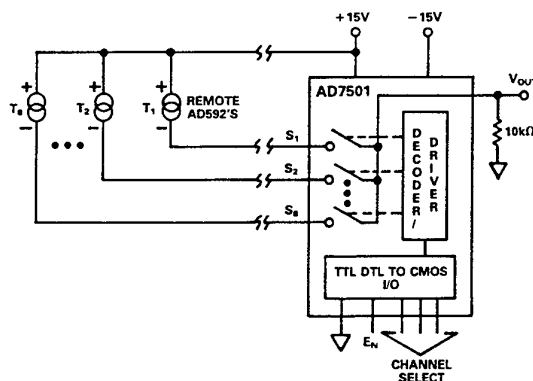


Figure 15. Remote Temperature Multiplexing

To minimize the number of MUXs required when a large number of AD592s are being used, the circuit can be configured in a matrix. That is, a decoder can be used to switch the supply voltage to a column of AD592s while a MUX is used to control which row of sensors are being measured. The maximum number of AD592s which can be used is the product of the number of channels of the decoder and MUX.

AD592

An example circuit controlling 80 AD592s is shown in Figure 16. A 7-bit digital word is all that is required to select one of the sensors. The enable input of the multiplexer turns all the sensors off for minimum dissipation while idling.

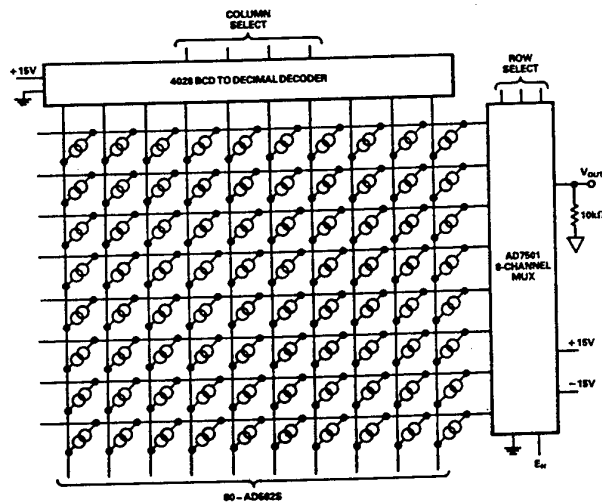


Figure 16. Matrix Multiplexer

To convert the AD592 output to °C or °F a single inexpensive reference and op amp can be used as shown in Figure 17. Although this circuit is similar to the two temperature trim circuit shown in Figure 6, two important differences exist. First, the gain resistor is fixed alleviating the need for an elevated temperature trim. Acceptable accuracy can be achieved by choosing an inexpensive resistor with the correct tolerance. Second, the AD592 calibration error can be trimmed out at a known convenient temperature (i.e., room temperature) with a single pot adjustment. This step is independent of the gain selection.

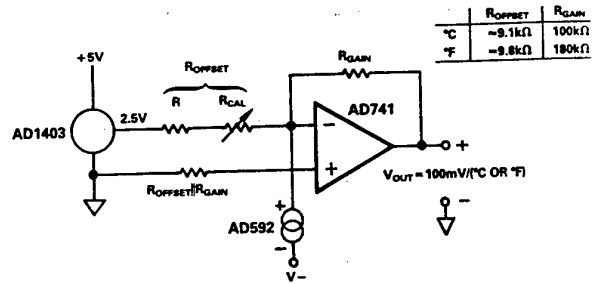
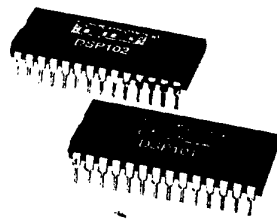


Figure 17. Celsius or Fahrenheit Thermometer

BURR-BROWN INT. s.a.
 18, avenue Dutartre
 B.P. 90
 78152 LE CHESNAY CEDEX
 TEL : (1) 39.54.35.58
 FAX : (1) 39.54.87.03



DSP101
DSP102

DSP-Compatible Sampling Single/Dual ANALOG-TO-DIGITAL CONVERTERS

FEATURES

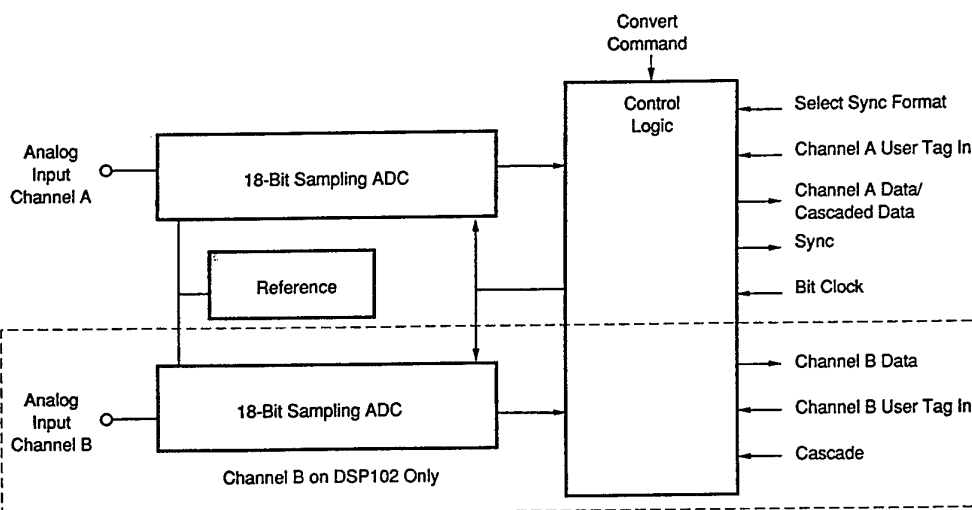
- ZERO-CHIP INTERFACE TO STANDARD DSP ICs: AD, AT&T, MOTOROLA, TI
- SINGLE CHANNEL: DSP101
- DUAL CHANNEL: DSP102
Two Serial Outputs or Cascade to Single 32-Bit Word
- SAMPLING RATE TO 200kHz
- DYNAMIC SPECIFICATIONS:
Signal/(Noise + Distortion) = 88dB;
Spurious-Free Dynamic Range = 94dB;
THD = -91dB
- SERIAL OUTPUT DATA COMPATIBLE WITH 16-, 24-, AND 32-BIT DSP IC FORMATS

DESCRIPTION

The DSP101 and DSP102 are high performance sampling analog-to-digital converters designed for simplicity of use with modern digital signal processing ICs. Both are complete with all interface logic for use directly with DSP ICs, and provide full sampling and conversion at rates up to 200kHz.

The DSP101 offers a single conversion channel, with 18 bits of serial data output, allowing the user to drive 16-bit, 24-bit, or 32-bit DSP ports. The DSP102 offers two complete conversion channels, with either two full 18-bit output ports, or a mode to cascade two 16-bit conversions into a 32-bit port as one word.

Both the DSP101 and DSP102 are packaged in standard, low-cost 28-pin plastic DIP packages. Each is offered in two performance grades to match application requirements.



International Airport Industrial Park • Mailing Address: PO Box 11400 • Tucson, AZ 85734 • Street Address: 6730 S. Tucson Blvd. • Tucson, AZ 85706
 Tel: (602) 746-1111 • Twx: 910-952-1111 • Cable: BBRCORP • Telex: 066-6491 • FAX: (602) 889-1510 • Immediate Product Info: (800) 548-6132

SPECIFICATIONS

ELECTRICAL

$T_A = 0^\circ\text{C}$ to 70°C , $\pm 2.75\text{V}$ input signal, sampling frequency (f_s) = 200kHz, $V_A+ = V_D = +5\text{V}$, $V_A- = -5\text{V}$, 16MHz external clock on OSC1, CLKOUT tied to CLKIN, 8MHz data transfer clock on XCLK, data analysis band-limited to 20kHz, unless otherwise specified.

PARAMETER	CONDITIONS	DSP101JP DSP102JP			DSP101KP DSP102KP			UNITS
		MIN	TYP	MAX	MIN	TYP	MAX	
RESOLUTION				18			*	Bits
ANALOG INPUT								
Voltage Range			$\pm 2.75\text{V}$			*		V
Impedance			1			*		k Ω
Capacitance			20			*		pF
THROUGHPUT SPEED								
Complete Cycle	Acquisition + Conversion			5	*		*	μs
Throughput Rate		200						kHz
AC ACCURACY ⁽¹⁾								
Signal to (Noise + Distortion) Ratio	$f_N = 1\text{kHz}$ $f_N = 1\text{kHz} (-60\text{dB})$	83	86 32		86	88 *		dB ⁽²⁾ dB
Total Harmonic Distortion	$f_N = 25\text{kHz}$		82			*		dB
Spurious-Free Dynamic Range	$f_N = 1\text{kHz}$	89	-90	-86		-91	-89	dB
Signal to Noise Ratio (SNR)	$f_N = 1\text{kHz}$	84	92 88		92 87	94 89		dB dB
DC ACCURACY								
Gain Error				± 5			*	%
Gain Error Mismatch				± 2			*	%
Integral Linearity	DSP102 Channels			Sufficient to meet AC Accuracy Specifications				
Differential Linearity	$\pm 2.75\text{V}$ Input Range			Sufficient to meet AC Accuracy Specifications				
Integral Linearity Error	$\pm 0.7\text{V}$ Input Range		± 0.003			*		%
Differential Linearity Error	$\pm 0.7\text{V}$ Input Range		± 0.002			*		%
No Missing Codes	$\pm 0.7\text{V}$ Input Range		14			*		Bits
Bipolar Zero Error ⁽³⁾			± 2					mV
Bipolar Zero Mismatch ⁽³⁾	DSP102 Channels		± 2			*		mV
Power Supply Sensitivity	$-5.25\text{V} < V_A- < -4.75\text{V}$ $+4.75\text{V} < V_A+, V_D+ < +5.25\text{V}$		-60 -60			*		dB dB
SAMPLING DYNAMICS								
Aperture Delay			30			*		ns
Aperture Jitter			100			*		ps,rms
Transient Response			1			*		μs
Overvoltage Recovery			5			*		μs
DIGITAL INPUTS								
Logic Levels (Except OSC1)								
V_L	$I_L = \pm 10\mu\text{A}$	0		+0.8	*		*	V
V_H	$I_H = \pm 10\mu\text{A}$	+2.4		+5	*		*	V
OSC1 Clock				74HC Compatible				
Frequency				16				MHz
Data Transfer Clock (XCLK)								
Frequency		0.1		12	*		*	MHz
Duty Cycle		40	50	60	*		*	%
Conversion Clock (CLKIN)								
Frequency		0.5		5.33	*		*	MHz
Duty Cycle		25	33	55	*		*	%
DIGITAL OUTPUTS								
Format				Serial; MSB first; 16/18-bit and Cascaded 32-bit Mode				
Coding				Binary Two's Complement				
Logic Levels (Except OSC2)								
V_{OL}	$I_{\text{SINK}} = 4\text{mA}$	0		+0.4	*		*	V
V_{OH}	$I_{\text{SOURCE}} = 4\text{mA}$	+2.4		+5	*		*	V
OSC2				Can only be used to drive crystal oscillator.				
Conversion Clock (CLKOUT)								
Drive Capability		$\pm 2\text{mA}$			*			mA
POWER SUPPLIES								
Rated Voltage								
V_A+		+4.75	+5	+5.25	*	*	*	V
V_A-		-5.25	-5	-4.75	*	*	*	V
V_D		+4.75	+5	+5.25	*	*	*	V
Power Consumption			250	425		*	*	mW
Supply Current	XCLK = OSC1 = 12MHz					*	*	mA
	XCLK = OSC1 = 12MHz		30	45		*	*	mA
			-18	-25		*	*	mA
			5	15		*	*	mA
TEMPERATURE RANGE								
Specification		0		+70	*		*	$^\circ\text{C}$
Storage		-65		+125	*		*	$^\circ\text{C}$

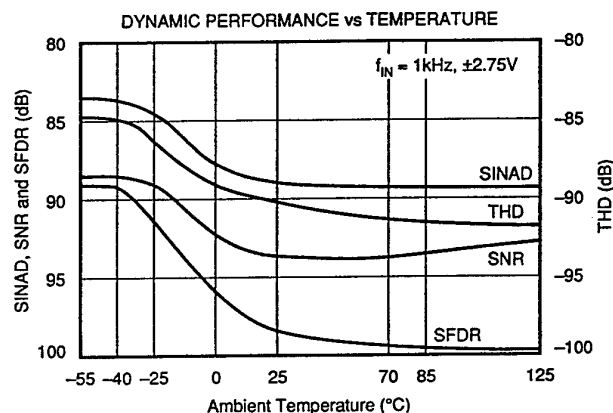
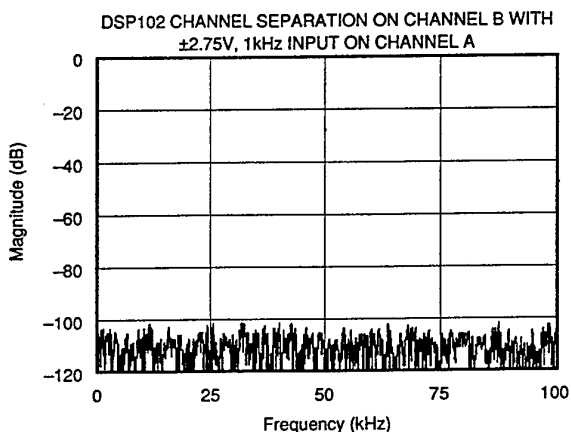
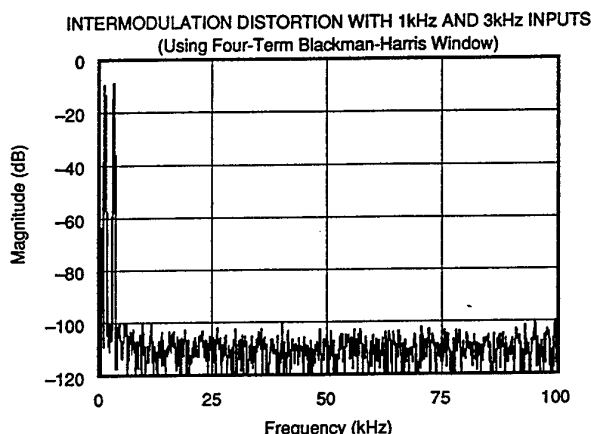
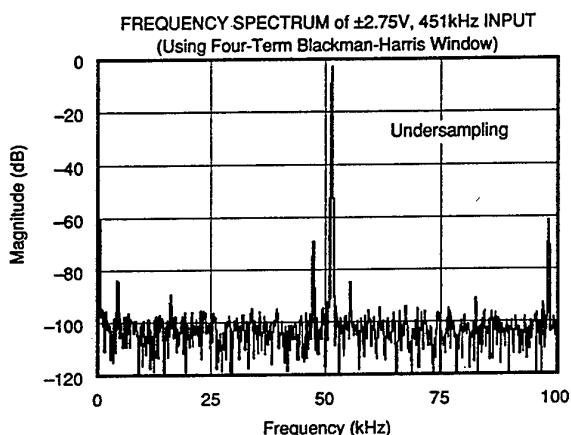
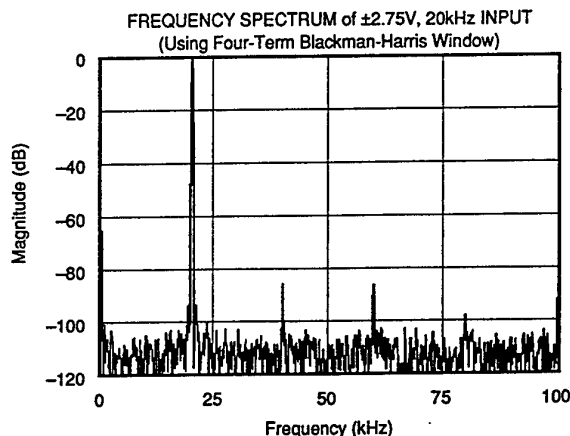
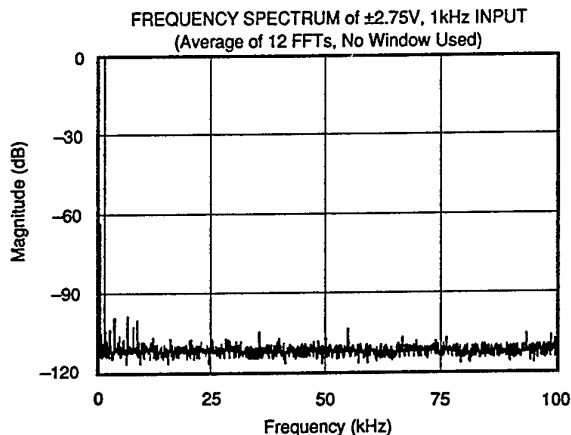
NOTES: (1) All dynamic specifications are based on 2048-point FFTs, using four-term Blackman-Harris window. (2) All specifications in dB are referred to a full-scale input, $\pm 2.75\text{Vp-p}$. (3) Adjustable to zero with external potentiometer.

TYPICAL PERFORMANCE CURVES

At $T_A = +25^\circ\text{C}$, $V_{A+} = V_{D+} = +5\text{V}$, $V_{A-} = V_{D-} = -5\text{V}$, Sampling Frequency $f_s = 200\text{kHz}$; External Clock Input at $\text{OSC1} = 80f_s = 16\text{MHz}$, $\text{XCLK} = 40f_s = 8\text{MHz}$; Using 2048 Point FFT; Data analysis limited to 0 to 20kHz band; Unless otherwise specified.

SINAD means Signal-to-(Noise + Distortion) Ratio.
SNR means Signal-to-Noise Ratio excluding harmonics
thru the 8th.

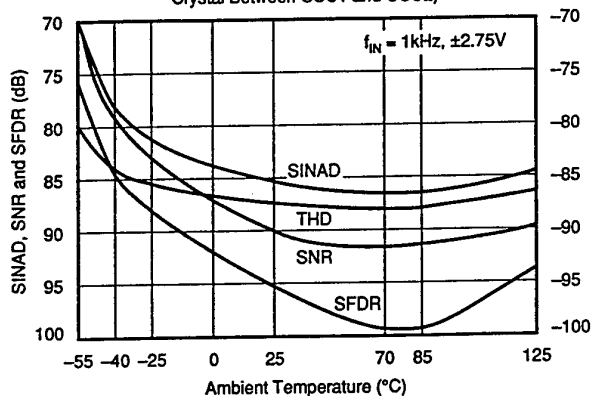
THD means Total Harmonic Distortion thru 8th harmonic.
SFDR means Spurious Free Dynamic Range, including
harmonics.



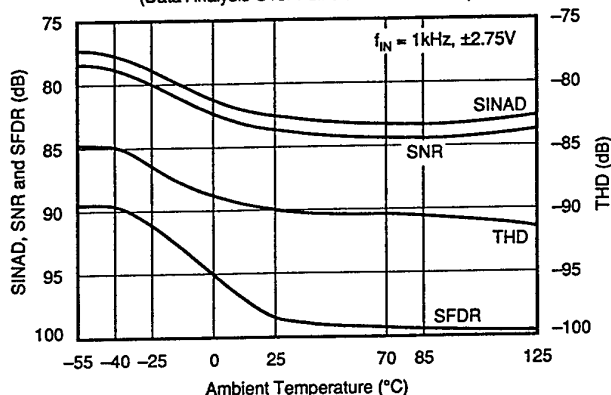
TYPICAL PERFORMANCE CURVES (CONT)

At $T_A = +25^\circ\text{C}$, $V_A = V_D = +5\text{V}$, $V_A = V_D = -5\text{V}$, Sampling Frequency $f_s = 200\text{kHz}$; External Clock Input at $\text{OSC1} = 80f_s = 16\text{MHz}$, $\text{XCLK} = 40f_s = 8\text{MHz}$; Using 2048 Point FFT; Data analysis limited to 0 to 20kHz band; Unless otherwise specified.

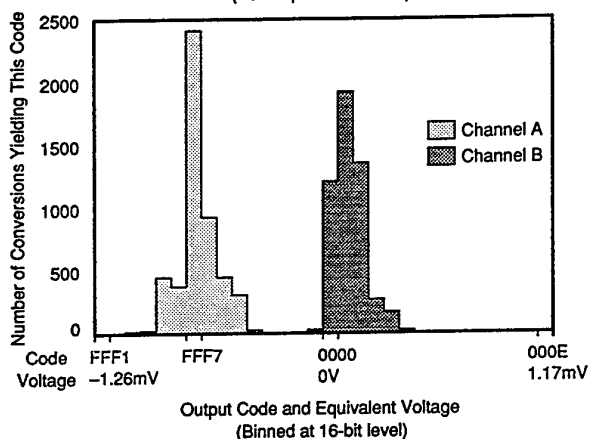
DYNAMIC PERFORMANCE vs TEMPERATURE
($f_s = 180\text{kHz}$ Asynchronous to 12.288MHz
Crystal Between OSC1 and OSC2)



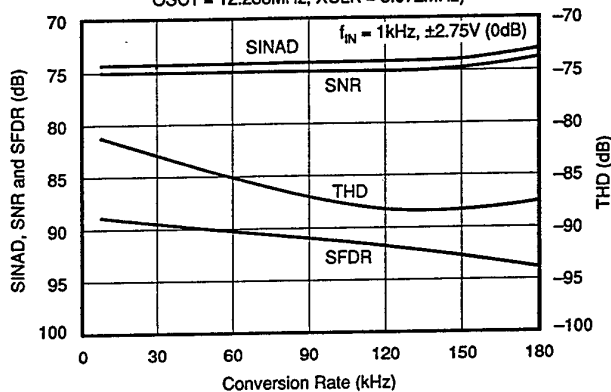
DYNAMIC PERFORMANCE vs TEMPERATURE
(Data Analysis Over Full 0 to 100kHz Band)



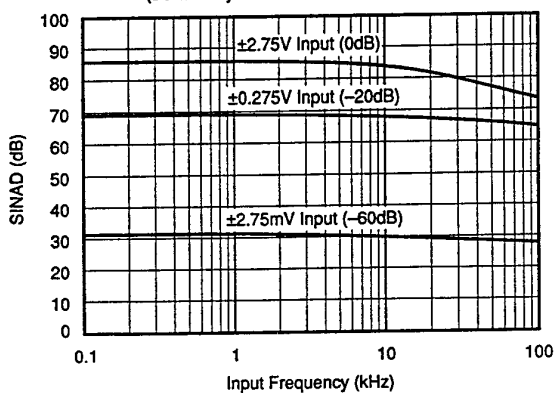
HISTOGRAM OF 5k CONVERSION RESULTS ON DSP102
(Both Inputs Grounded)



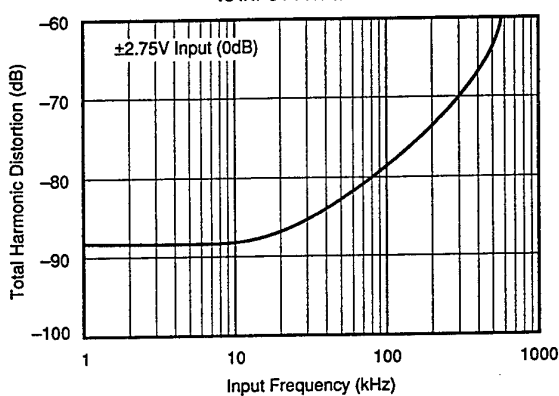
DYNAMIC PERFORMANCE vs CONVERSION RATE
(Data Analysis over Full 0 to $f_s/2$ Band,
 $\text{OSC1} = 12.288\text{MHz}$, $\text{XCLK} = 3.072\text{MHz}$)



SINAD vs INPUT FREQUENCY
(Data Analysis over Full 0 to 100kHz Band)

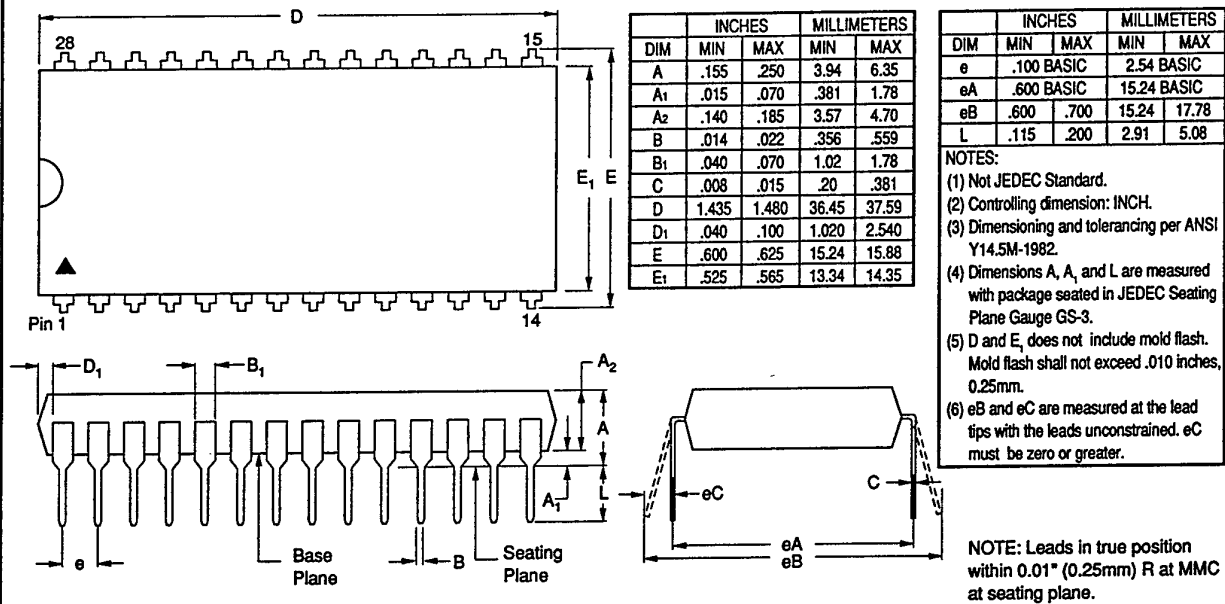


TOTAL HARMONIC DISTORTION vs INPUT FREQUENCY

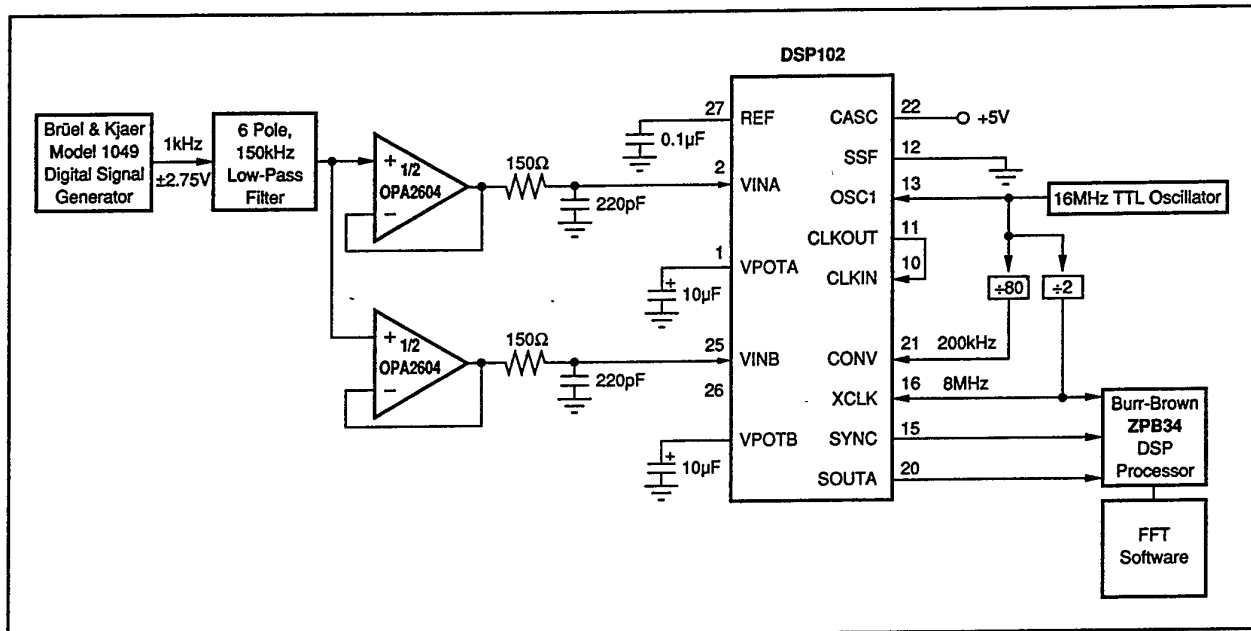


MECHANICAL

P Package — 28-Pin Plastic, Double-Wide DIP



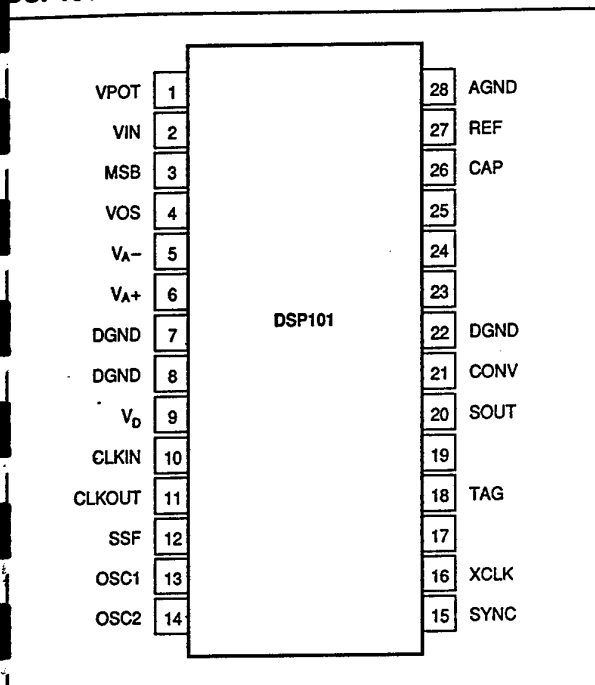
TYPICAL DSP102 FFT SETUP



ABSOLUTE MAXIMUM RATINGS

V_{A+} to Analog Common	+7V
V_{A-} to Analog Common	-7V
V_D to Digital Common	+7V
Analog Common to Digital Common	$\pm 1V$
Control Inputs to Digital Common	-0.5 to $V_D + 0.5V$
Analog Input Voltage	$\pm 5V$
Maximum Junction Temperature	150°C
Internal Power Dissipation	825mW
Lead Temperature (soldering, 10s)	+300°C
Thermal Resistance, θ_{JA} , Plastic DIP	50°C/W

DSP101 PIN CONFIGURATION

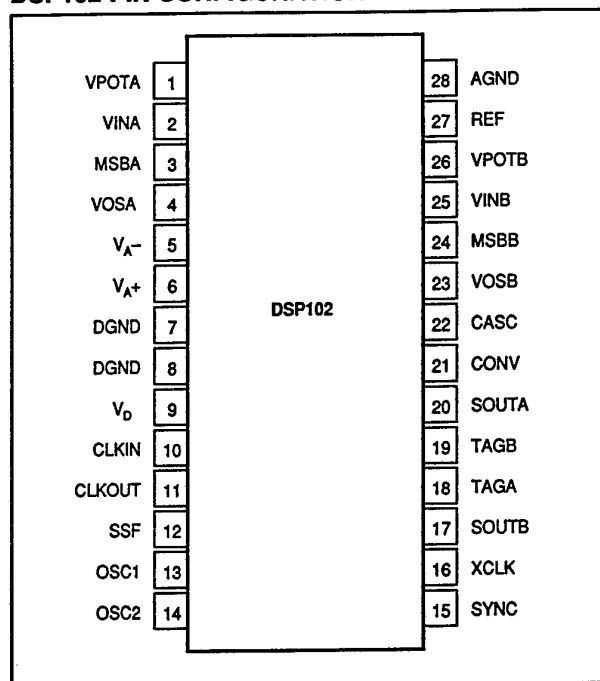


DSP101 PIN ASSIGNMENTS

PIN #	NAME	DESCRIPTION
1	VPOT	Trim Reference Out. 10 μ F Tantalum to AGND.
2	VIN	Analog In.
3	MSB	MSB Adjust In.
4	VOS	VOS Adjust In.
5	V_{A-}	-5V Analog Power.
6	V_{A+}	+5V Analog Power.
7	DGND	Digital Ground.
8	DGND	Digital Ground.
9	V_D	+5V Digital Power.
10	CLKIN	Conversion Clock In.
11	CLKOUT	Conversion Clock Out. Can drive multiple DSP101/DSP102s to synchronize conversion.
12	SSF	Select Synch Format In. If HIGH, SYNC will be active High. If LOW, SYNC will be active Low. See timing diagram (Figure 1).
13	OSC1	Oscillator Point 1 Input/External Clock In. If using external clock, drive with 74HC logic levels. Connect to DGND if not used.
14	OSC2	Oscillator Point 2 Output. Provides drive for crystal oscillator. Make no electrical connection if using external clock.
15	SYNC	Data Synchronization Out. Active High when SSF is HIGH; active Low when SSF is LOW.
16	XCLK	Data Transfer Clock In.
17		No Internal Connection.
18	TAG	User Tag In. Data clocked into this pin is appended to the conversion results on SOUT. See timing diagram (Figure 1).
19		No Internal Connection.
20	SOUT	Serial Data Out. MSB first, Binary Two's Complement format.
21	CONV	Convert Command In. Falling edge puts converter into hold state, initiates conversion, and transmits previous conversion results to DSP IC with appropriate SYNC pulse.
22	DGND	Digital Ground.
23		No Internal Connection.
24		No Internal Connection.
25		No Internal Connection.
26	CAP	Bypass Capacitor. 10 μ F Tantalum to AGND.
27	REF	Reference Bypass. 0.1 μ F Ceramic to AGND.
28	AGND	Analog Ground.

The information provided herein is believed to be reliable; however, BURR-BROWN assumes no responsibility for inaccuracies or omissions. BURR-BROWN assumes no responsibility for the use of this information, and all use of such information shall be entirely at the user's own risk. Prices and specifications are subject to change without notice. No patent rights or licenses to any of the circuits described herein are implied or granted to any third party. BURR-BROWN does not authorize or warrant any BURR-BROWN product for use in life support devices and/or systems.

DSP102 PIN CONFIGURATION

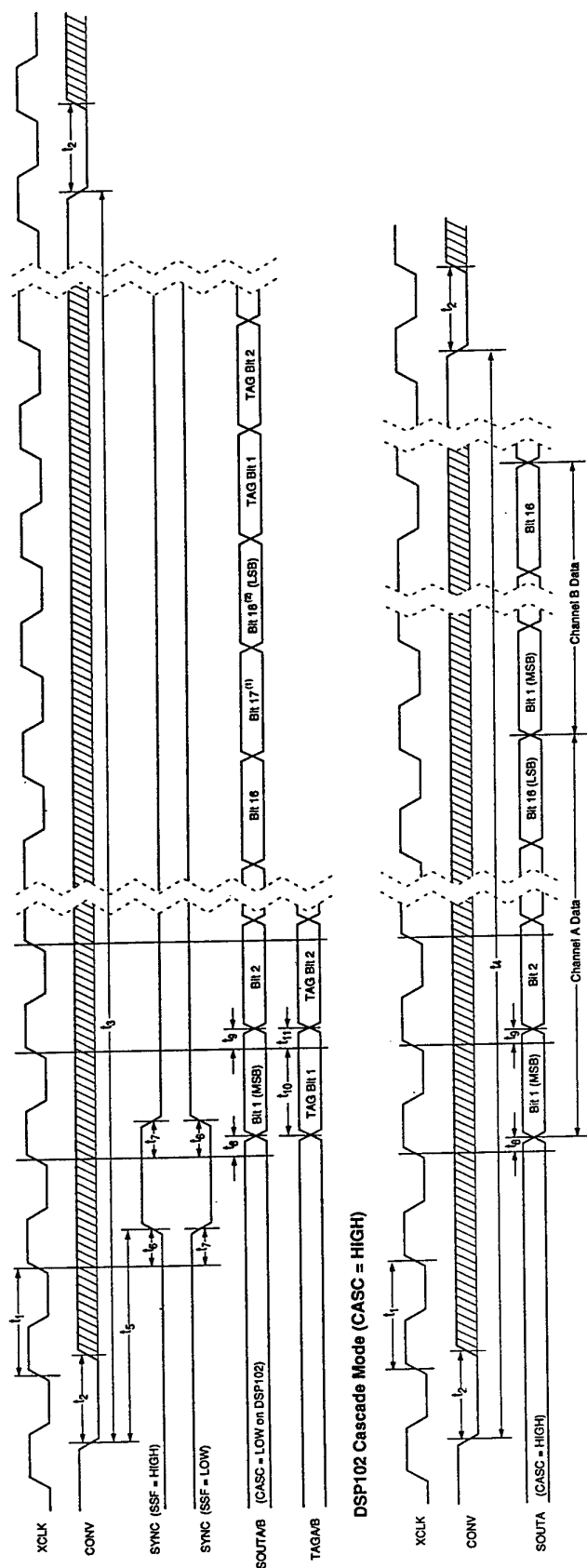


ORDERING INFORMATION

MODEL	NUMBER OF CHANNELS	SIGNAL-TO-(NOISE + DIST.) RATIO dB min
DSP101JP	1	83
DSP101KP	1	86
DSP102JP	2	83
DSP102KP	2	86

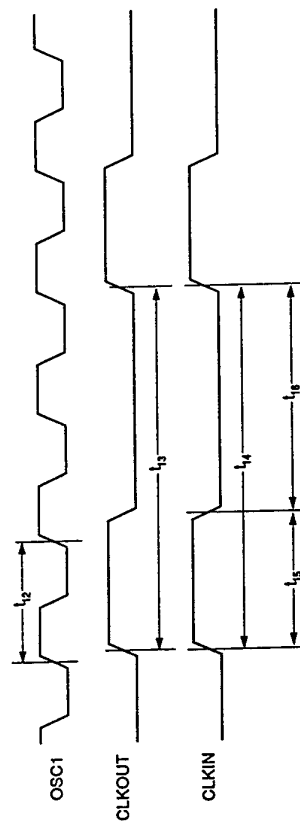
DSP102 PIN ASSIGNMENTS

PIN #	NAME	DESCRIPTION
1	VPOTA	Channel A Trim Reference Out. 10μF Tantalum to AGND.
2	VINA	Channel A Analog In.
3	MSBA	Channel A MSB Adjust In.
4	VOSA	Channel A VOS Adjust In.
5	V _{A-}	-5V Analog Power.
6	V _{A+}	+5V Analog Power.
7	DGND	Digital Ground.
8	DGND	Digital Ground.
9	V _D	+5V Digital Power.
10	CLKIN	Conversion Clock In.
11	CLKOUT	Conversion Clock Out. Can drive multiple DSP101/DSP102s to synchronize conversion.
12	SSF	Select Synch Format In. If HIGH, SYNC will be active High. If LOW, SYNC will be active Low. See timing diagram (Figure 1).
13	OSC1	Oscillator Point 1 Input / External Clock In. If using external clock, drive with 74HC logic levels. Connect to DGND if not used.
14	OSC2	Oscillator Point 2 Output. Provides drive for crystal oscillator. Make no electrical connection if using external clock.
15	SYNC	Data Synchronization Out. Active High when SSF is HIGH; active Low when SSF is LOW.
16	XCLK	Data Transfer Clock In.
17	SOUTB	Channel B Serial Data Out. MSB first, Binary Two's Complement format.
18	TAGA	Channel A User Tag In. Data clocked into this pin is appended to the conversion results of SOUTA. See timing diagram (Figure 1).
19	TAGB	Channel B User Tag In. Data clocked into this pin is appended to the conversion results of SOUTB. See timing diagram (Figure 1).
20	SOUTA	Channel A Serial Data Out. MSB first, Binary Two's Complement format. If CASC is HIGH, 32 bits of data output, with first 16 bits being Channel A data.
21	CONV	Convert Command In. Falling edge puts converter into hold state, initiates conversion, and transmits previous conversion results to DSP IC with appropriate SYNC pulse.
22	CASC	Select Cascade Mode In. If HIGH, DSP102 transmits a 32-bit word on SOUTA, with the first 16 bits being data on Channel A. If LOW, DSP102 transmits data for both channels simultaneously.
23	VOSB	Channel B VOS Adjust In.
24	MSBB	Channel B MSB Adjust In.
25	VINB	Channel B Analog In.
26	VPOTB	Channel B Trim Reference Out. 10μF Tantalum to AGND.
27	REF	Reference Bypass. 0.1μF Ceramic to AGND.
28	AGND	Analog Ground.



SYMBOL	DESCRIPTION ($C_L = 50\text{pF}$)	MIN	MAX	UNITS
t_1	XCLK period, Duty Cycle $50\% \pm 10\%$	83		ns
t_2	Convert Command LOW Time	50		ns
t_3	Convert Period (CASC = LOW on DSP102)	24		t_1
t_4	Convert Period (CASC = HIGH on DSP102)	40		t_1
t_5	SYNC Active Delay after Convert Falling Edge	$t_1 + 40$	2 t_1	ns
t_6	SYNC LOW to HIGH Delay from XCLK Rising		15	ns
t_7	SYNC HIGH to LOW Delay from XCLK Rising		15	ns
t_8	SOUTA/B Data Valid Delay from XCLK Rising		15	ns
t_9	SOUTA/B Data Valid After from XCLK Rising	10		ns
t_{10}	TAGA/B Data Setup before XCLK Rising	20		ns
t_{11}	TAGA/B Data Hold after XCLK Rising	62	0	ns
t_{12}	OSC1 Period, ⁽¹⁾ Duty Cycle $50\% \pm 10\%$	3 t_2	667	ns
t_{13}	CLKOUT Period, Duty Cycle $33\% \pm 10\%$	186	2000	ns
t_{14}	CLKIN Period, Duty Cycle $33\% \pm 20\%$	62	1050	ns
t_{15}	CLKIN HIGH	84	1340	ns
t_{16}	CLKIN LOW			

Conversion Clock Timing ⁽²⁾



NOTES: (1) When using a DSP IC in a 16-bit mode, these data bits will be ignored by the processor. (2) t_{osc1} must be at least 72 times faster than the conversion rate. ($t_3, t_4 \geq 72 t_{osc1}$)

FIGURE 1. DSP101 and DSP102 Timing.



DSP101/102

THEORY OF OPERATION

The DSP101 and DSP102 are sampling analog-to-digital converters optimized for handling dynamic signals. They have complete logic interface circuitry for ease of use with standard digital signal processing ICs, and transmit data words in a serial stream. The successive approximation conversion architecture is combined with an inherently sampling switched capacitor array to provide maximum user flexibility over sampling and conversion timing.

The DSP101 and DSP102 are pipelined internally. When the user gives a convert command at time (t), two actions are initiated. First, the internal sample/holds are switched to the hold state, and a conversion cycle is initiated. At the same time, the DSP101 or DSP102 transmits a synchronization pulse and starts shifting out the conversion results from the previous convert command at (t-1) using the system bit clock. The data from the conversion at time (t) is shifted out of the converter after the next convert command is received.

Both the DSP101 and the DSP102 are 18-bit A/Ds internally. When the DSP IC is programmed to accept 16-bit word lengths, the processor will ignore the last two data bits transmitted from the DSP101 or DSP102. A Cascade Mode on the DSP102 can be invoked to transmit data for both conversion channels over a single serial line as a 32-bit word. In this mode, the first 16 bits of data transmitted after the Sync pulse contain data from channel A, followed by 16 bits of information from channel B, allowing a single 32-bit word to contain data for both channels.

A unique Tag feature allows additional digital data to be appended to the conversion results, so that a single data word contains conversion results plus other signal information, such as gain settings or multiplexer channel settings in front of the converter.

The DSP101 and DSP102 are high-resolution A/D converters complete with sampling capability and on-board references. They can acquire and convert analog signals at up to a 200kHz sampling rate. Both operate from $\pm 5V$ supplies, and have full-scale analog input ranges of $\pm 2.75V$.

BASIC OPERATION

Figure 2 shows the minimum connections required to operate the DSP101. The falling edge of a convert command on pin 21 puts the internal sampling capacitor array into the hold state. The falling edge on pin 21 also starts the process to initiate a conversion and transmit data from the previous conversion, synchronizing both appropriately to the 10MHz clock input on pin 13. Figure 1 shows the timing relationship between the convert command, the output data, and the synchronization pulse.

In this basic system, the 10MHz clock is used both to generate a 3.33MHz conversion clock and as the data transfer bit clock for outputting data. Per Figure 1, there must be at least 72 clock pulses on pin 13 between convert commands, so that this circuit can sample and convert at up to 138kHz.

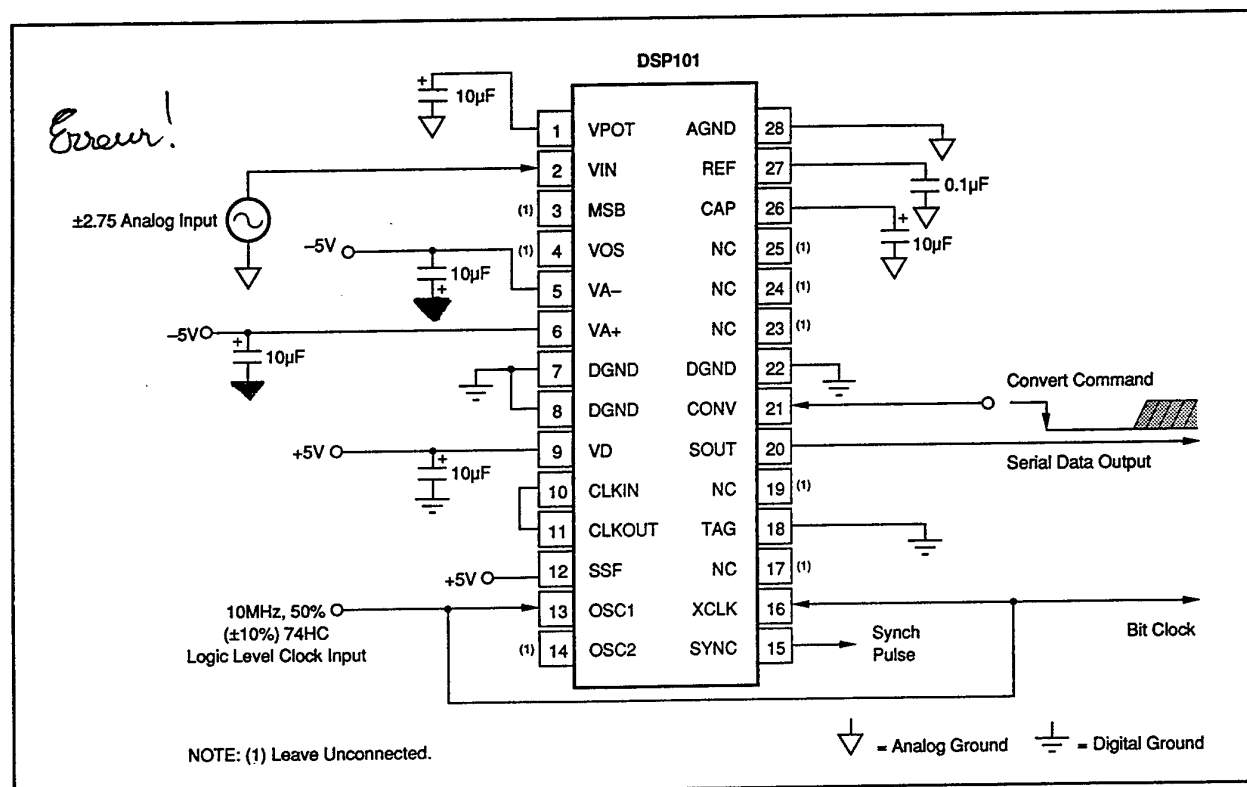


FIGURE 2. DSP101 Basic Operation.

The convert command at pin 21 causes a Sync pulse to be output on pin 15, followed by the data from the previous conversion output on pin 20. The Sync pulse will be HIGH for one bit clock cycle, since pin 12 is tied HIGH. (A LOW Sync pulse will be output on pin 15 if pin 12 is tied LOW.) Data is serially transmitted in an MSB-first data stream, in Binary Two's Complement format. Both the Sync pulse (pin 15) and the data stream (pin 20) are synchronized to the bit clock (at pins 13 and 16), with the timing relationships shown in Figure 1.

After the 18 bits of data from the previous conversion have been transmitted, pin 20 will continue to clock out LOWs until a new convert command restarts the process, since pin 18 (the Tag input) is grounded. If pin 18 is tied HIGH, pin 20 will clock out HIGHs between conversion cycles.

CONVERSION

A falling edge on pin 21 (CONV) puts the internal sampling capacitors in the hold state with minimum aperture jitter, initiates a conversion synchronized to the conversion clock, and outputs the data from the previous conversion with an appropriate Sync pulse. On the DSP102, a single convert command simultaneously samples both channels. The timing relationship between the convert command, Sync and the output data is shown in Figure 1. Both Sync and the output data are synchronized to XCLK, the system bit clock. Following a convert command falling edge, pin 21 must be held LOW at least 50ns.

Convert commands can be sent to the DSP101 and DSP102 completely asynchronous to other clocks in the system. This allows external events to be used to trigger conversions.

From Figure 1, it can be seen that two different clocking conditions must be considered in determining the minimum acceptable time between convert commands. First, there need to be a minimum of 24 XCLK periods between convert commands, to allow internal synchronization and transmission of Sync and the data. (In the Cascade Mode on the DSP102, there need to be at least 40 XCLK periods between convert commands, to allow transmission of the 32-bit data words.) When used with DSP processors programmed for data words longer than 16-bits, the transmission time to the processor may determine the minimum time between convert commands.

The second limitation on convert commands is the requirement that the internal analog-to-digital converter be given enough time to complete a conversion, shift the data to the output register, and acquire a new sample. This condition is met by having a minimum of 24 CLKIN periods between convert commands, or a minimum of 72 clock cycles on OSC1, if it is used to generate the conversion clock (CLKOUT driving CLKIN).

SIGNAL ACQUISITION

After a conversion is completed, the DSP101 or DSP102 will switch back to the sampling mode. With at least 24

CLKIN periods between convert commands, the A/D will have had sufficient time to acquire a new input sample to full rated accuracy.

DATA FORMAT AND INPUT LEVELS

The DSP101 and DSP102 output serial data, MSB first, in Binary Two's Complement format. In the Cascade Mode on the DSP102, the serial data will first contain 16 bits of data for channel A, MSB-first, followed by channel B data, again MSB-first. The analog input levels that generate specific output codes are shown in Table I.

As with all standard A/Ds, the first output transition will occur at an analog input voltage $1/2$ LSB above negative full scale ($-2.75V + 1/2$ LSB) and the last transition will occur $3/2$ LSB below positive full scale ($+2.75V - 3/2$ LSB.) See Figure 3.

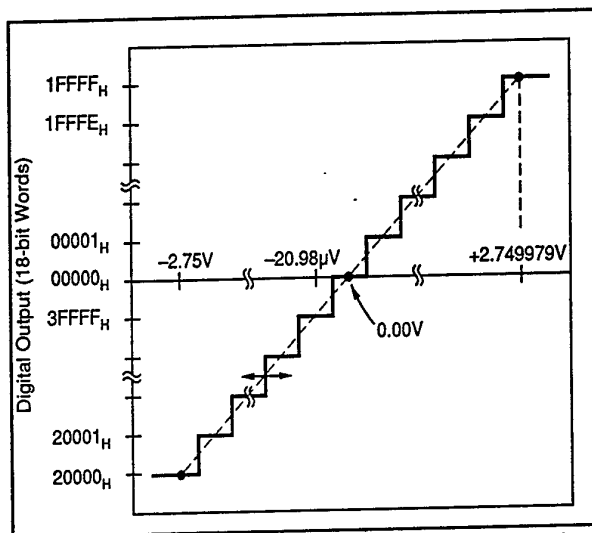


FIGURE 3. Analog Input to Digital Output Diagram.

DESCRIPTION	ANALOG INPUT	DIGITAL OUTPUT (BINARY TWO'S COMPLEMENT)		
		BINARY CODE	16-BIT WORDS (HEX)	18-BIT WORDS (HEX)
Least Significant Bit (LSB = $\frac{5.5V}{2^n}$)				
16-bit Words	84µV			
18-bit Words	21µV			
Input Range	±2.75V			
+ Full Scale (2.75V-1LSB)	+2.749916V +2.749979V	011...111	7FFF	1FFFF
Bipolar Zero (Midscale)	0V	000...000	0000	00000
One LSB below Bipolar Zero	-84µV -21µV	111...111	FFFF	3FFFF
- Full Scale	-2.75V	100...000	8000	20000

TABLE I. Ideal Input Voltage vs Output Code.

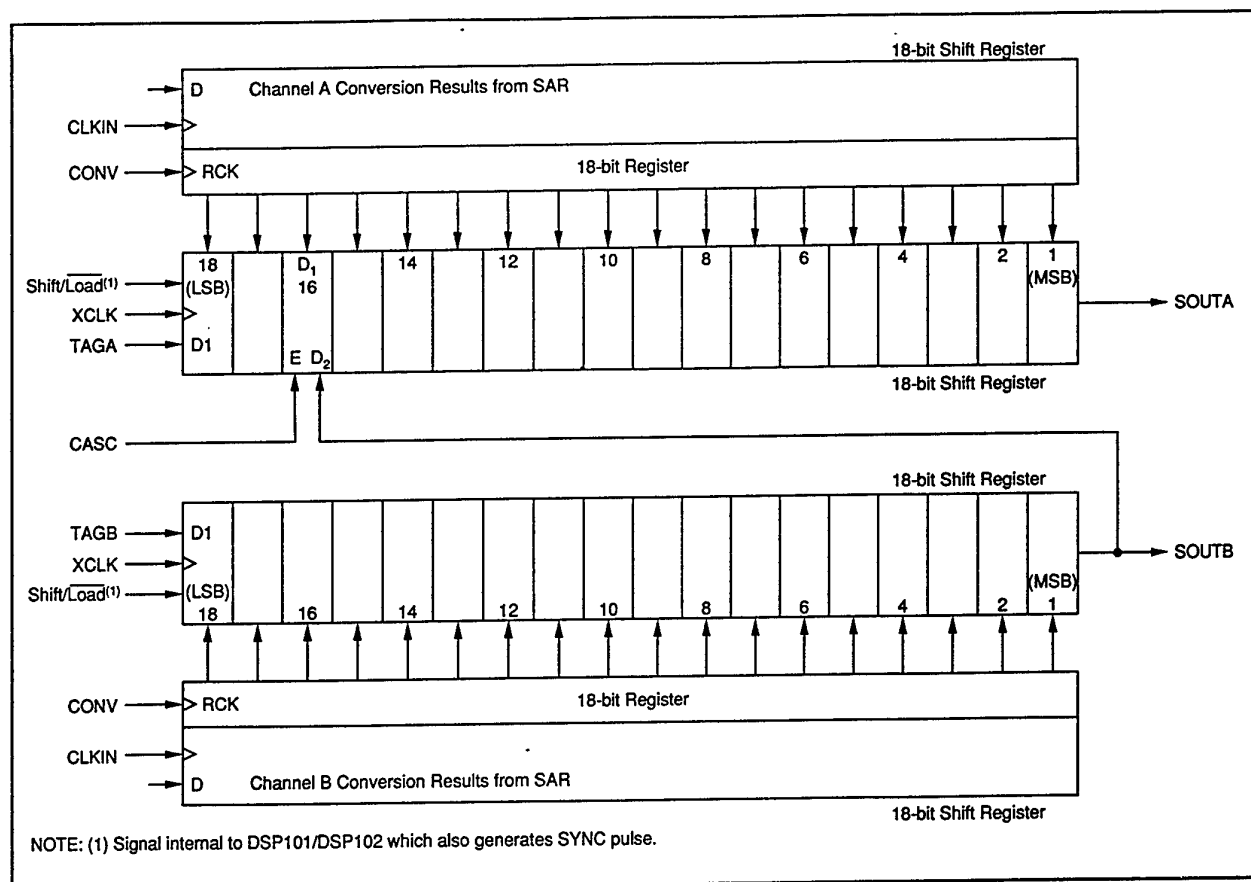


FIGURE 4. Output Structure of DSP102.

DATA TRANSFER

The internal A/Ds generate 18 bits of data, transmitting the data MSB first. When read by a DSP IC programmed to accept 16 bits of data, the first 16 MSB bits of data from the DSP101, or each channel of the DSP102, will be shifted into the processor's input shift register, and the last two least significant bits of data from the A/D will be ignored, although they will still be present on the serial data line. When the DSP processor is programmed to accept words of more than 16-bit length (typically 24-bit or 32-bit), the DSP101 and DSP102 will transmit the full 18-bit conversion results, after which the information input on the TAG input (or TAGA and TAGB on the DSP102) will be appended to the output word. (See Tag Feature below.)

In the Cascade Mode, the DSP102 will first transmit the 16 MSBs from channel A, followed by the full 18-bits from channel B, although DSP processors programmed to accept 32 bits of data will ignore the final two bits of information on Channel B. See the DSP102 Cascade Mode section below for details of the Cascade mode.

DATA SYNCHRONIZATION

A convert command both initiates a conversion and starts the process for transmitting data from the previous conversion. Convert commands can come at any time, completely asynchronous to the conversion clock or the bit clock, and

the conversion clock may also be independent of the bit clock. The DSP101 and DSP102 internally synchronize the output data, Sync pulse, and Tag inputs to the bit clock.

While the convert command, conversion clock and bit clock can be asynchronous, system performance is usually enhanced by synchronizing all of them to a system master clock, whenever the application permits. This minimizes changes in digital loads and currents when the critical S/H transition and A/D bit decisions are occurring. Within the DSP101 and DSP102 themselves, running asynchronous convert commands, conversion clocks and bit clocks typically degrades performance only several dB, as shown in the various typical performance curves, but the system board design can easily have more effect.

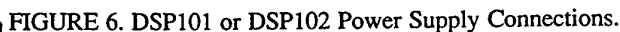
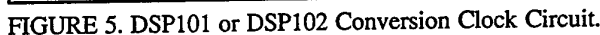
When a convert command is received, the internal logic generates an appropriate Sync pulse, synchronized to XCLK, as shown in Figure 1. The output Sync pulse will be active High or active Low depending on whether a HIGH or a LOW, respectively, is input at SSF (pin 12).

The convert command also causes the conversion results from the previous conversion to be loaded into the output shift register, synchronous to XCLK. Figure 4 shows the operation of the internal data shift registers on the DSP102. The DSP101 is basically similar, but includes only the top of the figure, showing the SOUTA path.

long enough for internal analog circuitry to settle sufficiently between bit decisions to insure rated accuracy. Bit decisions in the A/D are then made on the rising edge of CLKIN.

XCLK is the data transfer clock, or bit clock, for the system, and is an input for the DSP101 or DSP102. This input is TTL- and 74HC-level compatible. The serial data and SYNC outputs are synchronized internally to this clock, with data valid on the rising edge of XCLK, per the timing shown in Figure 1. Data input on pin 18 (TAG) on the DSP101, or on pins 18 and 19 on the DSP102 (TAGA and TAGB), will be clocked into the output shift register on the rising edge of XCLK, as discussed in the Tag Feature section.

The analog-to-digital converter sections in the DSP101 and DSP102 were designed to provide accurate conversions under worst case conditions of supplies, temperatures, etc. In order to achieve a full 200kHz sampling capability, they were designed to use a 33% duty cycle conversion clock (CLKIN on pin 10) as shown in Figure 1. The clock is LOW



When a convert command is received, the DSP101 or DSP102 immediately switches the sampling capacitors to the hold state, and then internally gates the conversion clock to the A/D appropriately. Allowing a minimum of 24 CLKIN pulses between conversions insures that there is sufficient time for complete, accurate conversions, and allows the input sampling capacitor to fully acquire the next sample, regardless of the timing between the convert command and CLKIN.

In most applications, CLKIN (pin 10) can be driven from a 50% duty cycle clock without performance degradation. During characterization of the DSP101 and DSP102, the performance of a number of parts was measured under various conditions with a 4.8MHz, 50% duty cycle input to CLKIN at a full 200kHz conversion rate without noticeable degradation.

OSCILLATOR INPUTS AND CLKOUT

The DSP101 or DSP102 can generate a 33% duty cycle conversion clock output on CLKOUT (pin 11). This is accomplished by dividing by three a clock from either an external 74HC-level clock or from a crystal oscillator. CLKOUT can deliver $\pm 2\text{mA}$, and can be used to drive multiple DSP101 or DSP102 CLKINs. See Figure 1 for the timing relationship between OSC1 and CLKOUT.

To use an external 74HC-level clock, drive the clock into OSC1 (pin 13), and leave OSC2 (pin 14) unconnected.

To use a crystal oscillator to generate the conversion clock, refer to Figure 5. Connect the oscillator between OSC1 and OSC2. OSC2 provides the drive for the crystal oscillator. This pin cannot be used elsewhere in the system.

If CLKOUT is not used, both it and OSC2 should be left unconnected, and OSC1 should be grounded.

TAG FEATURE

Figure 4 shows the implementation of the TAG feature on the DSP101 and DSP102. When a convert command is received, the internal Shift/Load signal loads conversion result data into the output shift register synchronous to XCLK. Between convert commands, the information input on TAG (on the DSP101) or on TAGA and TAGB (on the DSP102) will be clocked into the output shift register on the rising edges of XCLK. Since this is an 18-bit shift register, the data input on the Tag lines will be output on SOUT (DSP101) or SOUTA and SOUTB (DSP102) delayed by 18 bit clocks.

The Tag Feature can be used in various ways. The Tag inputs can be tied HIGH or LOW to differentiate between two converters in a system. As discussed in the Applications section below, the Tag feature can be used to append to the serial output data word information on multiplexer channel address, or other digital data related to the input signal (such as the setting on a programmable gain amplifier.) Another option would be to daisy-chain multiple DSP101 or DSP102 converters, linking the serial output of one to the Tag input of the next. This can simplify the transmission of data from multiple A/Ds over a single optical isolation channel.

DSP102 CASCADE MODE

If pin 22 (CASC) is tied HIGH, the DSP102 will be in the Cascade Mode. In this mode, when a convert command is received, the DSP102 will transmit a 32-bit data word on pin

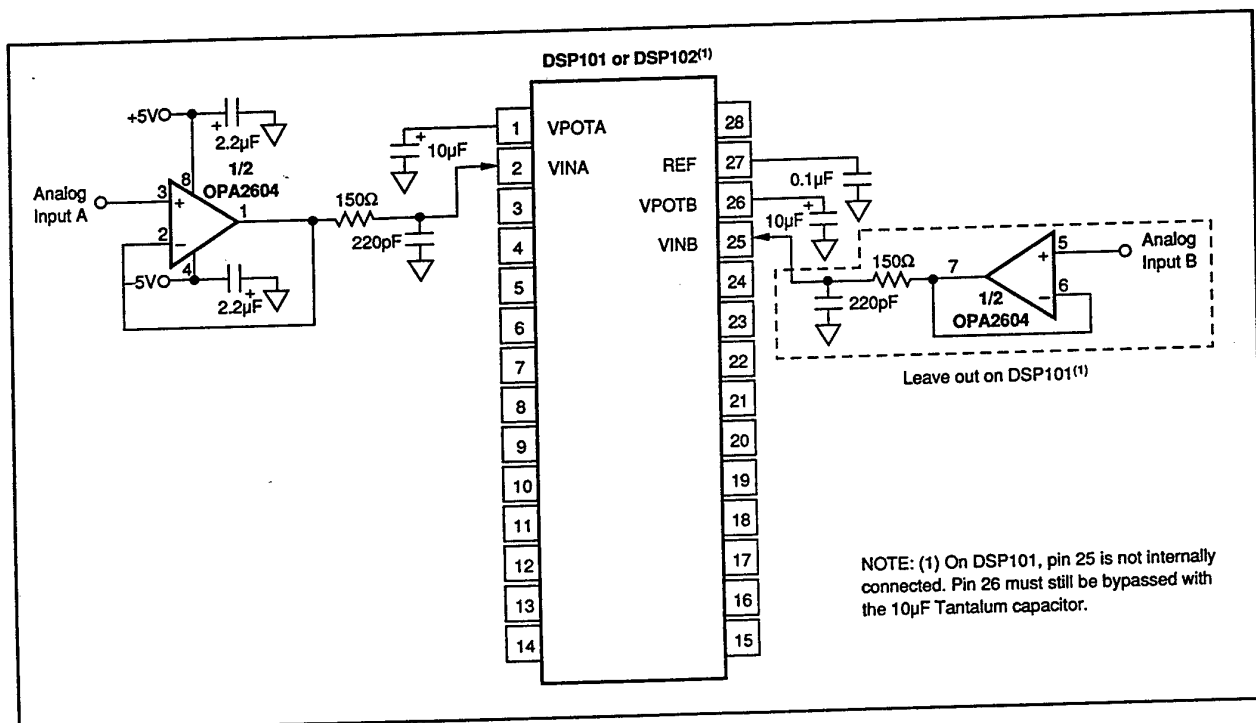


FIGURE 7. DSP101 or DSP102 Input Buffering.

20 (SOUTA) containing data for both input channels in two 16-bit words. Referring to Figure 1, the first 16 bits of data will be the results for channel A, followed by 16 bits of information for channel B. The data will be transferred MSB first. A convert command at time (t) will initiate the transmission of the results of the conversion initiated at time (t - 1).

From the descriptions above of the internal shift registers shown in Figure 4, it can be seen that the DSP102 in the Cascade Mode actually continues to shift out data after the 32nd bit of the data word. The next two bits clocked out will be the last two data bits from the full 18-bit conversion on channel B, after which the information output on SOUTA will be the information clocked into TAGB 35 bit clock cycles earlier.

In the Cascade mode on the DSP102, SOUTB will still output channel B conversion data and tag data as usual.

ANALOG PERFORMANCE

LINEARITY

The DSP101 and DSP102 are optimized for signal processing applications with wide dynamic range requirements. Linearity is trimmed for best performance in the range around 0V, which is critical for handling low amplitude signals. The DSP101 and DSP102 typically have integral and differential non-linearity below $\pm 0.003\%$ in the input range of $\pm 0.7V$, with there being no missing codes at the 14-bit level in this range. Over the full $\pm 2.75V$ input range, the largest non-linearities are centered around the bit #2 transition points at $+1.375V$ and $-1.375V$ levels.

NOISE AND BIPOLAR ZERO ERROR

The equivalent input noise and bipolar zero error of the DSP101 and DSP102 is shown in the typical performance section for both channels on a DSP102. The inputs to both channels were grounded, and the results of 5,000 conversions was recorded. The data shown is binned at the 16-bit level. The noise results from all sources in the circuit, including clocks, reference noise, etc.

In a theoretically ideal converter with no offset and no noise, the results of all 5,000 conversion for each channel would lie in the bin corresponding to bipolar zero, code 0000. The typical DSP101 or DSP102 will have offset errors in the range of 1 to 2mV, and the two channels on the DSP102 will be matched closer than 2mV. The DSP102 shown in the typical performance section has the worst offset, $-0.8mV$, on channel A, with channel B being less than 1mV different, and the three sigma noise on either channel being less than $250\mu V$.

INPUT BANDWIDTH

From the typical performance curves, it can be seen that there is very little degradation in Signal-to-(Noise + Distortion) for input signals up to 100kHz. The wideband sampling input typically maintains a 60dB Signal-to-(Noise + Distortion) Ratio undersampling 500kHz input signals.

LAYOUT CONSIDERATIONS

Because of the high resolution, linearity and speed of the DSP101 and DSP102, system design problems such as ground path resistance, contact resistance and power supply quality become very important.

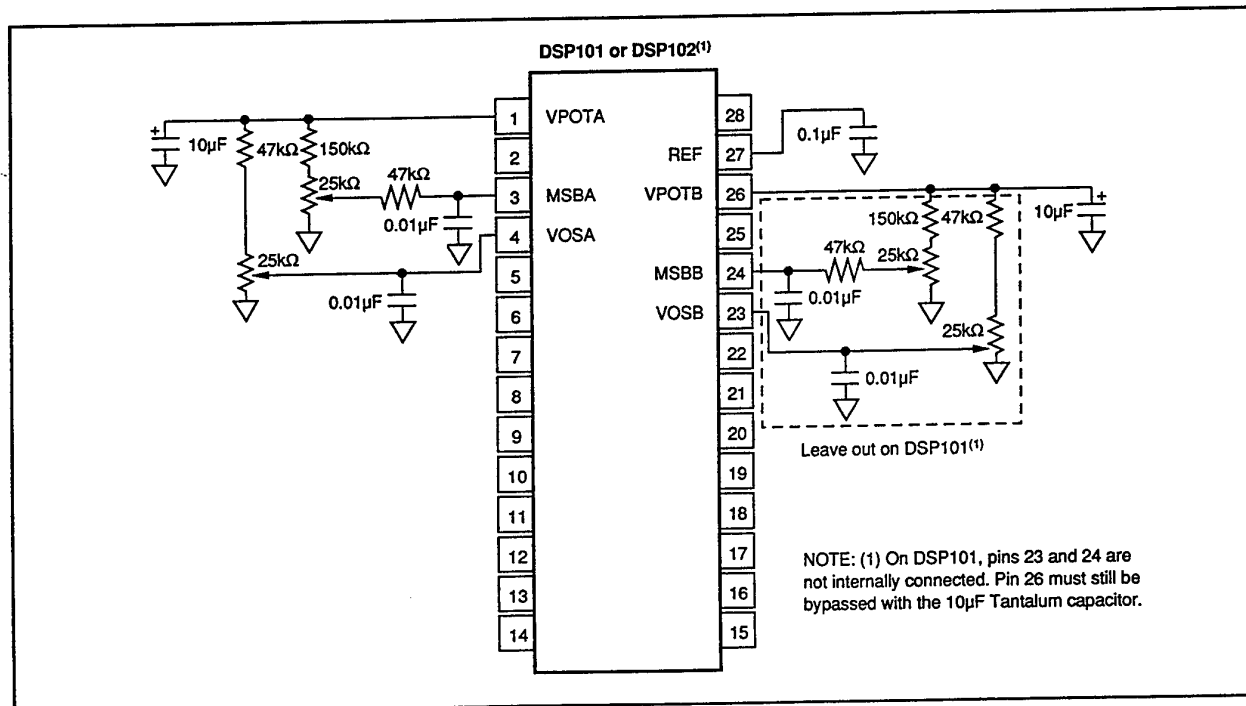


FIGURE 8. DSP101 or DSP102 Optional MSB and Offset Adjust.

Short traces on the board, and bypass capacitors as close as possible to the A/D, will further improve dynamic performance.

To achieve the maximum performance from the DSP101 or DSP102, care should be taken to minimize the effect of changes in current flowing in the system grounds, particularly while bit decisions are being made in the successive approximation converter's comparator. Pin 28 (AGND) on both the DSP101 and the DSP102 is the most critical, and care should be taken to make this pin as close as possible to the same potential as the system analog ground.

Whenever possible, it is strongly recommended that separate analog and digital ground planes be used. With an LSB level of $84\mu\text{V}$ at the 16-bit level, and one-quarter of that at the 18-bit level, the currents switched in a typical DSP system can easily corrupt the accuracy of the A/Ds unless great care is taken to analyze and design for current flows.

All of the supplies should be decoupled to the appropriate grounds using tantalum capacitors in parallel with ceramic capacitors, as shown in Figure 6. For optimum performance of any high resolution A/D, all of the supplies should be as clean as possible. If separate digital and analog supplies are available in a system, care should be taken to insure that the difference between the analog and the digital supplies is not more than 0.5V for more than a few hundred milliseconds, as may occur at power-on.

To avoid introducing distortion, the DSP101 and DSP102 analog inputs must be driven by a source with low impedance over the input bandwidth needed in the application. Op amps such as the NE5532 or Burr-Brown's OPA2604 work well over audio bandwidths. Figure 7 shows an appropriate input driver circuit. The 150 Ω and 220pF shown on the input help reduce the dynamic load on the input signal conditioning amp in front of the A/D, since all switched capacitor array architectures exhibit fast changes in input current load as the input sampling switch is opened and closed. These dynamic changes in the load can affect any signal conditioning circuit at the input. Other R and C combinations can be



used, but the resistor should not exceed 200Ω , or the output settling time of the signal conditioning amplifier may be too long.

EXTERNAL ADJUSTMENTS

All of the specifications for the DSP101 and DSP102, plus the typical performance curves, are based on the performance of these A/Ds without external trims. In most applications, external trims are not required.

OFFSET ADJUST

Where required by specific applications, offsets can be adjusted using Figure 8. When not adjusted, VOS (pin 4) on the DSP101, and VOSA (pin 4) and VOSB (pin 23) on the DSP102, should be left open. If these pins are connected to traces on the board, they should be bypassed to ground with $0.01\mu\text{F}$ capacitors, as close as possible to the A/D.

To trim offset, one alternative is to ground the analog input while converting continually. Then adjust the trimpot (on VOS for the DSP101, on VOSA and VOSB for the DSP102) until the output code is toggling between the codes FFFF and 0000 (Hex) at the 16-bit level (3FFFF and 00000 at the 18-bit level.) This will center the offset at $1/2$ LSB below 0V,

which is respectively $-42\mu\text{V}$ or $-10\mu\text{V}$ at the 16- and 18-bit levels.

The offset can also be adjusted by providing a sine wave to the A/D input. Using FFT, or even simple averaging of several thousand conversion results at a time, the trimpots can be adjusted until there is no DC offset of the signal.

Grounding the input, or providing the sine wave, as far in front of the A/D as possible allows offset from intervening signal conditioning components to be also corrected by this procedure.

MSB ADJUST

In most applications, adjustment of the Most Significant Bit weight will not be required. When not adjusted, MSB (pin 3) on the DSP101, and MSBA (pin 3) and MSBB (pin 24) on the DSP102, should be left open. If these pins are connected to traces on the board, they should be bypassed to ground with $0.01\mu\text{F}$ capacitors, as close as possible to the A/D.

MSB (pin 3) on the DSP101, and MSBA (pin 3) and MSBB (pin 24) on the DSP102, are internally connected to a resistor divider network that is used to laser-trim the weight of the MSB capacitor in the CDAC. These pins are nomi-

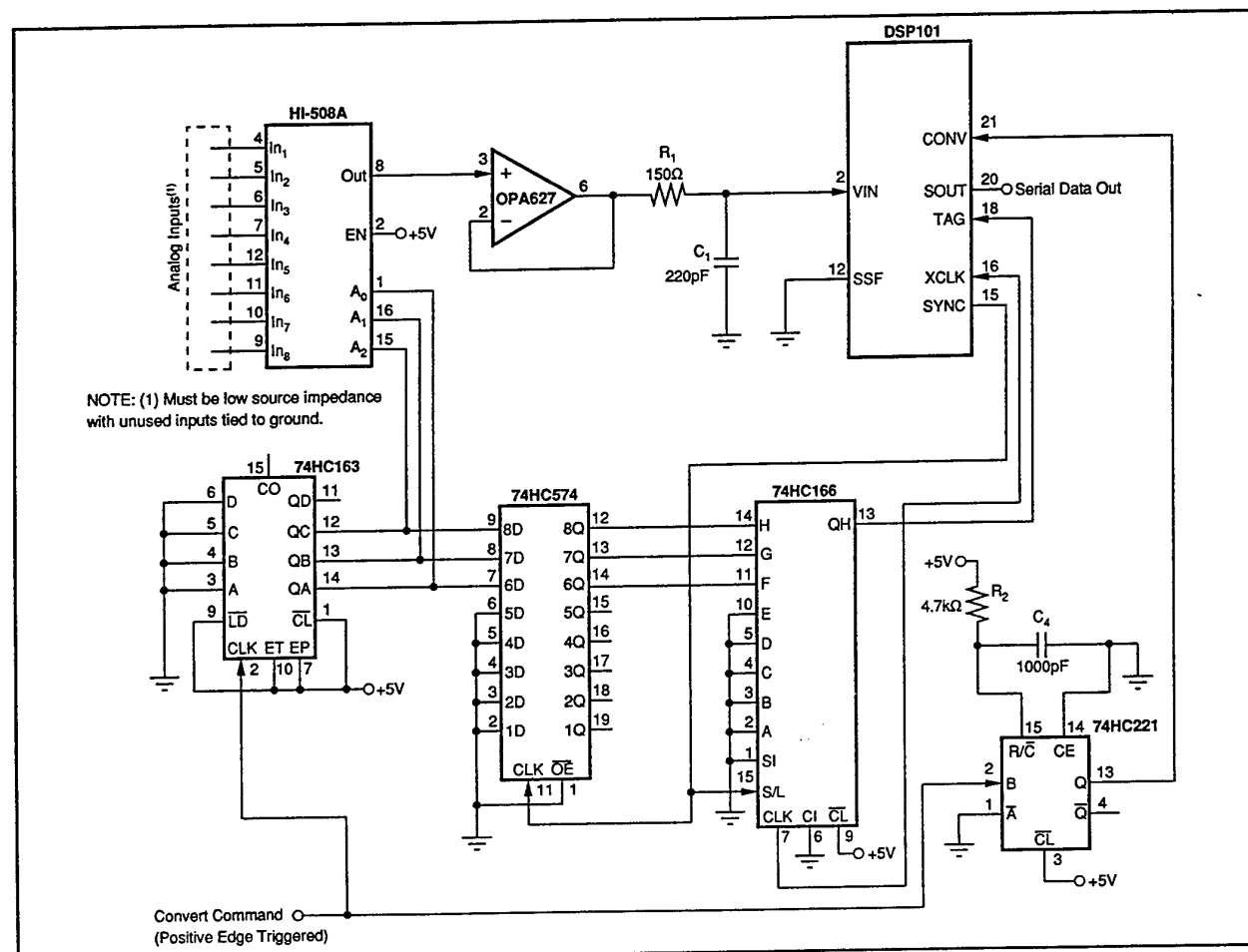


FIGURE 10. A Complete Eight-Channel Analog Input System Using the DSP202 and the HI-508A.

nally at +100mV after laser-trimming during manufacturing. They can handle external inputs up to about one diode drop below ground (-0.6V) before internal clamping circuitry is triggered.

Figure 8 shows an appropriate circuit for adjusting the weight of the most significant bit to minimize differential non-linearity at the critical major-carry transition. To adjust, provide a small amplitude sine wave to the selected A/D input pin while converting continually, and adjust for maximum Signal-to-(Noise + Distortion) ratio, using appropriate signal analysis software.

GAIN ADJUST

If circuit gain needs to be adjusted in hardware, rather than in system software, appropriate trimpots should be included in the analog signal conditioning section in front of the DSP101 or DSP102. No specific gain adjust circuitry is included in the parts.

APPLICATIONS

INTERFACING DSP101 TO PARALLEL PORTS

Figure 9 shows a circuit for converting the serial output data from the DSP101 into 16 bits of parallel data, within the timing constraints of the serial bit-stream from the DSP101. In many applications, this circuit can be easily incorporated into gate arrays or other programmed logic circuits already used in the system, since the extra gate count is not high.

This circuit adds an additional pipeline delay to the conversion data, so that the parallel data from a conversion at time

(t) is valid one conversion cycle plus 17 XCLK clocks later (at t+1 plus 17 times XCLK). A convert command at time (t+1) generates a Sync and begins transmitting serial data from SOUT. The serial data is shifted into the 74HC594 shift registers, and Sync is shifted through the 74HC164 shift registers. The Q1 output of the 74HC74 dual D-type flip-flops clocks the conversion data into the output register of the 74HC594s, and triggers a data valid signal on its Q2 output. The user can then read the data at any time before the next conversion is started, and the Read signal will reset the data valid output from Q2.

In many systems, galvanic isolation of signals is required. Using opto-couplers on the serial data lines in Figure 9 allows a fully isolated system to be built using a DSP101 and only three couplers across the barrier (for serial data, XCLK and SYNC.)

MULTIPLEXING INPUTS TO THE DSP101

Figure 10 shows a complete circuit for sequentially scanning eight analog input channels with a single DSP101, and using the Tag feature on the DSP101 to append the multiplexer channel address to the serial output conversion results.

The circuit in Figure 10 includes the required digital logic and timing logic. The 74HC163 counter provides the scan sequence to the Burr-Brown HI-508A analog multiplexer. In order to allow the HI-508A enough time to switch to the next channel and settle before the DSP101 begins a conversion, a 74HC221 one-shot introduces a 3µs delay for the DSP101 convert command input.

The Burr-Brown OPA627 provides a low impedance source for the DSP101, buffering it from the output impedance of

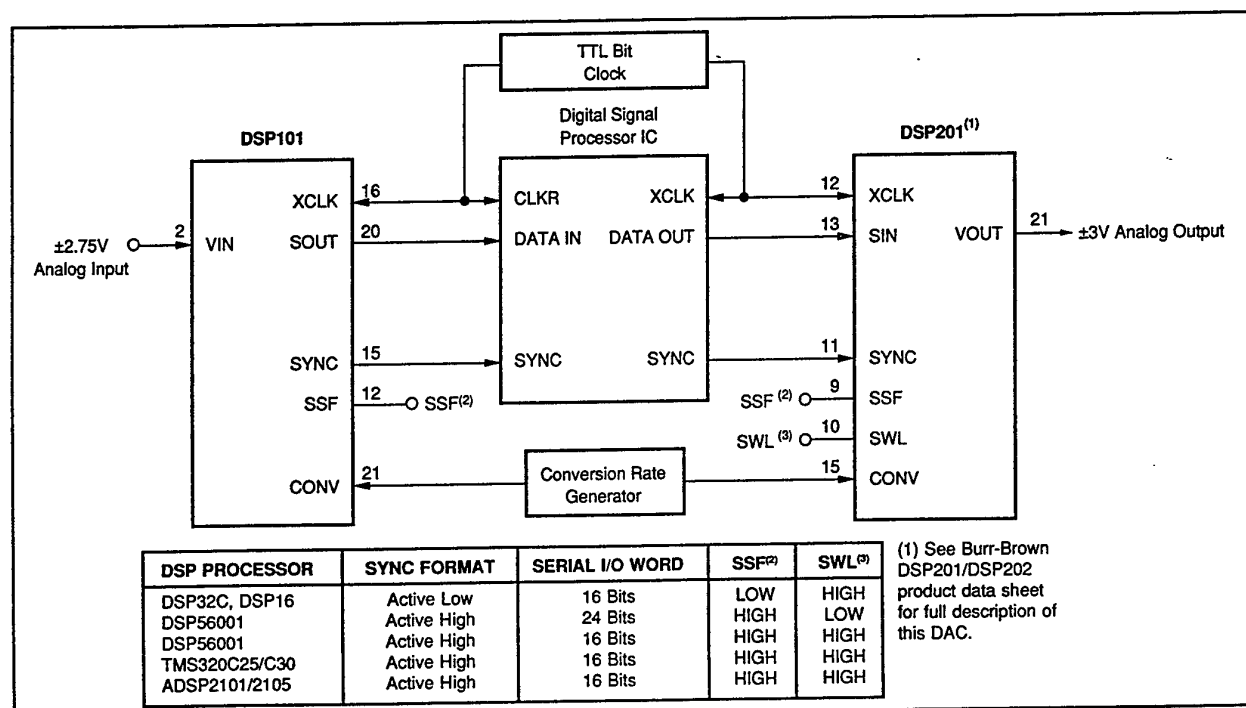


FIGURE 11. Analog Input and Analog Output System.

the multiplexer. This unity-gain buffer minimizes distortion, taking full advantage of the resolution and bandwidth of the DSP101.

The 74HC574D register delays the multiplexer address data by one conversion before appending the channel data to the serial conversion results from the DSP101. This attaches the channel address to the correct conversion results. Since the channel scanning shown in Figure 10 is sequential, this delay latch could be left out and software could recognize that the time (t) conversion results have the MUX address from the time (t-1) conversion appended. However, for systems using non-sequential scan lists, this delay latch is essential to maintain the conversion data and channel address integrity.

The 74HC166 synchronous loading shift register loads the channel address tag data into the shift register on the rising edge of the bit clock, in conjunction with the Sync output of the DSP101. The channel address tag data is then clocked into the DSP101 Tag input (pin 18) by the bit clock, while the conversion data is clocked out the other end of the

DSP101 shift register (discussed in another section of this data sheet.)

Figure 10 was developed and tested using a Burr-Brown ZPB34 DSP board, which contains an AT&T DSP32C, so that the SYNC output is programmed to be active LOW. The circuit needs to be modified for DSP processors from ADI, TI, and Motorola, which use active HIGH Sync pulses. For these processors, tie SSF (pin 12) on the DSP101 HIGH, and use a 74HC04 hex inverter to invert the Sync signal to the 74HC574 and 74HC166.

The same basic circuit can be duplicated to drive two channels in a DSP102, or can be easily modified for more or less than eight channels of analog input.

USING DSP101 AND DSP102 WITH TEXAS INSTRUMENTS DSP ICS

Figures 11 thru 17 show various ways to use the DSP101 and DSP102 with DSP ICs from the Texas Instruments TMS320Cxx series. For simplicity, all of these circuits are

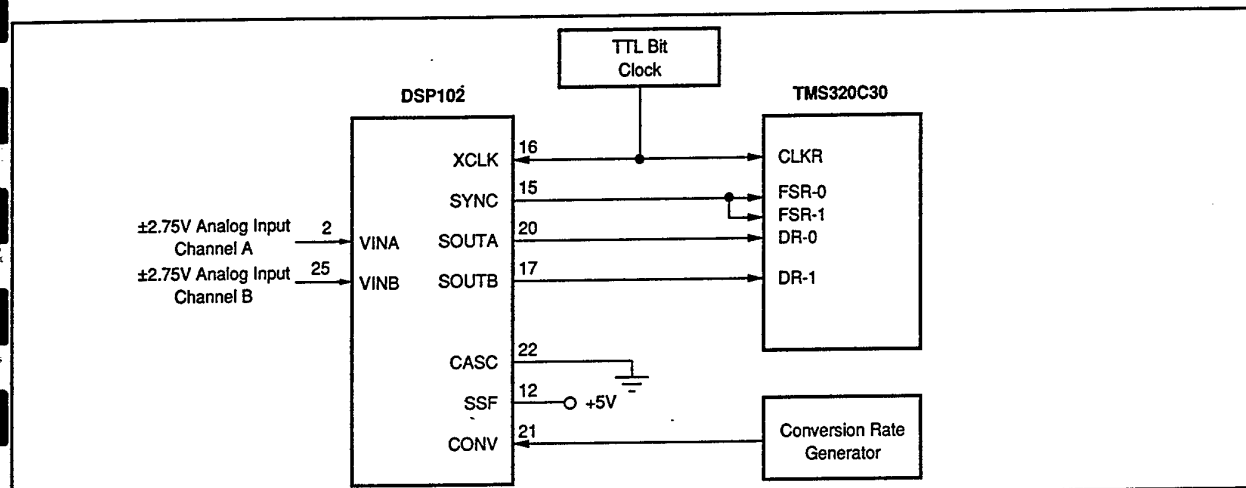


FIGURE 12. Using DSP102 with TMS320C30.

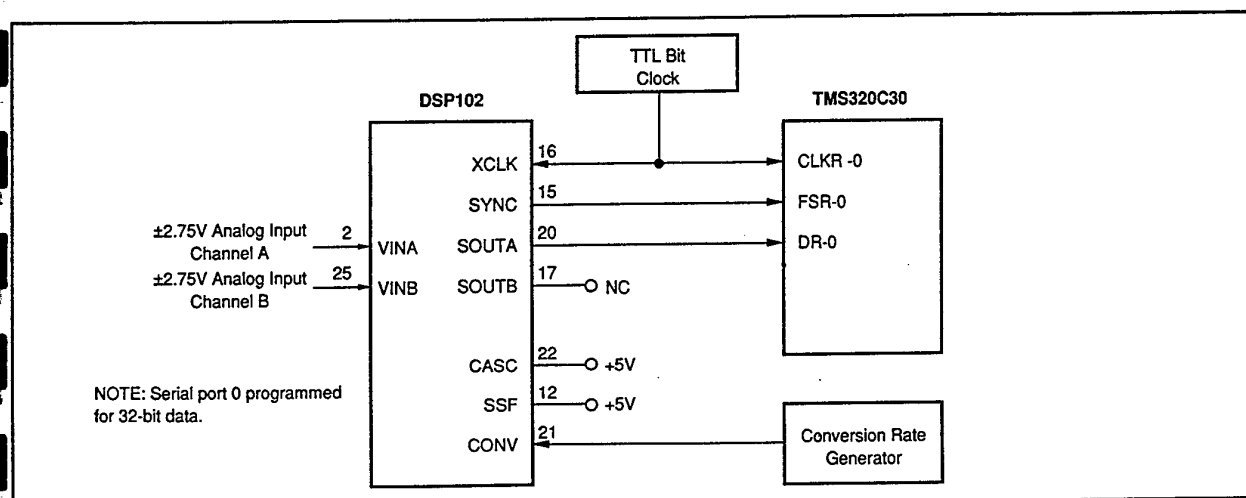


FIGURE 13. Using DSP102 with TMS320C30 in Cascade Mode.

based on using the TME320Cxx in the mode where SSF (Select Synch Format, pin 12) is tied HIGH, so that there is an active High synchronization pulse generated by the DSP101 or DSP102 after receiving a convert command. The synchronization pulse can be changed to active Low simply by making SSF LOW, where appropriate, without changing the basic operation of the A/Ds.

In all cases, the DSP101 and DSP102 will transmit data MSB-first, and the TMS320Cxx needs to be programmed for this.

Figure 11 shows a circuit for using the TMS320C25 or TMS320C30 in a complete analog input and analog output system using the DSP101 along with the Burr-Brown DSP201 D/A.

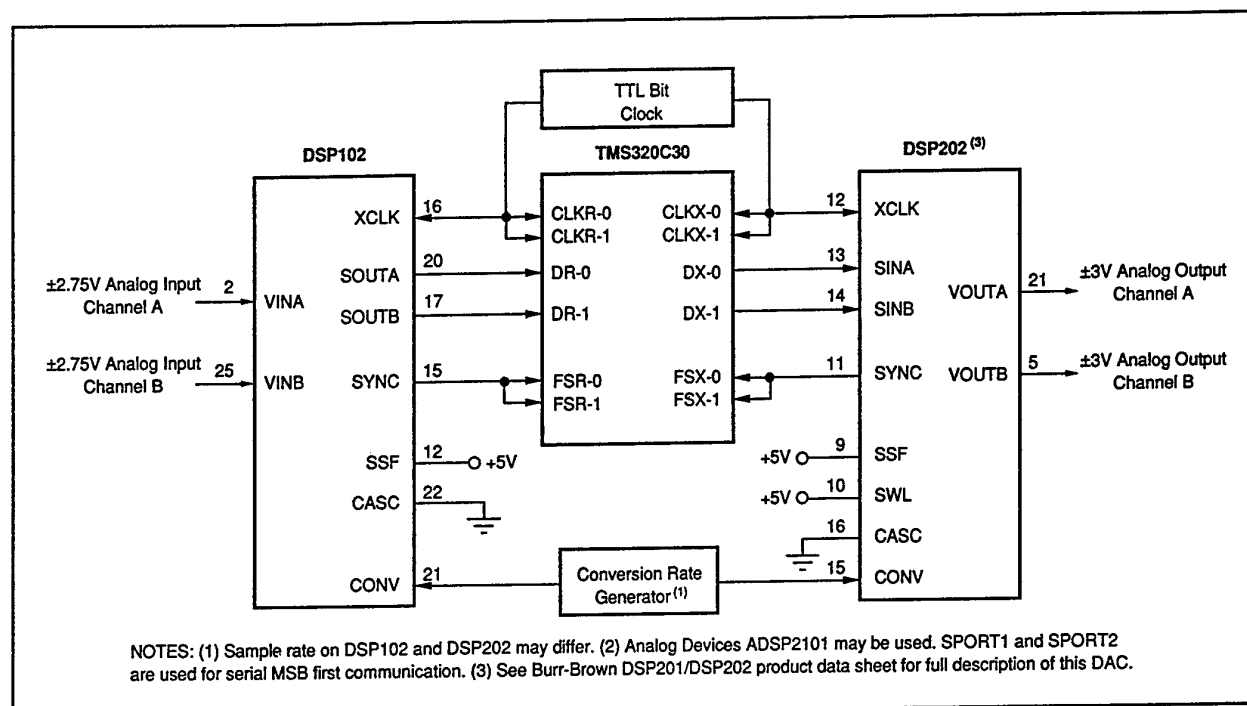


FIGURE 14. Two-Channel Analog Input and Output System with TMS320C30.

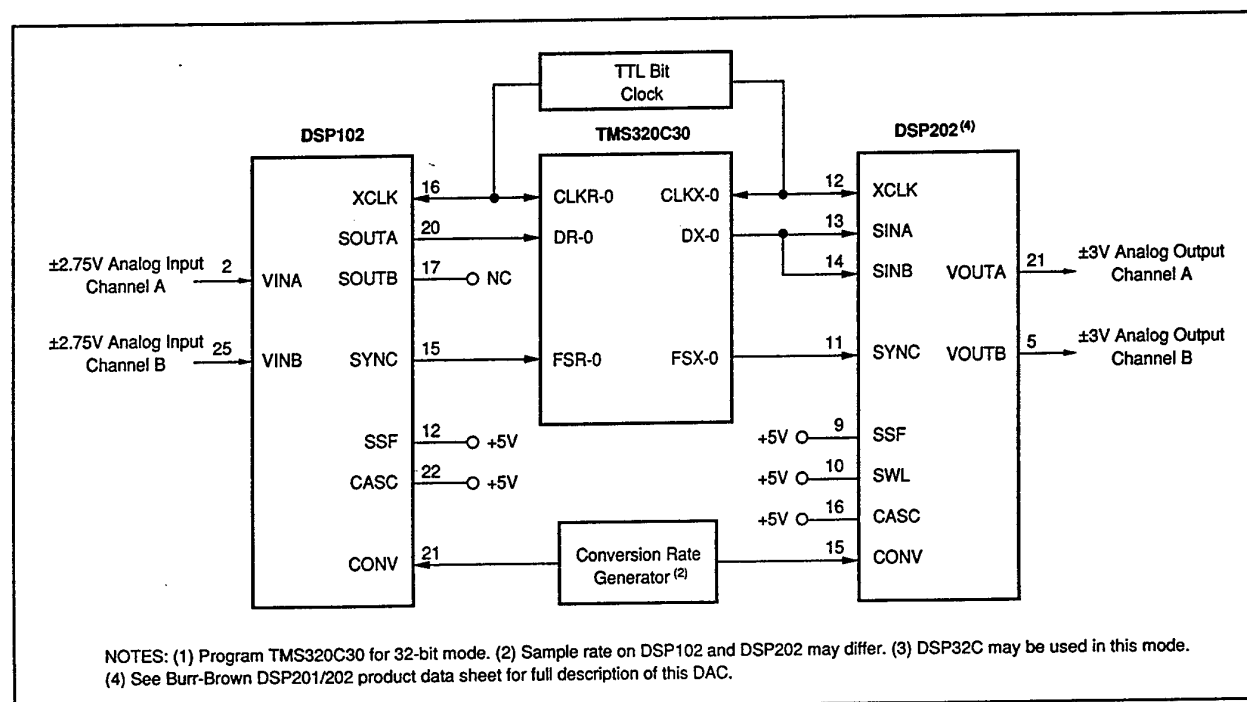


FIGURE 15. Two-Channel Analog Input and Output System with TMS320C30 in Cascade Mode.

USING TMS320C31 TO GENERATE ALL CONTROL SIGNALS

Figure 17 shows a circuit for using the TMS320C31 with a DSP102 and a Burr-Brown DSP202 D/A to provide a two channel analog I/O system. The flexibility of the TMS320C31 allows it to generate the data transfer clock (XCLK) and the Convert Command, minimizing additional circuitry and synchronizing the timing signals to the processor's master

clock. In this circuit, the DSP102 and DSP202 are used in their Cascade modes, transmitting and receiving two channels of data in a single 32-bit word. (See the Cascade Mode section above.)

Table II shows how to set up the circuit in Figure 17 for a 44.1kHz conversion rate for both channels of the DSP102 A/D and both channels of the DSP202 D/A. Both inputs and outputs will be simultaneously converted.

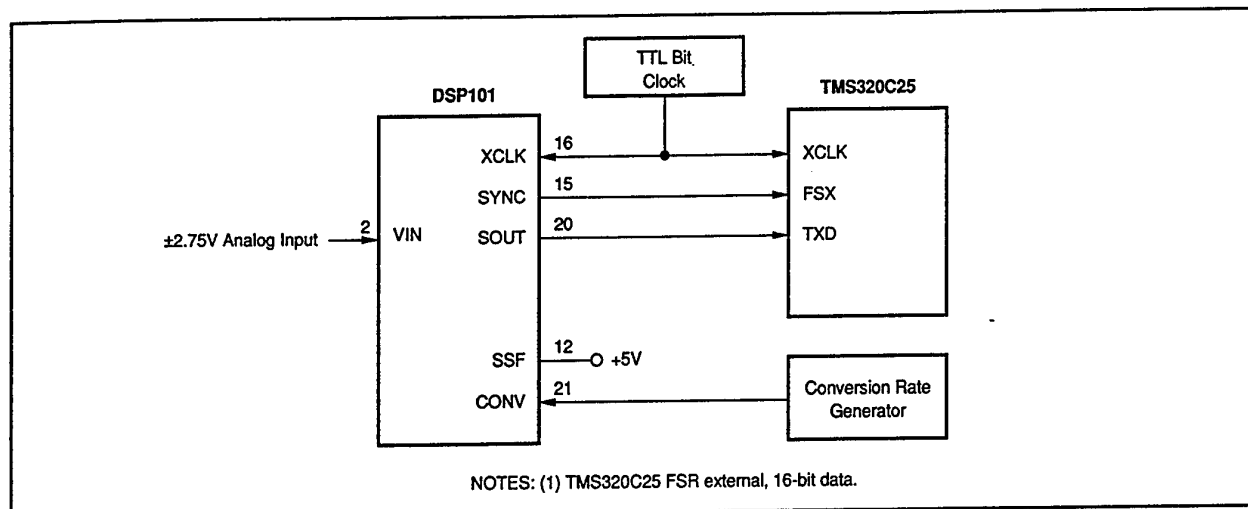


FIGURE 16. Using DSP101 with TMS320C25.

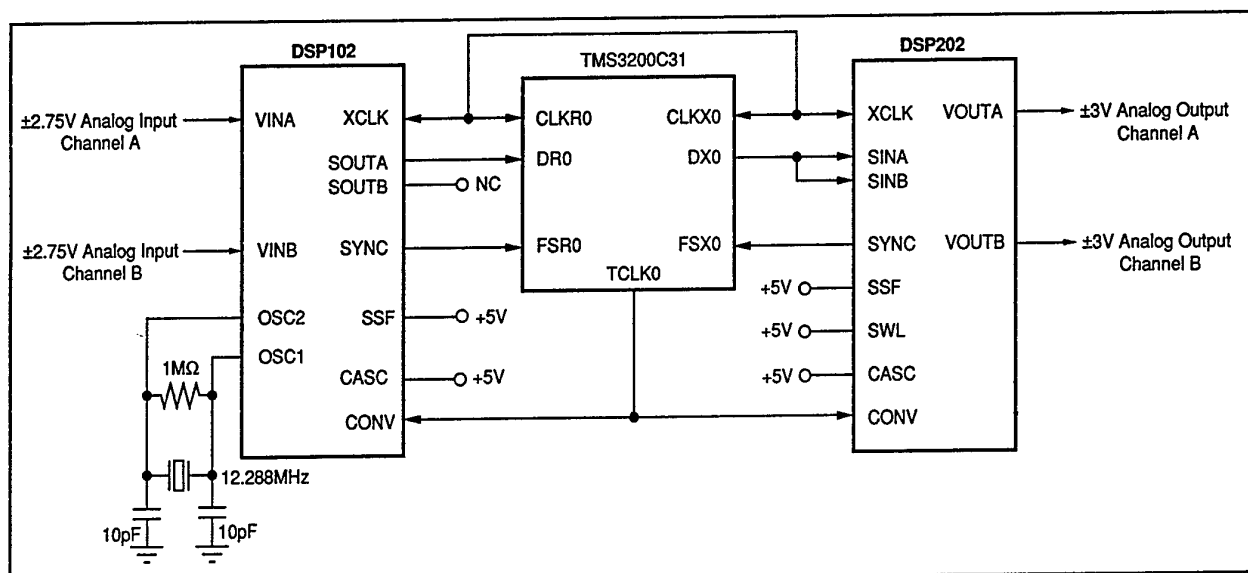


FIGURE 17. Two Channel Analog I/O Using TMS320C31.

SERIAL PORT

Port Global Control Register	0x0EBC040
FSX/DX/CLKX Port Control Register	0x00000111
FSR/DR/CLKR Port Control Register	0x00000111
Receive/Transmit Timer Control Register	0x0000000F

TIMER

Timer Global Control Register	0x000002C1
Timer Period Register	0x000000B5

NOTE: Assumes TMS320C31 has 32MHz Master Clock.

TABLE II. TMS320C31 Register Settings for 44.1kHz Conversion Rate in Figure 17.

USING DSP101 AND DSP102 WITH MOTOROLA DSP ICS

Figure 18 shows how to use the DSP101 with a Motorola DSP56001. Using the DSP102 requires using two DSP56001s. The DSP56001 needs to be programmed to receive data MSB-first with SYNC in the Bit Mode.

SSF (pin 12) needs to be tied HIGH for using either the DSP101 or the DSP102 with DSP56001s. This will cause the DSP101 or DSP102 to transmit an appropriate active High synchronization pulse on SYNC (pin 15) after a convert command is received by the A/D. Timing is shown in Figure 1.

USING DSP101 AND DSP102 WITH AT&T DSP ICS

Figures 11, 19, 20, and 21 show how to use the DSP101 and

DSP102 with the DSP16 and DSP32C in different modes. The AT&T processors need to be programmed to accept data MSB-first, and the DSP101 or DSP102 needs to have SSF (pin 12) tied LOW, so that an appropriate active Low synchronization pulse will be transmitted by the A/D after a convert command is received.

Figures 19 and 20 show the DSP32C and DSP16 respectively used with the DSP101 to handle a single analog input channel.

Figure 21 shows how to transmit to a single DSP32C conversion results from both DSP102 channels in a single 32-bit word, using the Cascade mode on the A/D.

Figure 11 indicates how to build a complete analog input and analog output system using a DSP32C or DSP16 with a DSP101 and a Burr-Brown DSP201 D/A.

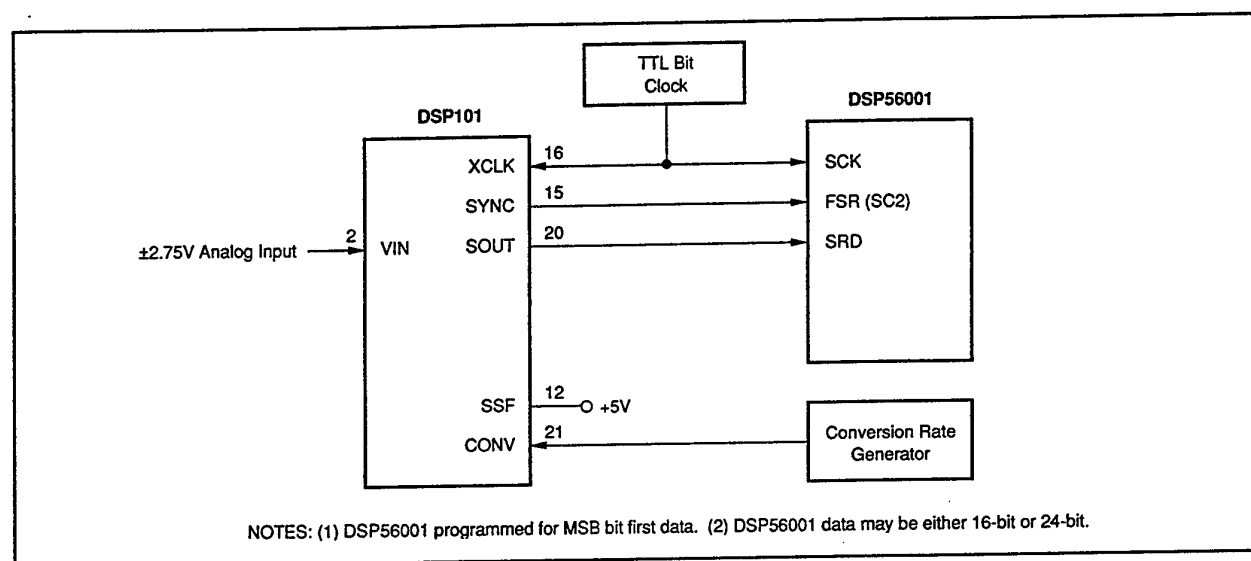


FIGURE 18. Using DSP101 with DSP56001.

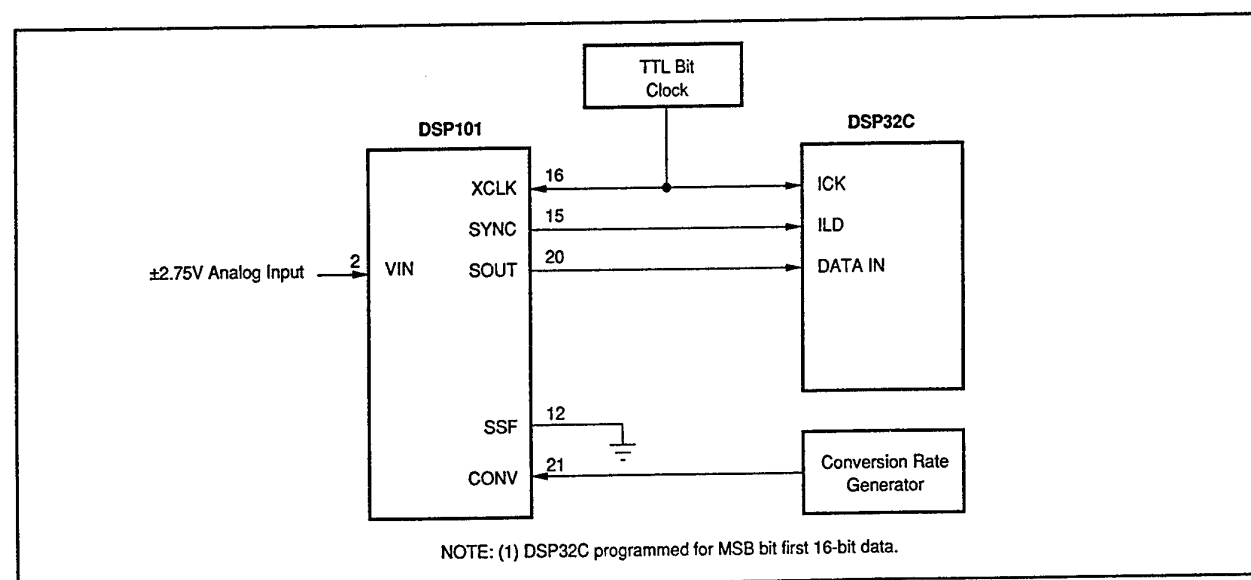


FIGURE 19. Using DSP101 with DSP32C.

USING DSP101 AND DSP102 WITH ADI DSP ICS

When using the DSP101 or DSP102 with the fixed-point ADSP21xx series, the processors need to be programmed to receive data MSB-first.

Figure 22 shows how to use the DSP102 with an ADSP2101 to provide a two-channel simultaneous sampling system.

Figure 23 shows the connections required to generate an analog input channel using an ADSP2105 with the DSP101.

The same basic circuit can be used to connect a DSP101 to the ADSP2101.

Figure 11 indicates how to build a complete analog I/O system using either the ADSP2101 or the ADSP2105 with a DSP101 and a Burr-Brown DSP201 D/A.

The two serial ports on the ADSP2101 can also be used with the DSP102 and the Burr-Brown DSP202 D/A to make two complete analog I/O channels, as indicated in footnote 2 of Figure 14.

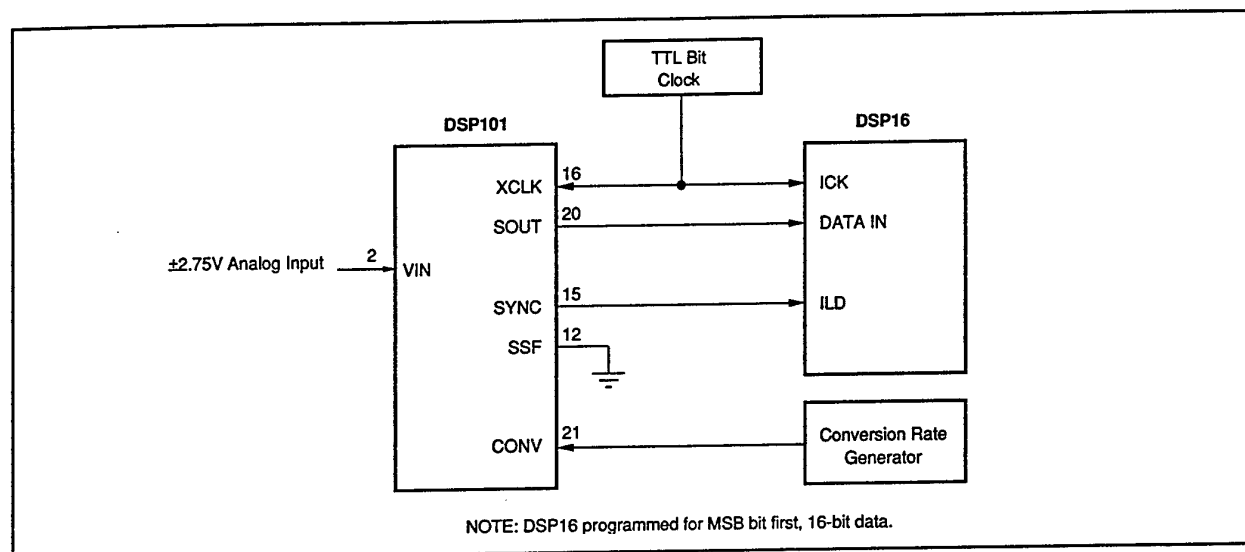


FIGURE 20. Using DSP101 with DSP16.

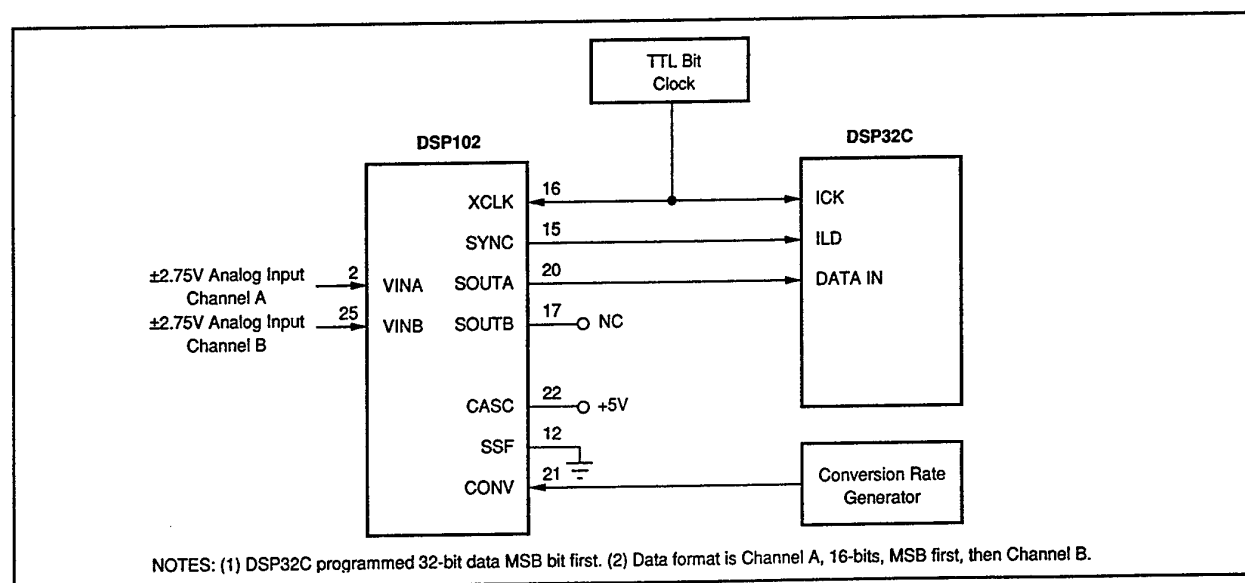


FIGURE 21. Using DSP102 with DSP32C in Cascade Mode.

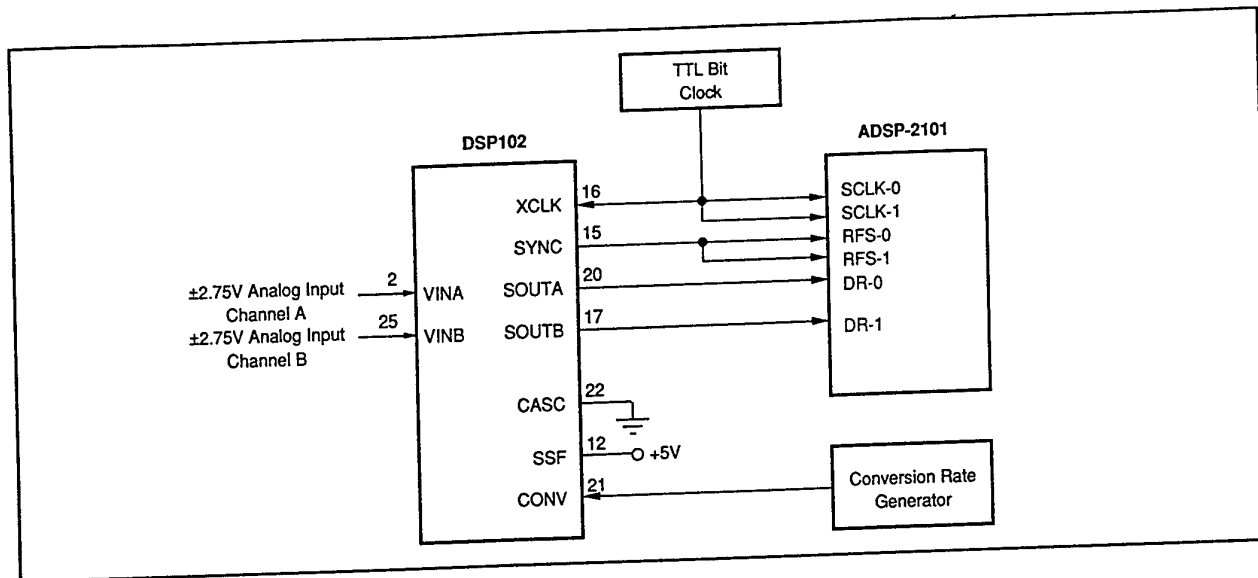


FIGURE 22. Using DSP102 with ADSP-2101.

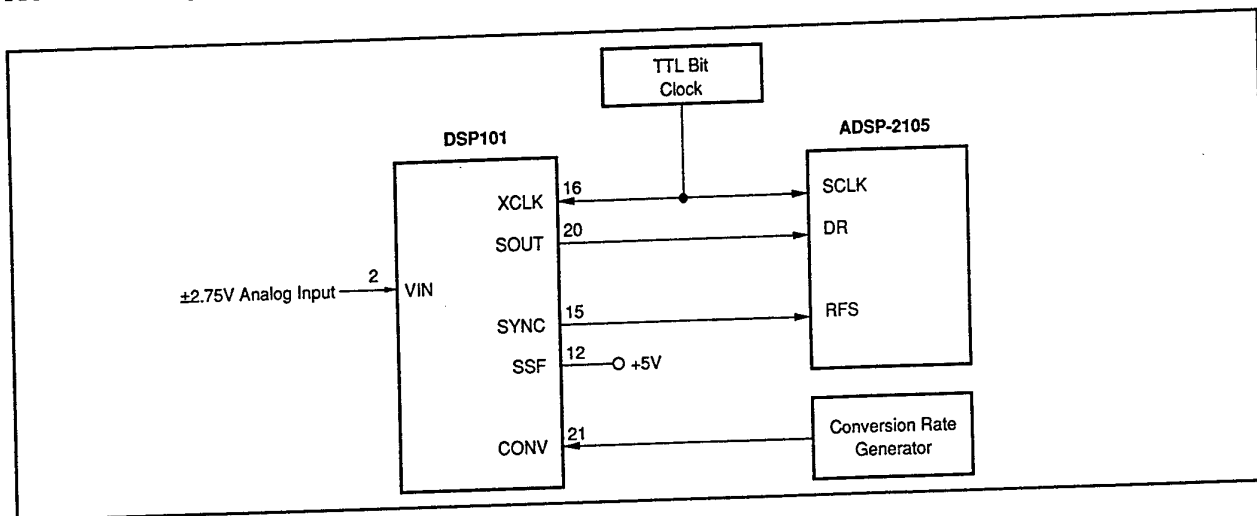


FIGURE 23. Using DSP101 with ADSP-2105.

DEM-DSP102/202 EVALUATION BOARD

An evaluation fixture, the DEM-DSP102/202, is available to simplify evaluation of the DSP101 and DSP102, and the companion digital-to-analog converters, the single DSP201 and dual DSP202. The DEM-DSP102/202 comes complete with a socketed DSP102 and DSP202, a breadboard area, TTL I/O headers and differential line drivers for data trans-

fer options, a complete clocking circuit for the conversion clock and bit clock, and analog filter modules. The board makes it easy to go from design concept to working prototype of a DSP-based system, offering two complete analog I/O channels.

Contact your local Burr-Brown representative for a full data sheet on the DEM-DSP102/202.

CMOS single-chip 8-bit microcontrollers

83C750/87C750

DESCRIPTION

The Philips 8XC750 offers the advantages of the 80C51 architecture in a small package and at low cost.

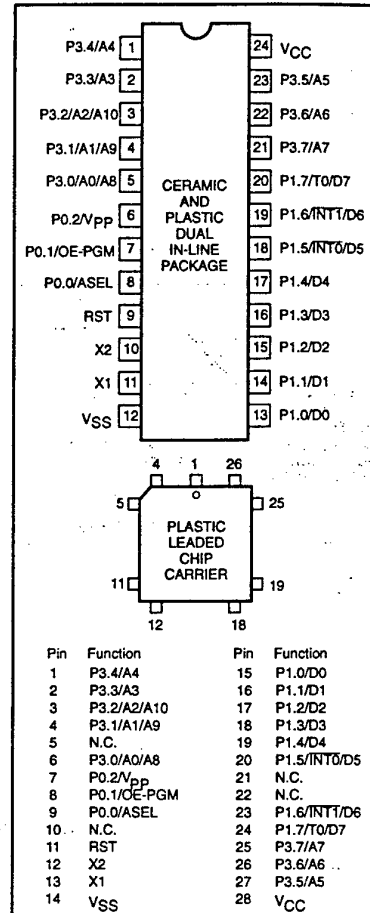
The 8XC750 Microcontroller is fabricated with Philips high-density CMOS technology. Philips epitaxial substrate minimizes CMOS latch-up sensitivity.

The 87C750 contains a 1k x 8 EPROM, a 64 x 8 RAM, 19 I/O lines, a 16-bit auto-reload counter/timer, a five-source, fixed-priority level interrupt structure and an on-chip oscillator.

FEATURES

- 80C51 based architecture
- Wide oscillator frequency range—up to 40MHz
- Small package sizes
 - 24-pin DIP (300 mil "skinny DIP")
 - 28-pin PLCC
- 87C750 available in erasable quartz lid or one-time programmable plastic packages
- Low power consumption:
 - Normal operation: less than 11mA @ 5V, 12MHz
 - Idle mode
 - Power-down mode
- 1k x 8 EPROM (87C750)
- 64 x 8 RAM
- 16-bit auto reloadable counter/timer
- Boolean processor
- CMOS and TTL compatible
- Well suited for logic replacement, consumer and industrial applications
- LED drive outputs

PIN CONFIGURATIONS



ORDERING INFORMATION

ROM	EPROM ¹		TEMPERATURE RANGE °C AND PACKAGE	FREQUENCY	DRAWING NUMBER
	P87C750EBF FA	UV	0 to +70, Ceramic Dual In-line Package	3.5 to 16MHz	0586B
	P87C750EFF FA	UV	−40 to +85, Ceramic Dual In-line Package	3.5 to 16MHz	0586B
P83C750EBP N	P87C750EBP N	OTP	0 to +70, Plastic Dual In-line Package	3.5 to 16MHz	0410D
P83C750EFP N	P87C750EFP N	OTP	−40 to +85, Plastic Dual In-line Package	3.5 to 16MHz	0410D
P83C750EBA A	P87C750EBA A	OTP	0 to +70, Plastic Lead Chip Carrier	3.5 to 16MHz	0401F
P83C750EFA A	P87C750EFA A	OTP	−40 to +85, Plastic Lead Chip Carrier	3.5 to 16MHz	0401F
P83C750PBP N	P87C750PBP N	OTP	0 to +70, Plastic Dual In-line Package	3.5 to 40MHz	0410D
P83C750PFP N	P87C750PFP N	OTP	−40 to +85, Plastic Dual In-line Package	3.5 to 40MHz	0410D
P83C750PBA A	P87C750PBA A	OTP	0 to +70, Plastic Lead Chip Carrier	3.5 to 40MHz	0401F
P83C750PFA A	P87C750PFA A	OTP	−40 to +85, Plastic Lead Chip Carrier	3.5 to 40MHz	0401F
	P87C750PBF FA	UV	0 to +70, Ceramic Dual In-line Package	3.5 to 40MHz	0586B
	P87C750PFF FA	UV	−40 to +85, Ceramic Dual In-line Package	3.5 to 40MHz	0586B

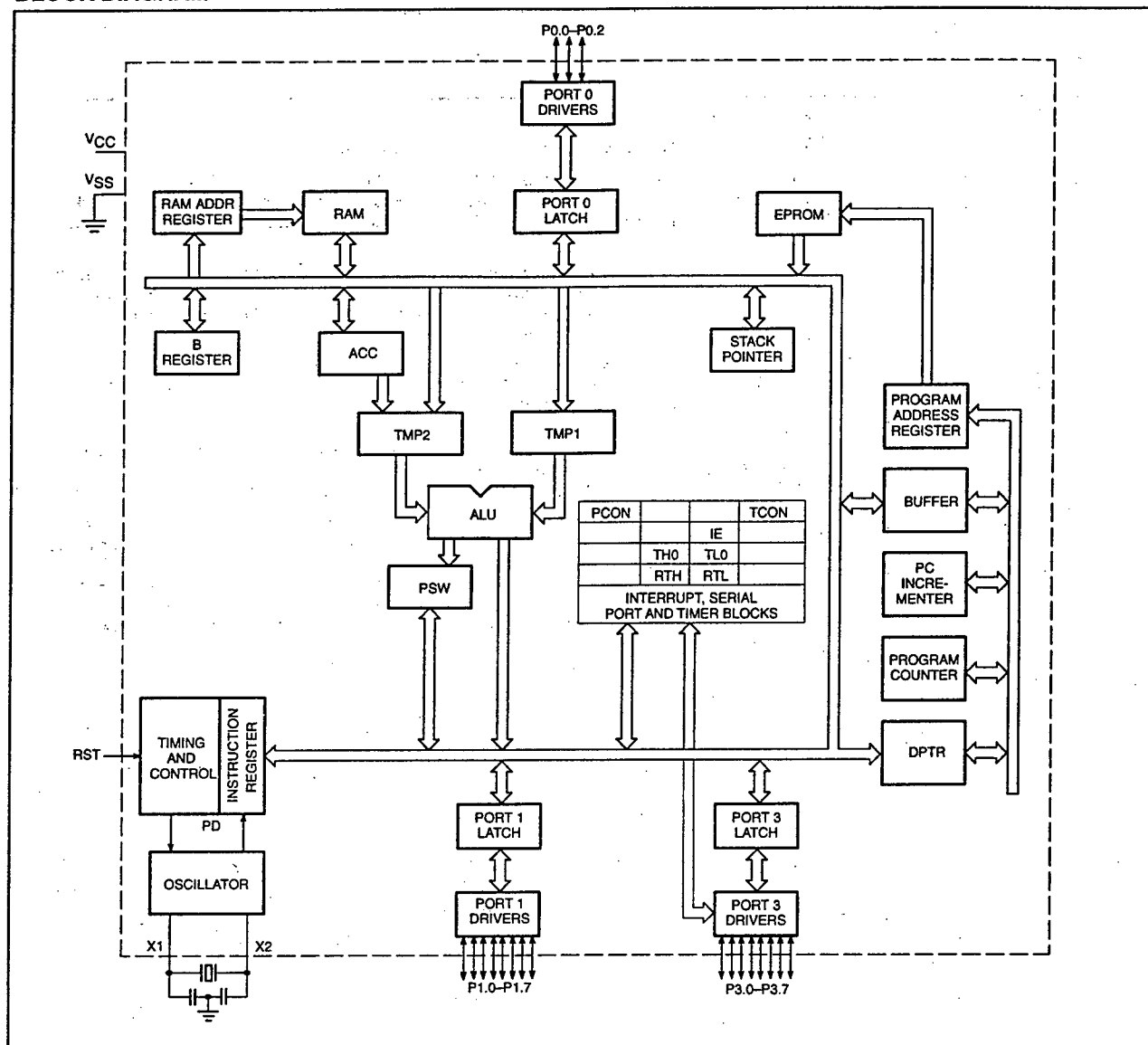
NOTE:

1. OTP = One Time Programmable EPROM. UV = UV Erasable EPROM.

CMOS single-chip 8-bit microcontrollers

83C750/87C750

BLOCK DIAGRAM



CMOS single-chip 8-bit microcontrollers

83C750/87C750

PIN DESCRIPTIONS

MNEMONIC	PIN NO.		TYPE	NAME AND FUNCTION
	DIP/SSOP	LCC		
V _{SS}	12	14	I	Circuit Ground Potential
V _{CC}	24	28	I	Supply voltage during normal, idle, and power-down operation.
P0.0-P0.2	8-6	9-7	I/O	Port 0: Port 0 is a 3-bit open-drain, bidirectional port. Port 0 pins that have 1s written to them float, and in that state can be used as high-impedance inputs. These pins are driven low if the port register bit is written with a 0. The state of the pin can always be read from the port register by the program. P0.0 and P0.1 are open drain bidirectional I/O pins with the electrical characteristics listed in the tables that follow. While these differ from "standard TTL" characteristics, they are close enough for the pins to still be used as general-purpose I/O. Port 0 also provides alternate functions for programming the EPROM memory as follows: V_{PP} (P0.2) – Programming voltage input. OE/PGM (P0.1) – Input which specifies verify mode (output enable) or the program mode. OE/PGM = 1 output enabled (verify mode). OE/PGM = 0 program mode. ASEL (P0.0) – Input which indicates which bits of the EPROM address are applied to port 3. ASEL = 0 low address byte available on port 3. ASEL = 1 high address byte available on port 3 (only the three least significant bits are used).
P1.0-P1.7	6	7	N/A	
	7	8	I	
	8	9	I	
	13-20	15-20, 23, 24	I/O	Port 1: Port 1 is an 8-bit bidirectional I/O port with internal pull-ups. Port 1 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 1 pins that are externally pulled low will source current because of the internal pull-ups. (See DC Electrical Characteristics: I_{IL}). Port 1 serves to output the addressed EPROM contents in the verify mode and accepts as inputs the value to program into the selected address during the program mode. Port 1 also serves the special function features of the 80C51 family as listed below: INT0 (P1.5): External interrupt. INT1 (P1.6): External interrupt. T0 (P1.7): Timer 0 external input.
P3.0-P3.7	5-1, 23-21	4-1, 6, 27-25	I/O	Port 3: Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. Port 3 pins that have 1s written to them are pulled high by the internal pull-ups and can be used as inputs. As inputs, port 3 pins that are externally being pulled low will source current because of the pull-ups. (See DC Electrical Characteristics: I_{IL}). Port 3 also functions as the address input for the EPROM memory location to be programmed (or verified). The 10-bit address is multiplexed into this port as specified by P0.0/ASEL.
RST	9	11	I	Reset: A high on this pin for two machine cycles while the oscillator is running, resets the device. An internal diffused resistor to V_{SS} permits a power-on RESET using only an external capacitor to V_{CC}. After the device is reset, a 10-bit serial sequence, sent LSB first, applied to RESET, places the device in the programming state allowing programming address, data and V_{PP} to be applied for programming or verification purposes. The RESET serial sequence must be synchronized with the X1 input.
X1	11	13	I	Crystal 1: Input to the inverting oscillator amplifier and input to the internal clock generator circuits. X1 also serves as the clock to strobe in a serial bit stream into RESET to place the device in the programming state.
X2	10	12	O	Crystal 2: Output from the inverting oscillator amplifier.

CMOS single-chip 8-bit microcontrollers

83C750/87C750

**OSCILLATOR
CHARACTERISTICS**

X1 and X2 are the input and output, respectively, of an inverting amplifier which can be configured for use as an on-chip oscillator.

To drive the device from an external clock source, X1 should be driven while X2 is left unconnected. There are no requirements on the duty cycle of the external clock signal, because the input to the internal clock circuitry is through a divide-by-two flip-flop. However, minimum and maximum high and low times specified in the data sheet must be observed.

RESET

A reset is accomplished by holding the RST pin high for at least two machine cycles (24 oscillator periods), while the oscillator is running. To insure a good power-up reset, the RST pin must be high long enough to allow the oscillator time to start up (normally a few milliseconds) plus two machine cycles. At power-up, the voltage on V_{CC} and RST must come up at the same time for a proper start-up.

IDLE MODE

In idle mode, the CPU puts itself to sleep while all of the on-chip peripherals stay active. The instruction to invoke the idle mode is the last instruction executed in the normal operating mode before the idle mode is activated. The CPU contents, the on-chip RAM, and all of the special function registers remain intact during this mode. The idle mode can be terminated either by any enabled interrupt (at which time the process is picked up at the interrupt service routine and continued), or by a hardware reset which starts the processor in the same manner as a power-on reset.

POWER-DOWN MODE

In the power-down mode, the oscillator is stopped and the instruction to invoke power-down is the last instruction executed. Only the contents of the on-chip RAM are preserved. A hardware reset is the only way to terminate the power-down mode, the control bits for the reduced power modes are in the special function register PCON.

**Table 1. External Pin Status
During Idle and
Power-Down Modes**

MODE	Port 0	Port 1	Port 2
Idle	Data	Data	Data
Power-down	Data	Data	Data

**DIFFERENCES BETWEEN THE
8XC750 AND THE 80C51****Program Memory**

On the 8XC750, program memory is 1024 bytes long and is not externally expandable, so the 80C51 instructions MOVX, LJMP, and LCALL are not implemented. The only fixed locations in program memory are the addresses at which execution is taken up in response to reset and interrupts, which are as follows:

Event	Program Memory Address
Reset	000
External INT0	003
Counter/timer 0	00B
External INT1	013

Counter/Timer Subsystem

The 87C750 has one counter/timer called timer/counter 0. Its operation is similar to mode 2 operation on the 80C51, but is extended to 16 bits with 16 bits of autoloader. The controls for this counter are centralized in a single register called TCON.

**Interrupt Subsystem – Fixed
Priority**

The IP register and the 2-level interrupt system of the 80C51 are eliminated. Simultaneous interrupt conditions are resolved by a single-level, fixed priority as follows:

Highest priority: Pin INT0
Counter/timer flag 0
Pin INT1

**Special Function Register
Addresses**

Special function registers for the 8XC750 are identical to those of the 80C51, except for the changes listed below:

80C51 special function registers not present in the 8XC750 are TMOD (89), P2 (A0) and IP (B8). The 80C51 registers TH1 and TL1 are replaced with the 87C750 registers RTH and RTL respectively (refer to Table 2).

CMOS single-chip 8-bit microcontrollers

83C750/87C750

Table 2. 87C750 Special Function Registers

SYMBOL	DESCRIPTION	DIRECT ADDRESS	BIT ADDRESS, SYMBOL, OR ALTERNATIVE PORT FUNCTION								RESET VALUE	
			MSB				LSB					
ACC*	Accumulator	E0H	E7	E6	E5	E4	E3	E2	E1	E0	00H	
B*	B register	F0H	F7	F6	F5	F4	F3	F2	F1	F0	00H	
DPTR:	Data pointer (2 bytes)											
DPH	High byte	83H									00H	
DPL	Low byte	82H									00H	
			AF	AE	AD	AC	AB	AA	A9	A8		
IE*#	Interrupt enable	A8H	EA	-	-	-	-	-	EX1	ET0	EX0	00H
							82	81	80			
P0*#	Port 0	80H	-	-	-	-	-	-	-	-	-	xxxxxx111B
			97	96	95	94	93	92	91	90		
P1*	Port 1	90H	T0	INT1	INT0	-	-	-	-	-	-	FFH
P3*	Port 3	B0H	B7	B6	B5	B4	B3	B2	B1	B0		FFH
PCON#	Power control	87H	-	-	-	-	-	-	-	PD	IDL	xxxxxxx00B
			D7	D6	D5	D4	D3	D2	D1	D0		
PSW*	Program status word	D0H	CY	AC	F0	RS1	RS0	OV	-	P		00H
SP	Stack pointer	81H										07H
			8F	8E	8D	8C	8B	8A	89	88		
TCON*#	Timer/counter control	88H	GATE	C/T	TF	TR	IE0	IT0	IE1	IT1		00H
TL#	Timer low byte	8AH										00H
TH#	Timer high byte	8CH										00H
RTL#	Timer low reload	8BH										00H
RTH#	Timer high reload	8DH										00H

* SFRs are bit addressable.

SFRs are modified from or added to the 80C51 SFRs.

ABSOLUTE MAXIMUM RATINGS^{1, 2}

PARAMETER	RATING	UNIT
Storage temperature range	-65 to +150	°C
Voltage from V _{CC} to V _{SS}	-0.5 to +6.5	V
Voltage from any pin to V _{SS} (except V _{PP})	-0.5 to V _{CC} + 0.5	V
Power dissipation	1.0	W
Voltage on V _{PP} pin to V _{SS}	0 to +13.0	V
Maximum I _{OL} per I/O pin	10	mA

NOTES:

- Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification is not implied.
- This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maxima.

CMOS single-chip 8-bit microcontrollers 83C750/87C750

DC ELECTRICAL CHARACTERISTICS

 $T_{amb} = 0^{\circ}\text{C to } +70^{\circ}\text{C or } -40^{\circ}\text{C to } +85^{\circ}\text{C}, V_{CC} = 5\text{V} \pm 10\%, V_{SS} = 0\text{V}^1$

SYMBOL	PARAMETER	TEST CONDITIONS	LIMITS		UNIT
			MIN	MAX	
V_{IL}	Input low voltage		-0.5	$0.2V_{DD} - 0.1$	V
V_{IH}	Input high voltage, except X1, RST		$0.2V_{CC} + 0.9$	$V_{CC} + 0.5$	V
V_{IH1}	Input high voltage, X1, RST		$0.7V_{CC}$	$V_{CC} + 0.5$	V
V_{OL}	Output low voltage, ports 1 and 3	$I_{OL} = 1.6\text{mA}^2$		0.45	V
V_{OL1}	Output low voltage, port 0	$I_{OL} = 3.2\text{mA}^2$		0.45	V
V_{OH}	Output high voltage, ports 1 and 3	$I_{OH} = -60\mu\text{A}$	2.4		V
		$I_{OH} = -25\mu\text{A}$	$0.75V_{CC}$		V
		$I_{OH} = -10\mu\text{A}$	$0.9V_{CC}$		V
C	Capacitance			10	pF
I_{IL}	Logical 0 input current, ports 1 and 3	$V_{IN} = 0.45\text{V}$		-50	μA
I_{TL}	Logical 1 to 0 transition current, ports 1 and 3 ³	$V_{IN} = 2\text{V}$ (0 to $+70^{\circ}\text{C}$)		-650	μA
		$V_{IN} = 2\text{V}$ (-40 to $+85^{\circ}\text{C}$)		-750	μA
I_{LI}	Input leakage current, port 0	$0.45 < V_{IN} < V_{CC}$		± 10	μA
R_{RST}	Internal pull-down resistor		25	175	k Ω
C_{IO}	Pin capacitance	Test freq = 1MHz, $T_{amb} = 25^{\circ}\text{C}$		10	pF
I_{PD}	Power-down current ⁴	$V_{CC} = 2$ to V_{CC} max		50	μA
V_{PP}	V_{PP} program voltage	$V_{SS} = 0\text{V}$ $V_{CC} = 5\text{V} \pm 10\%$ $T_{amb} = 21^{\circ}\text{C to } 27^{\circ}\text{C}$	12.5	13.0	V
I_{PP}	Program current	$V_{PP} = 13.0\text{V}$		50	mA
I_{CC}	Supply current (see Figure 2) ^{5, 6}				

NOTES:

- Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to V_{SS} unless otherwise noted.
- Under steady state (non-transient) conditions, I_{OL} must be externally limited as follows:
 Maximum I_{OL} per port pin: 10mA (NOTE: This is 85°C spec.)
 Maximum I_{OL} per 8-bit port: 26mA
 Maximum total I_{OL} for all outputs: 67mA
 If I_{OL} exceeds the test condition, V_{OL} may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
- Pins of ports 1 and 3 source a transition current when they are being externally driven from 1 to 0. The transition current reaches its maximum value when V_{IN} is approximately 2V.
- Power-down I_{CC} is measured with all output pins disconnected; port 0 = V_{CC} ; X2, X1 n.c.; RST = V_{SS} .
- Active I_{CC} is measured with all output pins disconnected; X1 driven with t_{CLCH} , $t_{CHCL} = 5\text{ns}$, $V_{IL} = V_{SS} + 0.5\text{V}$, $V_{IH} = V_{CC} - 0.5\text{V}$; X2 n.c.; RST = port 0 = V_{CC} . I_{CC} will be slightly higher if a crystal oscillator is used.
- Idle I_{CC} is measured with all output pins disconnected; X1 driven with t_{CLCH} , $t_{CHCL} = 5\text{ns}$, $V_{IL} = V_{SS} + 0.5\text{V}$, $V_{IH} = V_{CC} - 0.5\text{V}$; X2 n.c.; port 0 = V_{CC} ; RST = V_{SS} .

AC ELECTRICAL CHARACTERISTICS

 $T_{amb} = 0^{\circ}\text{C to } +70^{\circ}\text{C or } -40^{\circ}\text{C to } +85^{\circ}\text{C}, V_{CC} = 5\text{V} \pm 10\%, V_{SS} = 0\text{V}^1, 2$

SYMBOL	PARAMETER	VARIABLE CLOCK				UNIT
		MIN	MAX	MIN	MAX	
$1/t_{CLCL}$	Oscillator frequency:	3.5	16	3.5	40	MHz
External Clock (Figure 1)						
t_{CHCX}	High time	20		10		ns
t_{CLCX}	Low time	20		10		ns
t_{CLCH}	Rise time		20		20	ns
t_{CHCL}	Fall time		20		20	ns

NOTES:

- Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to V_{SS} unless otherwise noted.
- Load capacitance for ports = 80pF.

CMOS single-chip 8-bit microcontrollers

83C750/87C750

EXPLANATION OF THE AC SYMBOLS

In defining the clock waveform, care must be taken not to exceed the MIN or MAX limits of the AC electrical characteristics table. Each timing symbol has five characters. The first character is always 't' (= time). The other characters, depending on their positions, indicate the name of a signal or the logical status of that signal. The designations are:

C – Clock

D – Input data

H – Logic level high
L – Logic level low
Q – Output data
T – Time
V – Valid
X – No longer a valid logic level
Z – Float

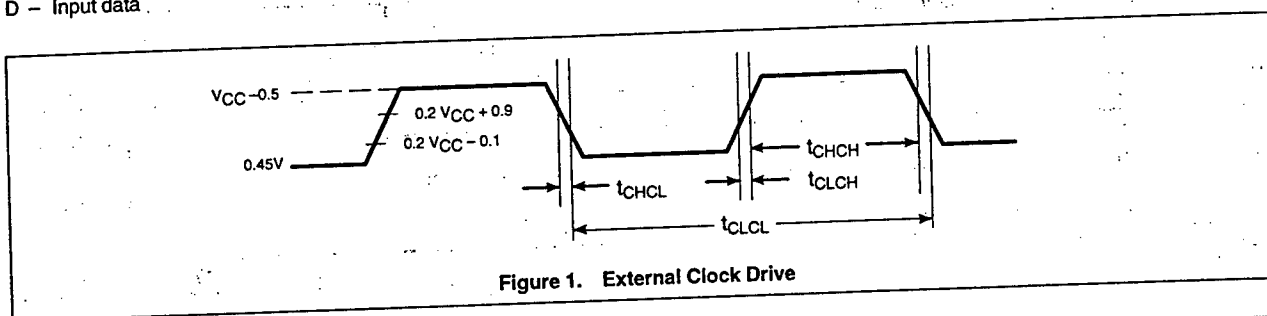
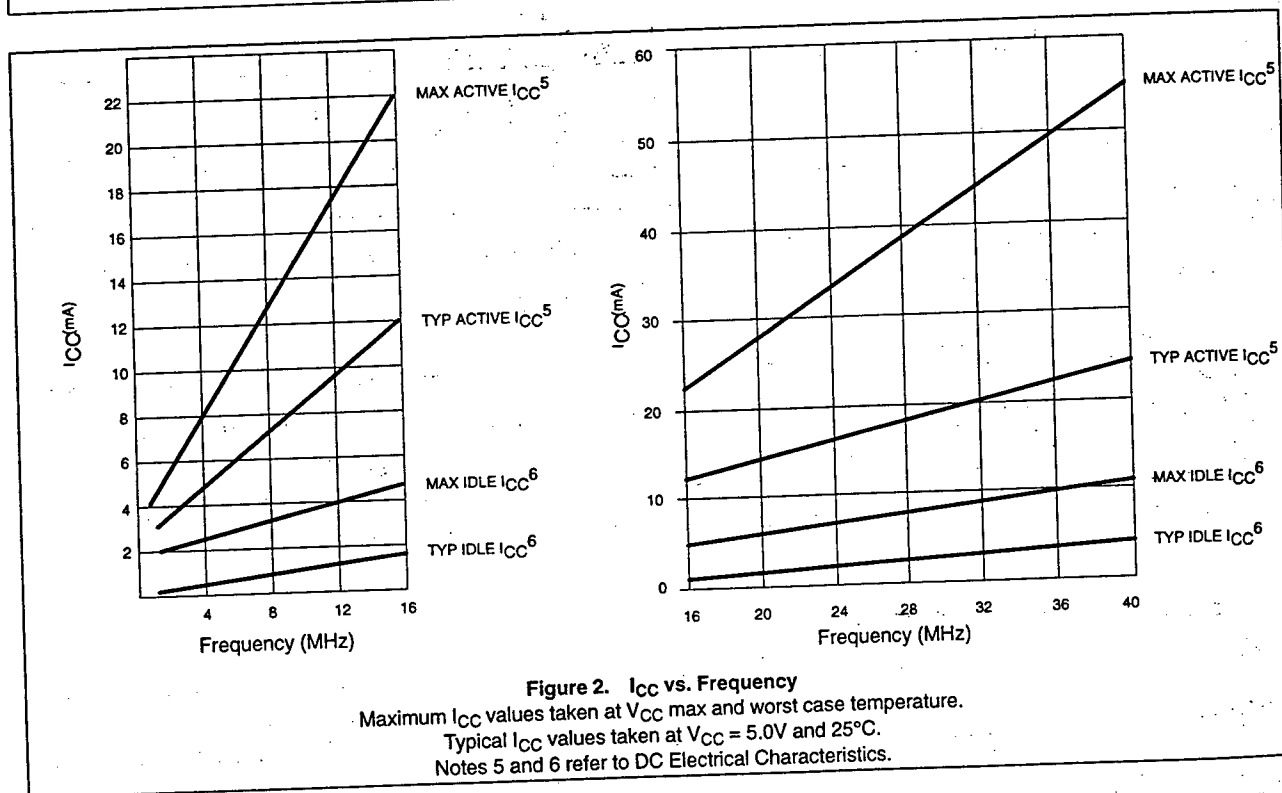


Figure 1. External Clock Drive



CMOS single-chip 8-bit microcontrollers

83C750/87C750

87C750 PROGRAMMING CONSIDERATIONS**EPROM Characteristics**

The 87C750 is programmed by using a modified Quick-Pulse Programming algorithm similar to that used for devices such as the 87C451 and 87C51. It differs from these devices in that a serial data stream is used to place the 87C750 in the programming mode.

Figure 3 shows a block diagram of the programming configuration for the 87C750. Port pin P0.2 is used as the programming voltage supply input (V_{PP} signal). Port pin P0.1 is used as the program (PGM) signal. This pin is used for the 25 programming pulses.

Port 3 is used as the address input for the byte to be programmed and accepts both the high and low components of the eleven bit address. Multiplexing of these address components is performed using the ASEL input. The user should drive the ASEL input high and then drive port 3 with the high order bits of the address. ASEL should remain high for at least 13 clock cycles. ASEL may then be driven low which latches the high order bits of the address internally. The high address should remain on port 3 for at least two clock cycles after ASEL is driven low. Port 3 may then be driven with the low byte of the address. The low address will be internally stable 13 clock cycles later. The address will remain stable provided that the low byte placed on port 3 is held stable and ASEL is kept low. **Note:** ASEL needs to be pulsed high only to change the high byte of the address.

Port 1 is used as a bidirectional data bus during programming and verify operations. During programming mode, it accepts the byte to be programmed. During verify mode, it provides the contents of the EPROM location specified by the address which has been supplied to Port 3.

The XTAL1 pin is the oscillator input and receives the master system clock. This clock should be between 1.2 and 6MHz.

The RESET pin is used to accept the serial data stream that places the 87C750 into various programming modes. This pattern consists of a 10-bit code with the LSB sent first. Each bit is synchronized to the clock input, X1.

Programming Operation

Figures 4 and 5 show the timing diagrams for the program/verify cycle. RESET should

initially be held high for at least two machine cycles. P0.1 (PGM) and P0.2 (V_{PP}) will be at V_{OH} as a result of the RESET operation. At this point, these pins function as normal quasi-bidirectional I/O ports and the programming equipment may pull these lines low. However, prior to sending the 10-bit code on the RESET pin, the programming equipment should drive these pins high (V_{IH}). The RESET pin may now be used as the serial data input for the data stream which places the 87C750 in the programming mode. Data bits are sampled during the clock high time and thus should only change during the time that the clock is low. Following transmission of the last data bit, the RESET pin should be held low.

Next the address information for the location to be programmed is placed on port 3 and ASEL is used to perform the address multiplexing, as previously described. At this time, port 1 functions as an output.

A high voltage V_{PP} level is then applied to the V_{PP} input (P0.2). (This sets Port 1 as an input port). The data to be programmed into the EPROM array is then placed on Port 1. This is followed by a series of programming pulses applied to the PGM/ pin (P0.1). These pulses are created by driving P0.1 low and then high. This pulse is repeated until a total of 25 programming pulses have occurred. At the conclusion of the last pulse, the PGM/ signal should remain high.

The V_{PP} signal may now be driven to the V_{OH} level, placing the 87C750 in the verify mode. (Port 1 is now used as an output port). After four machine cycles (48 clock periods), the contents of the addressed location in the EPROM array will appear on Port 1.

The next programming cycle may now be initiated by placing the address information at the inputs of the multiplexed buffers, driving the V_{PP} pin to the V_{PP} voltage level, providing the byte to be programmed to Port 1 and issuing the 26 programming pulses on the PGM/ pin, bringing V_{PP} back down to the V_C level and verifying the byte.

Programming Modes

The 87C750 has four programming features incorporated within its EPROM array. These include the USER EPROM for storage of the application's code, a 16-byte encryption key array and two security bits. Programming and verification of these four elements are selected by a combination of the serial data stream applied to the RESET pin and the voltage levels applied to port pins P0.1 and

P0.2. The various combinations are shown in Table 3.

Encryption Key Table

The 87C750 includes a 16-byte EPROM array that is programmable by the end user. The contents of this array can then be used to encrypt the program memory contents during a program memory verify operation. When a program memory verify operation is performed, the contents of the program memory location is XNOR'ed with one of the bytes in the 16-byte encryption table. The resulting data pattern is then provided to port 1 as the verify data. The encryption mechanism can be disabled, in essence, by leaving the bytes in the encryption table in their erased state (FFH) since the XNOR product of a bit with a logical one will result in the original bit. The encryption bytes are mapped with the code memory in 16-byte groups. The first byte in code memory will be encrypted with the first byte in the encryption table; the second byte in code memory will be encrypted with the second byte in the encryption table and so forth up to and including the 16th byte. The encryption repeats in 16-byte groups; the 17th byte in the code memory will be encrypted with the first byte in the encryption table, and so forth.

Security Bits

Two security bits, security bit 1 and security bit 2, are provided to limit access to the USER EPROM and encryption key arrays. Security bit 1 is the program inhibit bit, and once programmed performs the following functions:

1. Additional programming of the USER EPROM is inhibited.
2. Additional programming of the encryption key is inhibited.
3. Verification of the encryption key is inhibited.
4. Verification of the USER EPROM and the security bit levels may still be performed.

(If the encryption key array is being used, this security bit should be programmed by the user to prevent unauthorized parties from reprogramming the encryption key to all logical zero bits. Such programming would provide data during a verify cycle that is the logical complement of the USER EPROM contents).

Security bit 2, the verify inhibit bit, prevents verification of both the USER EPROM array and the encryption key arrays. The security bit levels may still be verified.

CMOS single-chip 8-bit microcontrollers

83C750/87C750

Programming and Verifying Security Bits

Security bits are programmed employing the same techniques used to program the USER EPROM and KEY arrays using serial data streams and logic levels on port pins indicated in Table 3. When programming either security bit, it is not necessary to provide address or data information to the 87C750 on ports 1 and 3.

Verification occurs in a similar manner using the RESET serial stream shown in Table 3. Port 3 is not required to be driven and the results of the verify operation will appear on ports 1.6 and 1.7.

Ports 1.7 contains the security bit 1 data and is a logical one if programmed and a logical zero if erased. Likewise, P1.6 contains the security bit 2 data and is a logical one if programmed and a logical zero if erased.

Erasure Characteristics

Erasure of the EPROM begins to occur when the chip is exposed to light with wavelengths shorter than approximately 4,000 angstroms. Since sunlight and fluorescent lighting have wavelengths in this range, exposure to these light sources over an extended time (about 1 week in sunlight, or 3 years in room level fluorescent lighting) could cause inadvertent

erasure. For this and secondary effects, it is recommended that an opaque label be placed over the window. For elevated temperature or environments where solvents are being used, apply Kapton tape Flourless part number 2345-5 or equivalent.

The recommended erasure procedure is exposure to ultraviolet light (at 2537 angstroms) to an integrated dose of at least 15W-s/cm². Exposing the EPROM to an ultraviolet lamp of 12,000μW/cm² rating for 20 to 39 minutes, at a distance of about 1 inch, should be sufficient.

Erasure leaves the array in an all 1s state.

Table 3. Implementing Program/Verify Modes

OPERATION	SERIAL CODE	P0.1 (PGM)	P0.2 (V _{PP})
Program user EPROM	296H	—1	V _{PP}
Verify user EPROM	296H	V _{IH}	V _{IH}
Program key EPROM	292H	—1	V _{PP}
Verify key EPROM	292H	V _{IH}	V _{IH}
Program security bit 1	29AH	—1	V _{PP}
Program security bit 2	298H	—1	V _{PP}
Verify security bits	29AH	V _{IH}	V _{IH}

NOTE:

1. Pulsed from V_{IH} to V_{IL} and returned to V_{IH}.

EPROM PROGRAMMING AND VERIFICATION

T_{amb} = 21°C to +27°C, V_{CC} = 5V ±10%, V_{SS} = 0V

SYMBOL	PARAMETER	MIN	MAX	UNIT
1/t _{CLCL}	Oscillator/clock frequency	1.2	6	MHz
t _{AVGL} ¹	Address setup to P0.1 (PROG-) low	10μs + 24t _{CLCL}		
t _{GHAX}	Address hold after P0.1 (PROG-) high	48t _{CLCL}		
t _{DVGL}	Data setup to P0.1 (PROG-) low	38t _{CLCL}		
t _{DVGL}	Data setup to P0.1 (PROG-) low	38t _{CLCL}		
t _{GHDX}	Data hold after P0.1 (PROG-) high	36t _{CLCL}		
t _{SHGL}	V _{PP} setup to P0.1 (PROG-) low	10		μs
t _{GHSL}	V _{PP} hold after P0.1 (PROG-)	10		μs
t _{GLGH}	P0.1 (PROG-) width	90	110	μs
t _{AVQV} ²	V _{PP} low (V _{CC}) to data valid		48t _{CLCL}	
t _{GHGL}	P0.1 (PROG-) high to P0.1 (PROG-) low	10		μs
t _{SYNL}	P0.0 (sync pulse) low	4t _{CLCL}		
t _{SYNH}	P0.0 (sync pulse) high	8t _{CLCL}		
t _{MASEL}	ASEL high time	13t _{CLCL}		
t _{MAHLD}	Address hold time	2t _{CLCL}		
t _{HASET}	Address setup to ASEL	13t _{CLCL}		
t _{ADSTA}	Low address to valid data		48t _{CLCL}	

NOTES:

1. Address should be valid at least 24t_{CLCL} before the rising edge of P0.2 (V_{PP}).
2. For a pure verify mode, i.e., no program mode in between, t_{AVQV} is 14t_{CLCL} maximum.

CMOS single-chip 8-bit microcontrollers

83C750/87C750

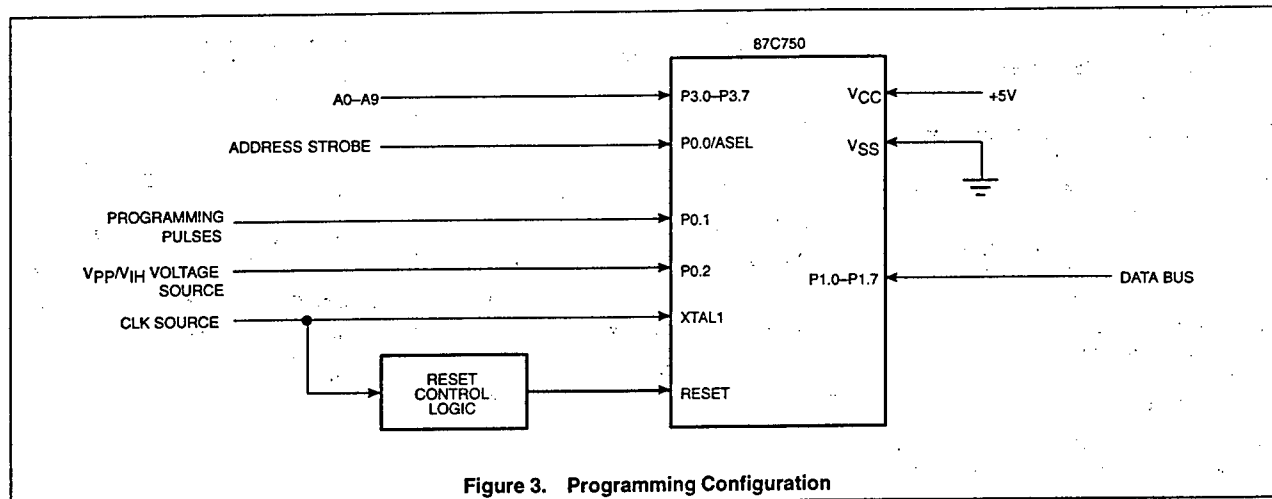


Figure 3. Programming Configuration

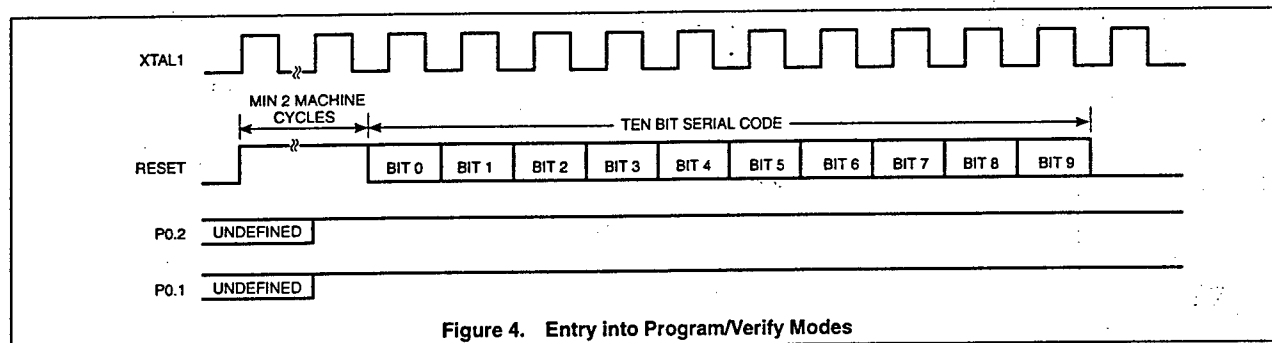


Figure 4. Entry into Program/Verify Modes

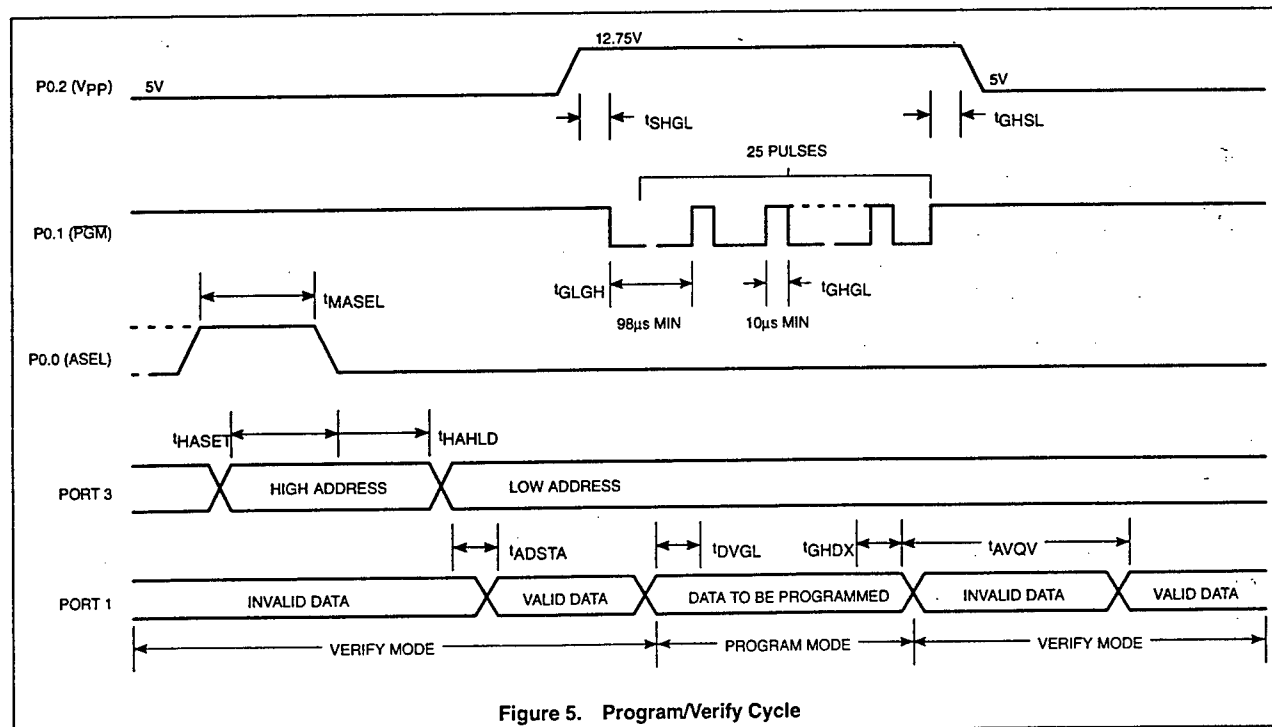


Figure 5. Program/Verify Cycle

80C51 Family

80C51 family programmer's guide
and instruction set

80C51 FAMILY INSTRUCTION SET

Table 7. 80C51 Instruction Set Summary

Interrupt Response Time: Refer to Hardware Description Chapter.			
Instructions that Affect Flag Settings ⁽¹⁾			
Instruction	Flag		
	C	OV	AC
ADD	X	X	X
ADDC	X	X	X
SUBB	X	X	X
MUL	0	X	
DIV	0	X	
DA	X		
RRC	X		
RLC	X		
SETB C	1		
Instruction	Flag		
	C	OV	AC
CLR C	0		
CPL C	X		
ANL C,bit	X		
ANL C,/bit	X		
ANL C,bit	X		
ORL C,bit	X		
MOV C,bit	X		
CJNE	X		

⁽¹⁾Note that operations on SFR byte address 208 or bit addresses 209-215 (i.e., the PSW or bits in the PSW) will also affect flag settings.

Notes on instruction set and addressing modes:

Rn Register R7-R0 of the currently selected Register Bank.

direct 8-bit internal data location's address. This could be an Internal Data RAM location (0-127) or a SFR [i.e., I/O port, control register, status register, etc. (128-255)].

@Ri 8-bit internal data RAM location (0-255) addressed indirectly through register R1 or R0.

#data 8-bit constant included in the instruction.

#data 16 16-bit constant included in the instruction

addr 16 16-bit destination address. Used by LCALL and LJMP. A branch can be anywhere within the 64k-byte Program Memory address space.

addr 11 11-bit destination address. Used by ACALL and AJMP. The branch will be within the same 2k-byte page of program memory as the first byte of the following instruction.

rel Signed (two's complement) 8-bit offset byte. Used by SJMP and all conditional jumps. Range is -128 to +127 bytes relative to first byte of the following instruction.

bit Direct Addressed bit in Internal Data RAM or Special Function Register.

MNEMONIC	DESCRIPTION	BYTE	OSCILLATOR PERIOD
ARITHMETIC OPERATIONS			
ADD A,Rn	Add register to Accumulator	1	12
ADD A,direct	Add direct byte to Accumulator	2	12
ADD A,@Ri	Add indirect RAM to Accumulator	1	12
ADD A,#data	Add immediate data to Accumulator	2	12
ADDC A,Rn	Add register to Accumulator with carry	1	12
ADDC A,direct	Add direct byte to Accumulator with carry	2	12
ADDC A,@Ri	Add indirect RAM to Accumulator with carry	1	12
ADDC A,#data	Add immediate data to ACC with carry	2	12
SUBB A,Rn	Subtract Register from ACC with borrow	1	12
SUBB A,direct	Subtract direct byte from ACC with borrow	2	12
SUBB A,@Ri	Subtract indirect RAM from ACC with borrow	1	12
SUBB A,#data	Subtract immediate data from ACC with borrow	2	12
INC A	Increment Accumulator	1	12
INC Rn	Increment register	1	12

All mnemonics copyrighted © Intel Corporation 1980

80C51 Family

80C51 family programmer's guide
and instruction set

Table 7. 80C51 Instruction Set Summary (Continued)

MNEMONIC		DESCRIPTION	BYTE	OSCILLATOR PERIOD
ARITHMETIC OPERATIONS (Continued)				
INC	direct	Increment direct byte	2	12
INC	@Ri	Increment indirect RAM	1	12
DEC	A	Decrement Accumulator	1	12
DEC	Rn	Decrement Register	1	12
DEC	direct	Decrement direct byte	2	12
DEC	@Ri	Decrement indirect RAM	1	12
INC	DPTR	Increment Data Pointer	1	24
MUL	AB	Multiply A and B	1	48
DIV	AB	Divide A by B	1	48
DA	A	Decimal Adjust Accumulator	1	12
LOGICAL OPERATIONS				
ANL	A,Rn	AND Register to Accumulator	1	12
ANL	A,direct	AND direct byte to Accumulator	2	12
ANL	A,@Ri	AND indirect RAM to Accumulator	1	12
ANL	A,#data	AND immediate data to Accumulator	2	12
ANL	direct,A	AND Accumulator to direct byte	2	12
ANL	direct,#data	AND immediate data to direct byte	3	24
ORL	A,Rn	OR register to Accumulator	1	12
ORL	A,direct	OR direct byte to Accumulator	2	12
ORL	A,@Ri	OR indirect RAM to Accumulator	1	12
ORL	A,#data	OR immediate data to Accumulator	2	12
ORL	direct,A	OR Accumulator to direct byte	2	12
ORL	direct,#data	OR immediate data to direct byte	3	24
XRL	A,Rn	Exclusive-OR register to Accumulator	1	12
XRL	A,direct	Exclusive-OR direct byte to Accumulator	2	12
XRL	A,@Ri	Exclusive-OR indirect RAM to Accumulator	1	12
XRL	A,#data	Exclusive-OR immediate data to Accumulator	2	12
XRL	direct,A	Exclusive-OR Accumulator to direct byte	2	12
XRL	direct,#data	Exclusive-OR immediate data to direct byte	3	24
CLR	A	Clear Accumulator	1	12
CPL	A	Complement Accumulator	1	12
RL	A	Rotate Accumulator left	1	12
RLC	A	Rotate Accumulator left through the carry	1	12
RR	A	Rotate Accumulator right	1	12
RRC	A	Rotate Accumulator right through the carry	1	12
SWAP	A	Swap nibbles within the Accumulator	1	12
DATA TRANSFER				
MOV	A,Rn	Move register to Accumulator	1	12
MOV	A,direct	Move direct byte to Accumulator	2	12
MOV	A,@Ri	Move indirect RAM to Accumulator	1	12

All mnemonics copyrighted © Intel Corporation 1980

Table 7. 80C51 Instruction Set Summary (Continued)

MNEMONIC		DESCRIPTION	BYTE	OSCILLATOR PERIOD
DATA TRANSFER (Continued)				
MOV	A,#data	Move immediate data to Accumulator	2	12
MOV	Rn,A	Move Accumulator to register	1	12
MOV	Rn,direct	Move direct byte to register	2	24
MOV	Rn,#data	Move immediate data to register	2	12
MOV	direct,A	Move Accumulator to direct byte	2	12
MOV	direct,Rn	Move register to direct byte	2	24
MOV	direct,direct	Move direct byte to direct	3	24
MOV	direct,@Ri	Move indirect RAM to direct byte	2	24
MOV	direct,#data	Move immediate data to direct byte	3	24
MOV	@Ri,A	Move Accumulator to indirect RAM	1	12
MOV	@Ri,direct	Move direct byte to indirect RAM	2	24
MOV	@Ri,#data	Move immediate data to indirect RAM	2	12
MOV	DPTR,#data16	Load Data Pointer with a 16-bit constant	3	24
MOVC	A,@A+DPTR	Move Code byte relative to DPTR to Acc	1	24
MOVC	A,@A+PC	Move Code byte relative to PC to Acc	1	24
MOVB	A,@Ri	Move external RAM (8-bit addr) to Acc	1	24
MOVB	A,@DPTR	Move external RAM (16-bit addr) to Acc	1	24
MOVB	A,@Ri,A	Move Acc to external RAM (8-bit addr)	1	24
MOVB	@DPTR,A	Move Acc to external RAM (16-bit addr)	1	24
PUSH	direct	Push direct byte onto stack	2	24
POP	direct	Pop direct byte from stack	2	24
XCH	A,Rn	Exchange register with Accumulator	1	12
XCH	A,direct	Exchange direct byte with Accumulator	2	12
XCH	A,@Ri	Exchange indirect RAM with Accumulator	1	12
XCHD	A,@Ri	Exchange low-order digit indirect RAM with Acc	1	12
BOOLEAN VARIABLE MANIPULATION				
CLR	C	Clear carry	1	12
CLR	bit	Clear direct bit	2	12
SETB	C	Set carry	1	12
SETB	bit	Set direct bit	2	12
CPL	C	Complement carry	1	12
CPL	bit	Complement direct bit	2	12
ANL	C,bit	AND direct bit to carry	2	24
ANL	C,/bit	AND complement of direct bit to carry	2	24
ORL	C,bit	OR direct bit to carry	2	24
ORL	C,/bit	OR complement of direct bit to carry	2	24
MOV	C,bit	Move direct bit to carry	2	12
MOV	bit,C	Move carry to direct bit	2	24
JC	rel	Jump if carry is set	2	24
JNC	rel	Jump if carry not set	2	24

All mnemonics copyrighted © Intel Corporation 1980

80C51 Family

Table 7. 80C51 Instruction Set Summary (Continued)

MNEMONIC	DESCRIPTION	BYTE	OSCILLATOR PERIOD
BOOLEAN VARIABLE MANIPULATION (Continued)			
JB rel	Jump if direct bit is set	2	24
JNB rel	Jump if direct bit is not set	2	24
JBC bit,rel	Jump if direct bit is set and clear bit	3	24
PROGRAM BRANCHING			
ACALL addr11	Absolute subroutine call	2	24
LCALL addr16	Long subroutine call	3	24
RET	Return from subroutine	1	24
RETI	Return from interrupt	1	24
AJMP addr11	Absolute jump	2	24
LJMP addr16	Long jump	3	24
SJMP rel	Short jump (relative addr)	2	24
JMP @A+DPTR	Jump indirect relative to the DPTR	1	24
JZ rel	Jump if Accumulator is zero	2	24
JNZ rel	Jump if Accumulator is not zero	2	24
CJNE A,direct,rel	Compare direct byte to Acc and jump if not equal	3	24
CJNE A,#data,rel	Compare immediate to Acc and jump if not equal	3	24
CJNE Rn,#data,rel	Compare immediate to register and jump if not equal	3	24
CJNE @Ri,#data,rel	Compare immediate to indirect and jump if not equal	3	24
DJNZ Rn,rel	Decrement register and jump if not zero	2	24
DJNZ direct,rel	Decrement direct byte and jump if not zero	3	24
NOP	No operation	1	12

All mnemonics copyrighted © Intel Corporation 1980

5/

TOSHIBA SEMICONDUCTOR
TECHNICAL DATA

TOTX195

2. Electrical and Optical Characteristics. (Ta=25°C, Vcc=5V)

Item	Symbol	Condition	MIN.	TYP.	MAX.	Unit.
Data Rate		NRZ Code (a)	DC	-	10	Mb/s
Transmission Distance		Using APF (a) and TORX194	-	-	50	m
Delay Time(L→H) (a)	t _{PLH}	Using APF and TORX194	-	-	120	ns
Delay Time(H→L) (a)	t _{PHL}	Using APF and TORX194	-	-	120	ns
Pulse Width	Δt _w	Using APF and TORX194	-30	-	30	ns
Distortion (a)		Pulse width 100 ns Repetition 200ns, Cl=10pf	-	-	-	-
Fiber Output Power	P _f	APF 2a, R=1.2kΩ (a)	-11	-	-6	dBm
Peak Emission Wavelength	λ _p		-	670	-	nm
Current Consumption	I _{cc}	R=1.2kΩ	-	35	55	mA
High Level Input Voltage	V _{IH}		2.0	-	-	V
Low Level Input Voltage	V _{IL}		-	-	0.8	V
High Level Input Current	I _{IH}		-	-	20	μA
Low Level Input Current	I _{IL}		-	-	-0.4	mA

- (a) LED is on when input signal is high level, it is off when low level.
(a) All Plastic fiber(980/1000μm) with polished surface.
(a) Between input of TOTX195 and output of TORX194.
(a) Measured with a standard optical fiber with fiber optic connectors.
Valued by peak.

TOTX195-2
1990-10-30
TOSHIBA CORP

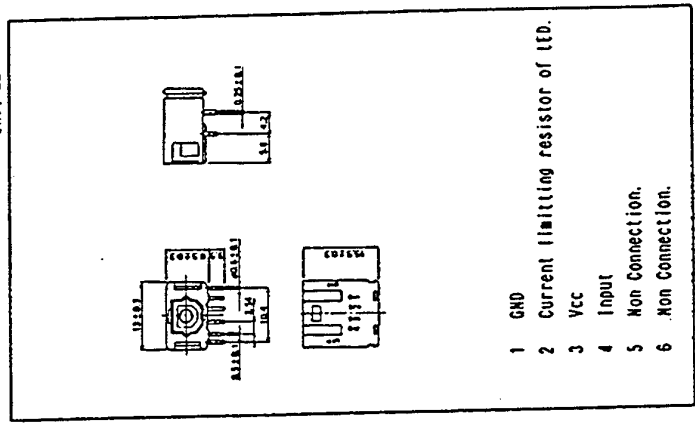
4/15

Fiber Optic Transmitting Module
TOTX195

TOSHIBA SEMICONDUCTOR
TECHNICAL DATA

Fiber Optic Transmitting Module for
Simplex Digital signal transmission.
- Data rate : DC to 10 Mb/s(NRZ code).
- Transmission distance : Up to 50 m.
- TTL interface.
- LED is driven by differential circuit.

Unit mm



S, Inc. Code: Vnnel
92315 SEVRS CDEX
RAYNICK BALLY
Tel: 33(1) 45 23 24 25
Fax: 33(1) 45 07 71 91
Telex: 269 746 805 exlusse
10 8746 - 16311C 4



1. Absolute Maximum Ratings. (Ta=25°C)

Item	Symbol	Rating	Unit
Storage Temperature	T _{stg}	-40 to 85	°C
Operating Temperature	T _{op}	-40 to 85	°C
Supply Voltage	V _{cc}	-0.5 to 7	V
Input Voltage	V _{in}	-0.5 to V _{cc} +0.5	V
Soldering Temperature	T _{sol}	260 (a)	°C

Note (a) Soldering time ≤ 3 seconds.

© The information contained herein is presented only as a guide for the applications of our products. No responsibility is assumed by TOSHIBA CORPORATION for any damages or losses resulting from the use of the information or other rights of the third parties which may result from the use. No license is granted by implication or otherwise under any intellectual property or other rights of TOSHIBA CORPORATION or others.

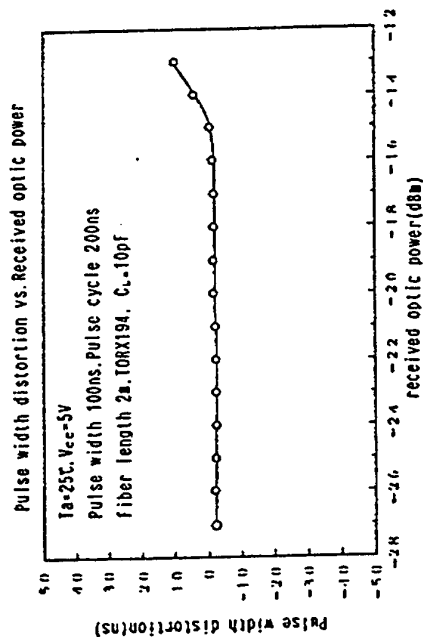
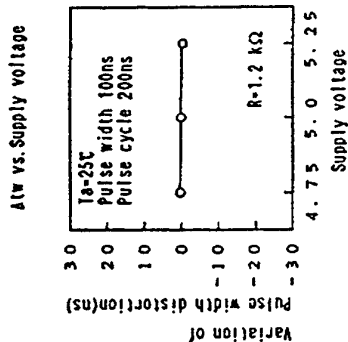
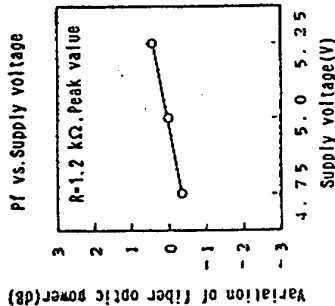
TOSHIBA CORPORATION

4/15

TOSHIBA SEMICONDUCTOR
TECHNICAL DATA

TOTX195

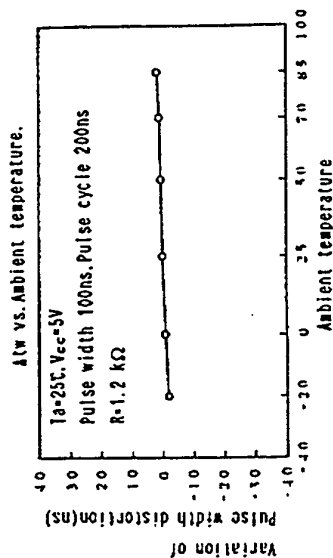
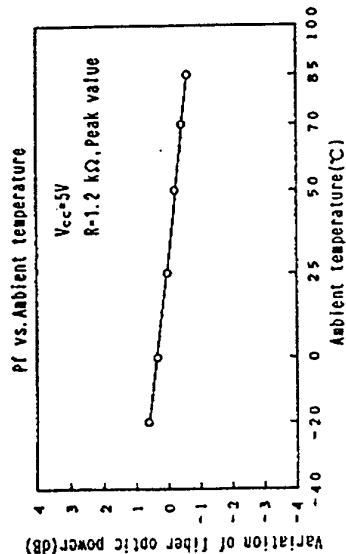
Example of Typical Characteristics



TOTX195-3
1990-10-30
TOSHIBA CORPORATION

TOSHIBA SEMICONDUCTOR
TECHNICAL DATA

TOTX195



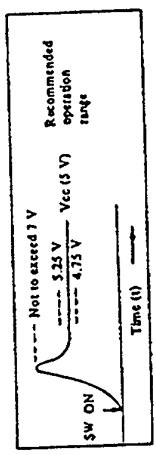
TOTX195-4
1990-10-30
TOSHIBA CORPORATION

TOSHIBA SEMICONDUCTOR TECHNICAL DATA

TOTX195

5. Precautions for operation.

- (1) The absolute maximum ratings shows the limits, which must not be exceeded even momentarily regardless of the external condition. Operation beyond the limit of the absolute maximum ratings may cause failure of the device.
- (2) Please be sure to solder Pins No. 5 and No. 6 of TO1X195 to PC board.
- (3) Power supply voltage.



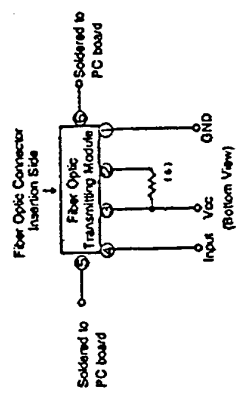
- (4) Do not use acid or alkaline soldering flux cleaner solvent. Please be careful not to inject the solvent into module through the fiber optic connector hole. If some solvent happens to be injected into the module, wipe off with a cotton ball. The recommended cleaner solvent is thichroethane.
- (5) When not using the module, always provide an attached protective cap to it.

TO1X195-6
1990-10-30
TOSHIBA CORP

TOSHIBA SEMICONDUCTOR TECHNICAL DATA

TOTX195

3. Connection Method



Note: Select a resistor value as follows:

Transmission Distance (m)	Resistor (Ω)
0.2 to 10	17.8 k
10 to 30	6.2 k
30 to 50	1.2 k

4. Applicable optical fiber with fiber optic connectors.

TOCP100-...HB, TOCP155-...HB, TOCP100P-...HB, TOCP155P-...HB.

TO1X195-5
1990-10-30
TOSHIBA CORPORATE

OSHIBA

SEMICONDUCTOR

TECHNICAL DATA

Fiber Optic Receiving Module for
Simplex Digital signal transmission.

- Data rate : DC to 10 M b/s (NRZ code).
- Transmission distance
: Up to 50 m (APF).
: Up to 1000 m (PCF).
- TTL Interface.
- ATC (Automatic Threshold Control)

Circuit is used for stabilized output
at a wide range of optical power level.



5, rue Curie, Vernier
92315 SEVRES CEDEX
FRANÇOIS BAILLY
Tel : 33 (1) 46 23 24 25
Fax : 33 (1) 45 07 21 91
Telex : 269740 socoedisso
H 6 B/46 FRANCE

1. Absolute Maximum Ratings (Ta=25 °C)

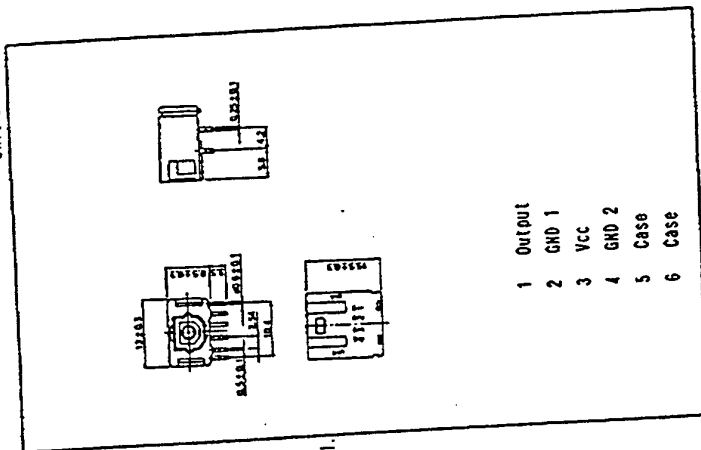
Item	Symbol	Rating	Unit
Storage Temperature	T _{STG}	-40 to 85	°C
Operating Temperature	T _{OPR}	-40 to 85	°C
Supply Voltage	V _{CC}	-0.5 to 7	V
Low Level Output Current	I _{OL}	20	mA
High Level Output Current	I _{IH}	-1	mA
Soldering Temperature	T _{STG}	260 (1)	°C

Note (1) Soldering time ≤ 3 seconds.

© The information contained herein is presented only as a guide for the applications of our products.
No responsibility is assumed by TOSHIBA CORPORATION for any infringement of intellectual property
or other rights of the third parties which may result from its use. No license is granted by implication
or otherwise under any intellectual property or other rights of TOSHIBA CORPORATION or others.

Fiber Optic Receiving Module
TORX194

Unit mm



- 1 Output
- 2 GND 1
- 3 Vcc
- 4 GND 2
- 5 Case
- 6 Case

TOSHIBA

SEMICONDUCTOR

TECHNICAL DATA

TORX194

2. Electrical and Optical Characteristics (Ta=25°C, Vcc=5V)

Item	Symbol	Condition	MIN.	TYP.	MAX.	Unit
Date	NRZ code (1)	Using APF (1), 10TX195	DC	-	10	MB/s
Transmission Distance		Using PCF (1), 10TX194	0.2	-	50	m
Delay time (t _{PL})		Fiber length 2m.	-	-	1000	ns
Delay time (t _{PH})		Fiber length 2m.	-	-	120	ns
Pulse Width	Δ t _W	Pulse width 100ns	-30	-	30	ns
Distortion (1)		Pulse cycle 200ns	-	-	-	-
		C _L 10pF	-	-	-	-
Maximum Receivable Power (1)	P _{MAX}	10Mb/s, APF, 10TX195	-14	-	-	dBm
		10Mb/s, PCF, 10TX194	-18	-	-	dBm
Minimum Receivable Power (1)	P _{MIN}	10Mb/s, APF, 10TX195	-	-	-27	dBm
		10Mb/s, PCF, 10TX194	-	-	-29	dBm
Current Consumption	I _{CC}		-	22	40	mA
High Level	V _{OH}		2.7	-	-	V
Output Voltage			-	-	-	-
Low Level	V _{OL}		-	-	0.4	V
Output Voltage			-	-	-	-

Note (1) The duty factor must be such as kept 25 to 75 %.

High level output when optical flux is received. Low level output when optical flux is not received.

(1) All Plastic fiber (980/1000 μm) with polished surface.

(1) Plastic clad silica fiber (200/300 μm) with polished surface.

(1) Between input of a fiber optic transmission module and output of TORX194.

(1) BER ≤ 10⁻⁹, valued by peak.

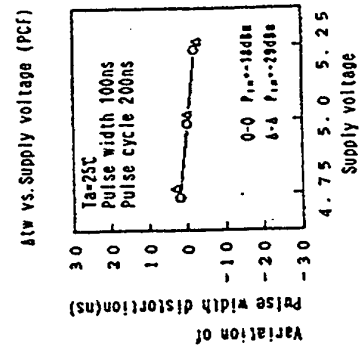
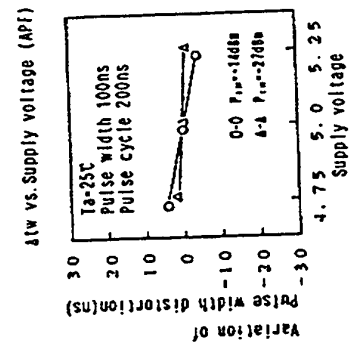
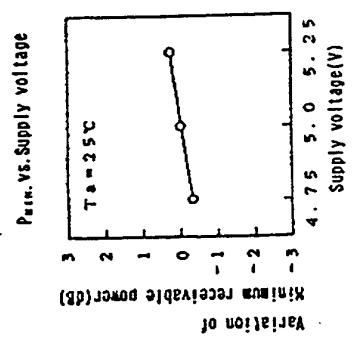
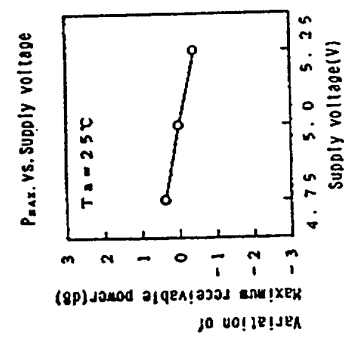
TORX194-2
1990-10-30
TOSHIBA CORPORATION

10/15

TOSHIBA SEMICONDUCTOR
TECHNICAL DATA

TORX194

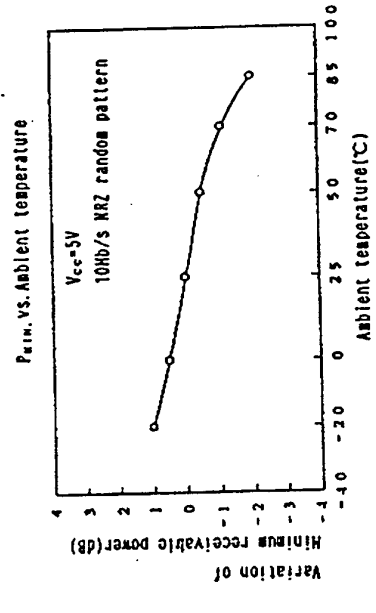
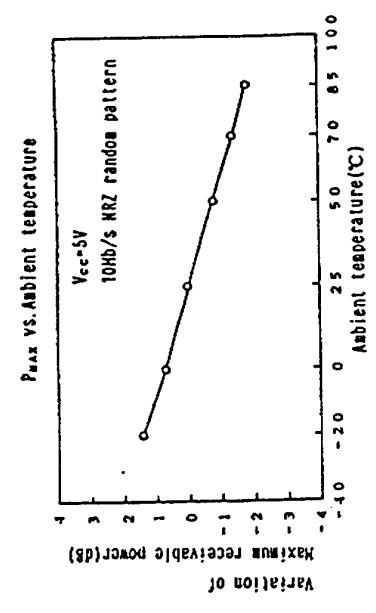
Example of Typical Characteristics



11/

TOSHIBA SEMICONDUCTOR
TECHNICAL DATA

TORX194



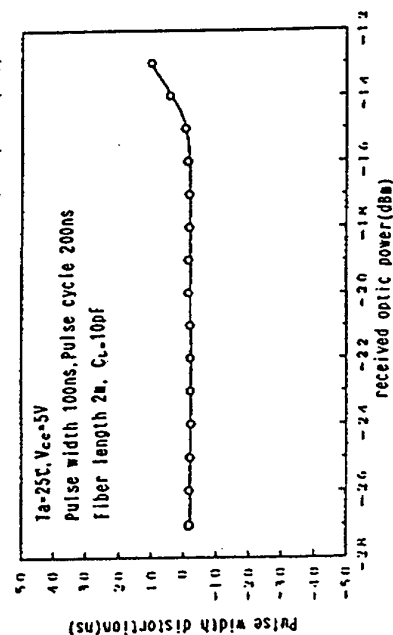
TORX194-3
1990-10-30
TOSHIBA CORPORATION

TORX194-4
1990-10-30
TOSHIBA CORPORATION

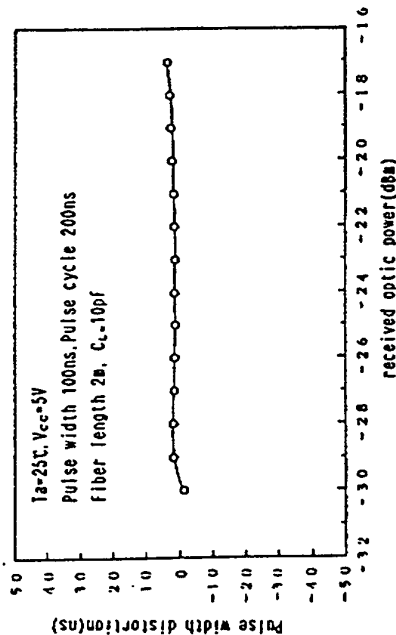
TOSHIBA
SEMICONDUCTOR
TECHNICAL DATA

TORX194

Pulse width distortion vs. Received optic power (APF)



Pulse width distortion vs. Received optic power (PCF)

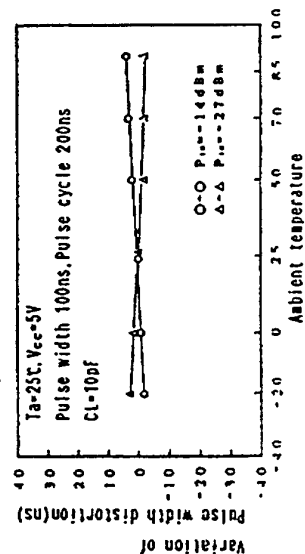


TORX194-5
1990-10-30
TOSHIBA CORPORATION

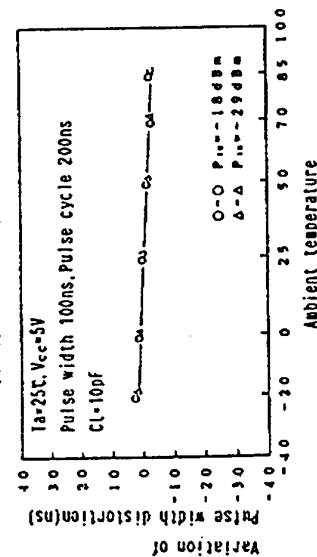
TOSHIBA
SEMICONDUCTOR
TECHNICAL DATA

TORX194

ΔIW vs. Ambient temperature. (APF)

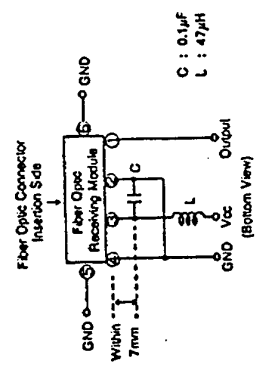


ΔIW vs. Ambient temperature. (PCF)



TORX194-6
1990-10-30
TOSHIBA CORPORATION

3. Connection Method



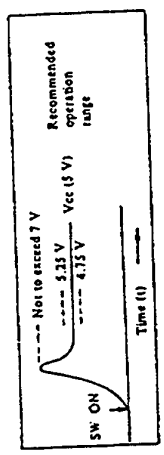
4. Applicable optic connal fiber with fiber optic connectors.

- TOCP100--HB, TOCP155--HB, TOCP100P--HB, TOCP155P--HB (APF).
- TOCP1000--HB, TOCP1500--HB, TOCP1010--HB, TOCP1510--HB, TOCP1560--HB
- TOCP100X--HB, TOCP150X--HB, TOCP101X--HB, TOCP151X--HB, TOCP156X--HB (PCF).

5. Precautions for operation

- (1) The absolute maximum ratings show the limits, which must not be exceeded even momentarily regardless of the external condition. Operation beyond the limit of the absolute maximum rating may cause failure of the device.
- (2) Pins No. 5 and No. 6 of TORX194 are ground pins of housing. The housing is made of conductive plastic for shielding purpose. Please be sure to ground these pins for efficient shielding.
- (3) Additional precaution is necessary to ensure that conductive housing dose not touch other potential patterns.

(4) Power supply voltage



- (5) Do not use acid or alkaline soldering flux cleaner solvent. Please be careful not inject the solvent into module through the fiber optic connector hole. If some solvent happens to be injected into the module, wipe it off with a cotton ball. The recommended cleaner solvent is thichrolethane.
- (6) When not using the module, always provide an attached protective cap to it.

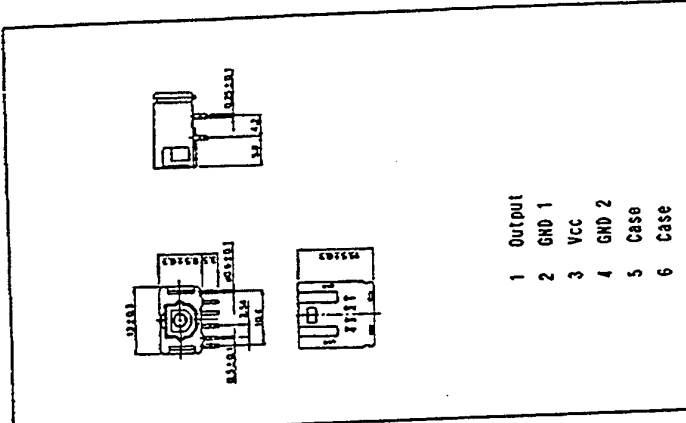
1/5

Fiber Optic Receiving Module
TORX194

OSHIBA SEMICONDUCTOR
TECHNICAL DATA

- Fiber Optic Receiving Module for
Simplex Digital signal transmission.
- Data rate : DC to 10 M b/s(MRZ code).
 - Transmission distance
 - : Up to 50 m(APF).
 - : Up to 1000 m(PCF).
 - TTL Interface.
 - AIC(Automatic Threshold Control)
- Circuit is used for stabilized output
at a wide range of optical power level.

Unit mm



5, rue Claude Vernier
92315 SURESNES CEDEX
Yvanick BAILLY
Tel : 33 (1) 46 23 24 25
Fax : 33 (1) 45 07 21 91
Telex : 269 746 sous adresse
Hx 8746 TEKLE C

1. Absolute Maximum Ratings (Ta=25 °C)

Item	Symbol	Rating	Unit
Storage Temperature	T _{STG}	-40 to 85	°C
Operating Temperature	T _{OPR}	-40 to 85	°C
Supply Voltage	V _{CC}	-0.5 to 7	V
Low Level Output Current	I _{OL}	20	mA
High Level Output Current	I _{IH}	-1	mA
Soldering Temperature	T _{STG}	260 (1)	°C

Note (1) Soldering time ≤ 3 seconds.

© The information contained herein is presented only as a guide for the applications of our products.
No responsibility is assumed by TOSHIBA CORPORATION for any infringement of intellectual property
or other rights of the third parties which may result from its use. No license is granted by implication
or otherwise under any intellectual property or other rights of TOSHIBA CORPORATION or others.

ATTN: 1

TOSHIBA CORPORATION

1/1

TOSHIBA SEMICONDUCTOR
TECHNICAL DATA

TORX194

2. Electrical and Optical Characteristics (Ta=25°C, Vcc=5V)

Item	Symbol	Condition	MIN.	TYP.	MAX.	Unit
Date Date		MRZ code (1)	DC	-	10	Mb/s
Transmission		Using APF (1), 10IX195	0.2	-	50	M
Distance		Using PCF (1), 10IX194	0.2	-	1000	M
Delay time (t _{HL} -t _{HL})	t _{HL}	Fiber length 2m.	-	-	120	ns
Delay time (t _{HL} -t _{HL})	t _{HL}	Fiber length 2m.	-	-	120	ns
Pulse Width	Δ t w	Pulse width 100ns	-30	-	30	ns
Distortion (1)		Pulse cycle 200ns	-	-	-	-
		C _L =10pF	-	-	-	-
Maximum Receivable Power (1)	P _{MAX}	10Mb/s. APF, 10IX195	-14	-	-	dBm
Minimum Receivable Power (1)	P _{MIN}	10Mb/s. PCF, 10IX194	-18	-	-	dBm
		10Mb/s. APF, 10IX195	-	-	-27	dBm
		10Mb/s. PCF, 10IX194	-	-	-29	dBm
Current Consumption	I _{CC}		-	22	40	mA
High Level	V _{OH}		2.7	-	-	V
Output Voltage	V _{OL}		-	-	-	V
Low Level			-	-	-	V
Output Voltage			-	-	-	V

Note (1) The duty factor must be such as kept 25 to 75 %.

High level output when optical flux is received. Low level output when optical flux is not received.

(1) All Plastic fiber (980/1000 μm) with polished surface.

(1) Plastic clad silica fiber (200/300 μm) with polished surface.

(1) Between input of a fiber optic transmission module and output of TORX194.

(1) BER ≤ 10⁻⁷, valued by peak.

TORX194-2
1990-10-30
TOSHIBA CORP

ADVANCE INFORMATION

All information in this data sheet is preliminary and subject to change.

8/93

MAXIM

Precision, Quad, SPDT, CMOS Analog Switch

General Description

The MAX333A is a precision, quad, single-pole double-throw (SPDT) analog switch. The four independent switches operate with bipolar supplies ranging from $\pm 4.5\text{V}$ to $\pm 20\text{V}$, or with a single-ended supply between $+10\text{V}$ and $+30\text{V}$. The MAX333A offers low on resistance (less than 35Ω), guaranteed to match within 2Ω between channels and to remain flat over the full analog signal range ($\Delta 3\Omega$ max). It also offers break-before-make switching (10ns typical), with turn-off times less than 145ns and turn-on times less than 175ns. The MAX333A is ideal for portable operation since quiescent current runs less than $1\mu\text{A}$ with all inputs high or low.

This monolithic, quad switch is fabricated with Maxim's new improved silicon-gate process. Design improvements guarantee extremely low charge injection (10pC), low power consumption (35 μW), and electrostatic discharge (ESD) greater than 2000V.

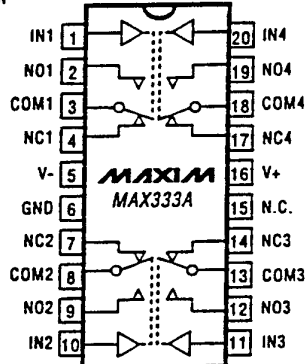
Logic inputs are TTL- and CMOS-compatible and guaranteed over a $+0.8\text{V}$ to $+2.4\text{V}$ range, regardless of supply voltage. Logic inputs and switched analog signals can range anywhere between the supply voltages without damage. This upgraded part is a replacement for a DG211/DG212 pair when used as a quad SPDT switch, or two DG403 dual SPDT switches.

Applications

Test Equipment
Communications Systems
PBX, PABX
Heads-Up Displays
Portable Instruments

Pin Configuration

TOP VIEW



DIP/SO

SWITCHES ARE SHOWN WITH LOGIC "0" INPUT

Features

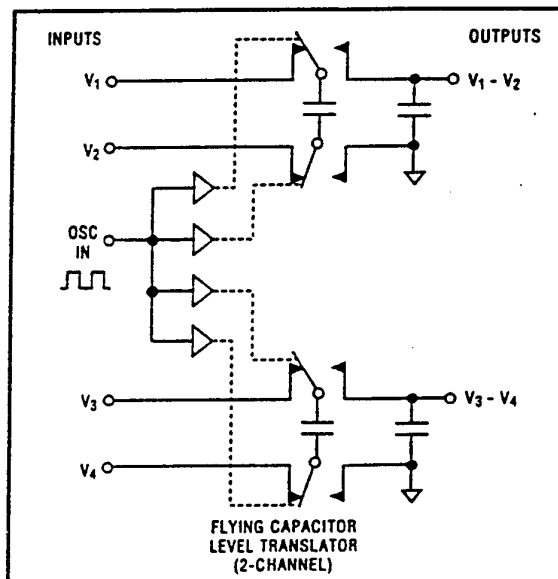
- ◆ Upgraded Replacement for a DG211/DG212 Pair or Two DG403s
- ◆ Low On Resistance $< 22\Omega$ Typical (35 Ω Max)
- ◆ Guaranteed Matched On Resistance Between Channels $< 2\Omega$
- ◆ Guaranteed Flat On Resistance over Full Analog Signal Range $\Delta 3\Omega$ Max
- ◆ Guaranteed Charge Injection $< 10\text{pC}$
- ◆ Guaranteed Off-Channel Leakage $< 6\text{nA}$ at $+85^\circ\text{C}$
- ◆ ESD Guaranteed $> 2000\text{V}$ per Method 3015.7
- ◆ Single-Supply Operation ($+10\text{V}$ to $+30\text{V}$)
Bipolar-Supply Operation ($\pm 4.5\text{V}$ to $\pm 20\text{V}$)
- ◆ TTL-/CMOS-Logic Compatibility
- ◆ Rail-to-Rail Analog Signal Handling Capability

Ordering Information

PART	TEMP. RANGE	PIN-PACKAGE
MAX333ACPP	0°C to $+70^\circ\text{C}$	20 Plastic DIP
MAX333ACWP	0°C to $+70^\circ\text{C}$	20 Wide SO
MAX333AC/D	0°C to $+70^\circ\text{C}$	Dice*
MAX333AEPP	-40°C to $+85^\circ\text{C}$	20 Plastic DIP
MAX333AEWP	-40°C to $+85^\circ\text{C}$	20 Wide SO
MAX333AMJP	-55°C to $+125^\circ\text{C}$	20 CERDIP

* Contact factory for dice specifications.

Typical Operating Circuit



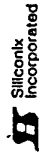
MAX333A

1

MAXIM

Maxim Integrated Products 1-17

Call toll free 1-800-998-8800 for free samples or literature.



5d

DG444/445

Monolithic Quad SPST

CMOS Analog Switches

FEATURES

- ± 15 Volt Input Range
- ON Resistance < 80 Ω
- Fast Switching Action
tON < 160 ns
tOFF < 80 ns
- TTL, CMOS Compatible
- DG211/DG212 Upgrades
- ESDS Protection > ± 4000 V

BENEFITS

- Wide Dynamic Range
- Low Signal Errors and Distortion
- Simple Interfacing
- Audio and Video Switching
- Communication Systems
- Battery Operated Systems

APPLICATIONS

- Sample and Hold circuits
- Data Acquisition
- Automatic Test Equipment
- Audio and Video Switching
- Communication Systems
- Battery Operated Systems

DESCRIPTION

The DG444 series of monolithic quad analog switches was designed to provide high speed, low error switching of analog signals. Combining low power (< 35 microwatts) with high speed (tON < 160 ns), the DG444/445 is ideally suited for upgrading DG211/DG212 sockets. Charge Injection has been minimized on the drain for use in sample-and-hold circuits.

To achieve high-voltage ratings and superior switching performance, the DG444 series was built on Siliconix's high-voltage silicon-gate process. An epitaxial layer prevents latchup.

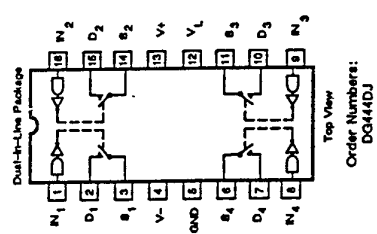
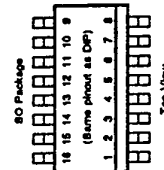
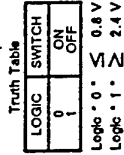
Each switch conducts equally well in both directions when ON, and blocks up to 30 volts peak-to-peak when OFF. ON resistance is very flat over the full ± 15 V analog range, rivaling JFET performance without the inherent dynamic range limitation.

The two devices in this series are differentiated by the type of switch action as shown in the functional block diagrams for each. Packaging options include the 16-pin plastic and small outline. The performance grade for this series is the Industrial, D suffix (-40 to 85°C) temperature range.

FUNCTIONAL BLOCK DIAGRAM AND PIN CONFIGURATION

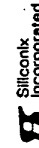
DG444

Four SPST Switches per Package



5-274

Preliminary

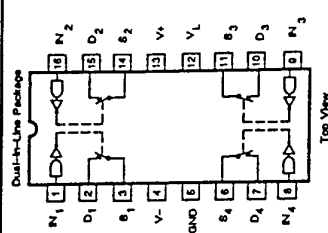
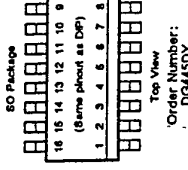
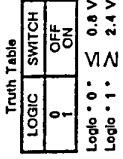


DG444/445

FUNCTIONAL BLOCK DIAGRAM PIN CONFIGURATION

DG445

Four SPST Switches per Package



ABSOLUTE MAXIMUM RATINGS

Voltages Referenced to V ₋	Operating Temperature (D Suffix)	Power Dissipation (Package*)
V ₊ 44 V	 -40 to 85°C	16-Pin Plastic DIP** 450 mW
QND 25 V		16-Pin SO** 600 mW
V _L (GND - 0.3 V) to 44 V		
Digital Inputs V _D , V _O (V ₋ minus 2 V) to (V ₊ plus 2 V)		
Current (Any Terminal) continuous 30 mA		
Current (S or D) Pulsed 1 ms, 10% duty 100 mA		
Storage Temperature (D Suffix) -65 to 125°C		

ELECTRICAL CHARACTERISTICS*

ELECTRONIC CHARACTERISTICS		SYMBOL	Test Conditions Unless Otherwise Specified: V _I = 15 V V ₋ = -15 V V _L = 5 V GND = 0 V V _N = 2.4, 0.8 V *	LIMITS			UNIT	
PARAMETER	SWITCH			TEMP	TYP ^d	D		
						1-25°C		2-85°C
SWITCH								
Analog Signal Range °	V _{ANALOG}					-15 15	V	
Drain-Source ON Resistance	r _{DS(ON)}		I _g = -10 mA, V _D = ± 8.5 V V _L = 13.5 V, V ₋ = -13.5 V	1, 2	1.3	80 100	Ω	
Switch OFF Leakage Current	I _{S(OFF)}		V _I = 18.5 V, V ₋ = -18.5 V V _D = ± 15.5 V, V _g = ± 15.5 V	1	1	-0.25 0.25	nA	
	I _{D(OFF)}			2	2	-20 20		
Channel ON Leakage Current	I _{D(ON)} + I _{S(ON)}		V _I = 18.5 V, V ₋ = -18.5 V V _g = V _D = ± 15.5 V	1	1	-0.25 0.25	nA	
				2	2	-20 20		
						-0.4 0.4	nA	
						-40 -40	nA	

Preliminary

5-275

ELECTRICAL CHARACTERISTICS ^a					
PARAMETER	SYMBOL	Test Conditions Unless Otherwise Specified: $V_i = 15\text{ V}$ $V_o = -15\text{ V}$ $V_L = 5\text{ V}$ $V_N = 0\text{ V}$ $V_N = 2.4, 0.5\text{ V}^*$	LIMITS		
			1-25°C	2-85°C	D SUFFIX -10 to 85°C
INPUT					
Input Current with V_{IN} LOW	I_L	V_N Under Test = 0.8 V All Other = 2.4 V	1.2	-0.5	0.5
Input Current with V_{IN} HIGH	I_H	V_N Under Test = 2.4 V All Other = 0.5 V	1.2	-0.5	0.5
DYNAMIC					
Turn-ON Time	t_{ON}	$R_L = 1\text{ k}\Omega$, $C_L = 5\text{ pF}$ See Figure 1 $V_o = \pm 10\text{ V}$	1		160
Turn-OFF Time	t_{OFF}		1		80
Charge Injection ^c	Q	$C_L = 10\text{ nF}$, $V_o = 0\text{ V}$ $V_{pen} = 0\text{ V}$, $R_{pen} = 0\Omega$	1		-10 10
SUPPLY					
Positive Supply Current	I_+		1		1
Negative Supply Current	I_-		1		-1
Logic Supply Current	I_L	$V_i = 15.5\text{ V}$, $V_o = -15.5\text{ V}$ $V_N = 0\text{ or }5\text{ V}$	1		1
Ground Current	I_{GND}		1		-1

Positive Supply Current	I_+		1		1
Negative Supply Current	I_-		1		-1
Logic Supply Current	I_L	$V_i = 15.5\text{ V}$, $V_o = -15.5\text{ V}$ $V_N = 0\text{ or }5\text{ V}$	1		1
Ground Current	I_{GND}		1		-1

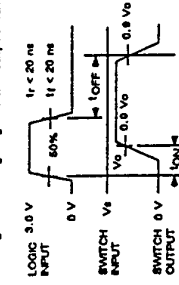
NOTES:
 a. There is no PROCESS OPTION FLOWCHART for additional information.
 b. The logic supply current is the most negative value in a minimum and the most positive a maximum, is used in this data sheet.
 c. Guaranteed by design, not subject to production test.
 d. Typical values are for DESIGN AID ONLY, not guaranteed nor subject to production testing.
 e. V_N = Input voltage to perform proper function.

ELECTRICAL CHARACTERISTICS ^a					
PARAMETER	SYMBOL	Test Conditions Unless Otherwise Specified: $V_i = 12\text{ V}$ $V_o = 5\text{ V}$ $V_L = 5\text{ V}$ $V_N = 2.4, 0.5\text{ V}^*$	LIMITS		
			1-25°C	2-85°C	D SUFFIX -10 to 85°C
DYNAMIC					
Turn-ON Time	t_{ON}	$R_L = 1\text{ k}\Omega$, $C_L = 35\text{ pF}$ See Figure 1 $V_o = 8\text{ V}$	1		400
Turn-OFF Time	t_{OFF}		1		200
Charge Injection ^c	Q	$C_L = 10\text{ nF}$, $V_i = 6.25\text{ V}$ $V_{pen} = 6.6\text{ V}$, $R_{pen} = 0\Omega$ $V_o = 13.2\text{ V}$	1		-40 40
SUPPLY					
Positive Supply Current	I_+	$V_i = 13.2\text{ V}$ $V_N = 0\text{ or }5\text{ V}$	1		1
Negative Supply Current	I_-	$V_N = 0\text{ or }5\text{ V}$	1		-1
Logic Supply Current	I_L	$V_L = 6.25\text{ V}$ $V_N = 0\text{ or }5\text{ V}$	1		1
Ground Current	I_{GND}	$V_N = 0\text{ or }5\text{ V}$	1		-1

NOTES:
 a. There is no PROCESS OPTION FLOWCHART for additional information.
 b. The logic supply current is the most negative value in a minimum and the most positive a maximum, is used in this data sheet.
 c. Guaranteed by design, not subject to production test.
 d. Typical values are for DESIGN AID ONLY, not guaranteed nor subject to production testing.
 e. V_N = Input voltage to perform proper function.

SWITCHING TIME TEST CIRCUIT

V_O is the steady state output with the switch ON. Feedthrough via switch capacitance may result in spikes at the leading and trailing edges of the output waveform.

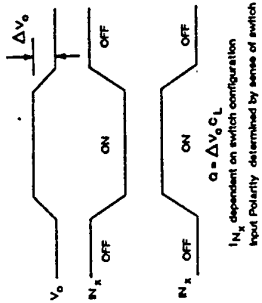
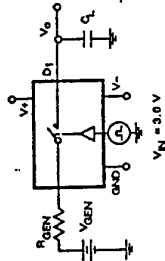


NOTE: Logic input waveform is inverted for test purposes that have the opposite logic sense.

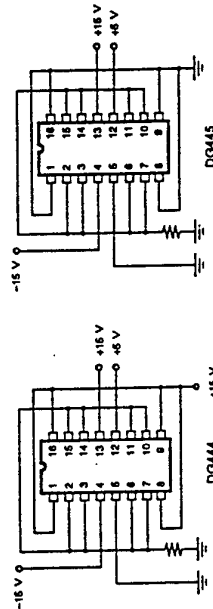
Repeat test for Ch 2, 3, 4
 For load conditions, See Electrical Characteristics
 C_L (Include fixture and stray capacitance)
 $V_O = V_{DD} - \frac{R_L}{R_L + r_{DS(ON)}}$

ELECTRICAL CHARACTERISTICS ^a					
PARAMETER	SYMBOL	Test Conditions Unless Otherwise Specified: $V_i = 12\text{ V}$ $V_o = 5\text{ V}$ $V_L = 5\text{ V}$ $V_N = 2.4, 0.5\text{ V}^*$	LIMITS		
			1-25°C	2-85°C	D SUFFIX -10 to 85°C
SWITCH					
Analog Signal Range ^c	V_{ANALOG}			0	12
Drift-Source ON Resistance	$r_{DS(ON)}$	$I_D = -10\text{ mA}$, $V_o = 3\text{ V}$, $V_L = 5\text{ V}$ $V_i = 10.5\text{ V}$, $V_L = 5.25\text{ V}$	1		160
			2,3		200

CHARGE INJECTION TEST CIRCUIT

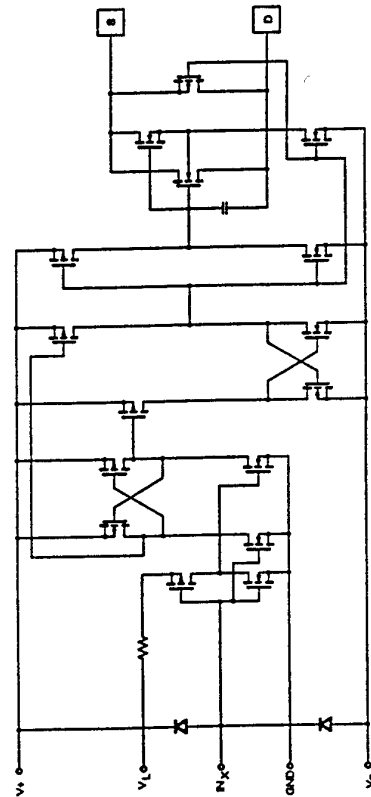


BURN-IN CIRCUITS

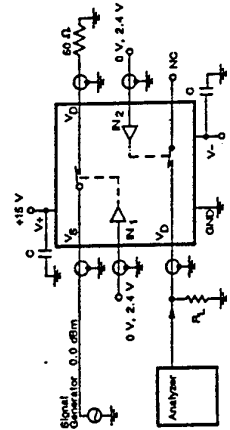


Note: All Resistors are 10 kΩ unless otherwise specified

SCHEMATIC DIAGRAM (TYPICAL CHANNEL)

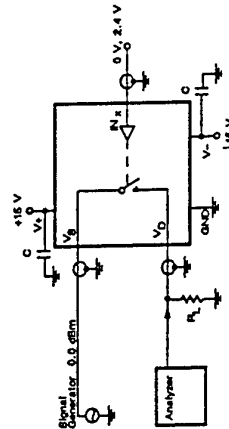


CROSSTALK TEST CIRCUIT



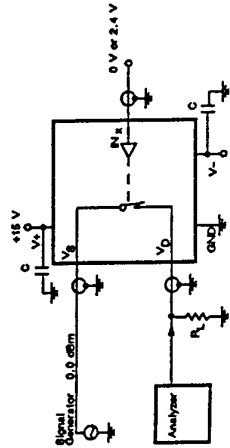
FREQUENCY TESTED	SIGNAL GENERATOR	ANALYZER
100 Hz to 13 MHz	HP3308B Automatic Synthesizer	HP3571A Tracking Spectrum Analyzer

OFF ISOLATION TEST CIRCUIT



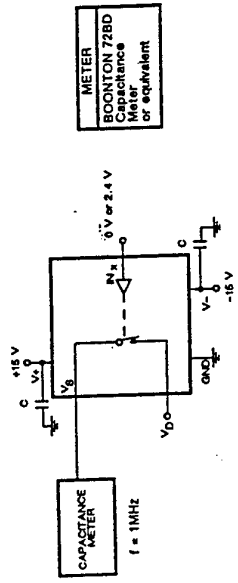
FREQUENCY TESTED	SIGNAL GENERATOR	ANALYZER
100 Hz to 13 MHz	HP3308B Automatic Synthesizer	HP3571A Tracking Spectrum Analyzer

INSERTION LOSS TEST CIRCUIT

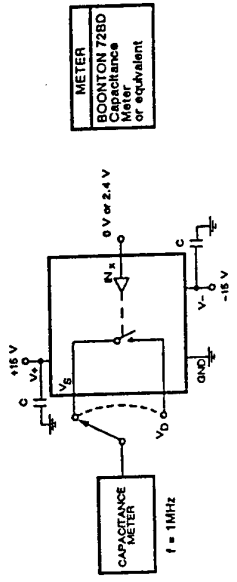


FREQUENCY TESTED	SIGNAL GENERATOR	ANALYZER
100 Hz to 13 MHz	HP3308B Automatic Synthesizer	HP3571A Tracking Spectrum Analyzer

SOURCE + DRAIN ON CAPACITANCE



SOURCE + DRAIN OFF CAPACITANCE



PIN DESCRIPTION

SYMBOL	DESCRIPTION
S	Analog Channel Input or Output
D	Analog Channel Output or Input
IN	Logic Control Input
V+	Positive Supply Voltage
V-	Negative Supply Voltage
GND	Digital Ground
V _L	Logic Supply Voltage

APPLICATIONS

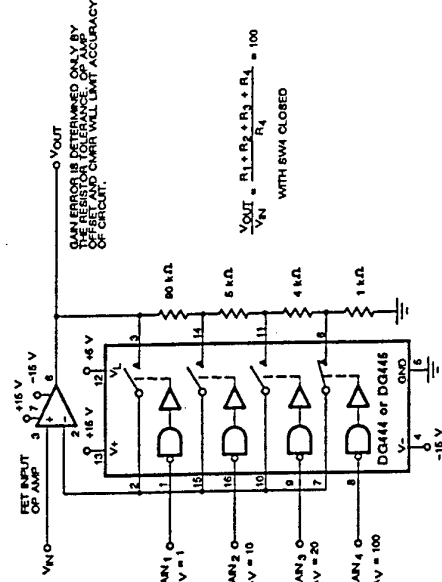


Figure 6. Precision-Weighted Resistor Programmable-Gain Amplifier

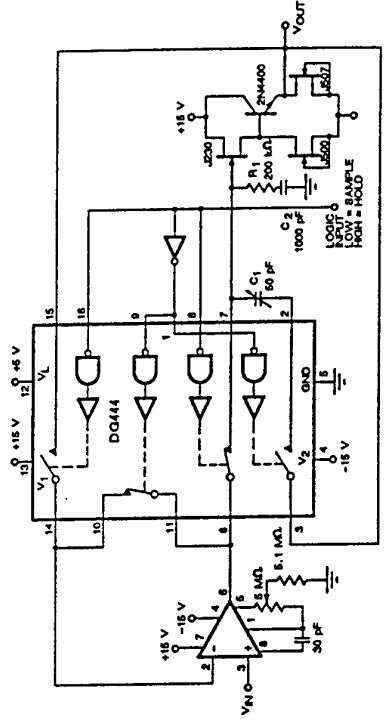


Figure 7. Precision Sample-and-Hold



LM199/LM299/LM399 Precision Reference

General Description

The LM199/LM299/LM399 are precision, temperature-stabilized monolithic zeners offering temperature coefficients a factor of ten better than high quality reference zeners. Constructed on a single monolithic chip is a temperature stabilizer circuit and an active reference zener. The active circuitry reduces the dynamic impedance of the zener to about 0.5Ω and allows the zener to operate over 0.5 mA to 10 mA current range with essentially no change in voltage or temperature coefficient. Further, a new subsurface zener structure gives low noise and excellent long term stability compared to ordinary monolithic zeners. The package is supplied with a thermal shield to minimize heater power and improve temperature regulation.

The LM199 series references are exceptionally easy to use and free of the problems that are often experienced with ordinary zeners. There is virtually no hysteresis in reference voltage with temperature cycling. Also, the LM199 is free of voltage shifts due to stress on the leads. Finally, since the unit is temperature stabilized, warm up time is fast.

The LM199 can be used in almost any application in place of ordinary zeners with improved performance. Some ideal applications are analog to digital converters,

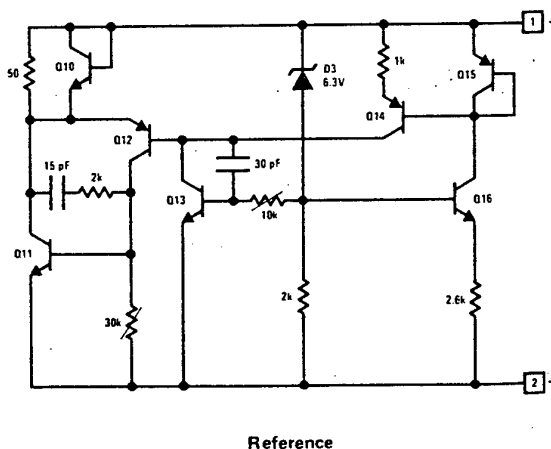
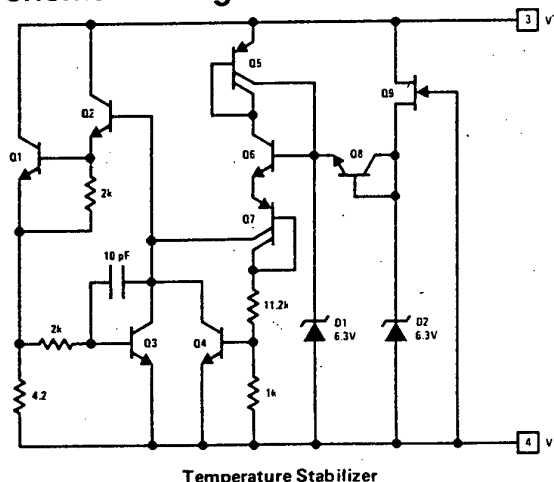
calibration standards, precision voltage or current sources or precision power supplies. Further in many cases the LM199 can replace references in existing equipment with a minimum of wiring changes.

The LM199 series devices are packaged in a standard hermetic TO-46 package inside a thermal shield. The LM199 is rated for operation from -55°C to $+125^\circ\text{C}$ while the LM299 is rated for operation from -25°C to $+85^\circ\text{C}$ and the LM399 is rated from 0°C to $+70^\circ\text{C}$.

Features

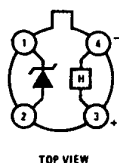
- Guaranteed $0.0001\%/^\circ\text{C}$ temperature coefficient
- Low dynamic impedance — 0.5Ω
- Initial tolerance on breakdown voltage — 2%
- Sharp breakdown at $400\mu\text{A}$
- Wide operating current — $500\mu\text{A}$ to 10 mA
- Wide supply range for temperature stabilizer
- Guaranteed low noise
- Low power for stabilization — 300 mW at 25°C
- Long term stability — 20 ppm

Schematic Diagrams



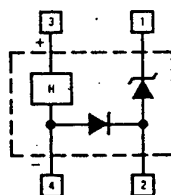
Connection Diagram

Metal Can Package



Order Number LM199H, LM299H
or LM399H
See NS Package H04D

Functional Block Diagram



Absolute Maximum Ratings

Temperature Stabilizer Voltage	40V
Reverse Breakdown Current	20 mA
Forward Current	1 mA
Reference to Substrate Voltage $V_{(RS)}$ (Note 1)	40V -0.1V
Operating Temperature Range	
LM199	-55°C to +125°C
LM299	-25°C to +85°C
LM399	0°C to +70°C
Storage Temperature Range	-55°C to +150°C
Lead Temperature (Soldering, 10 seconds)	300°C

Electrical Characteristics (Note 2)

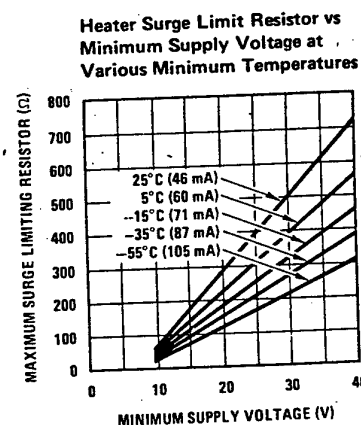
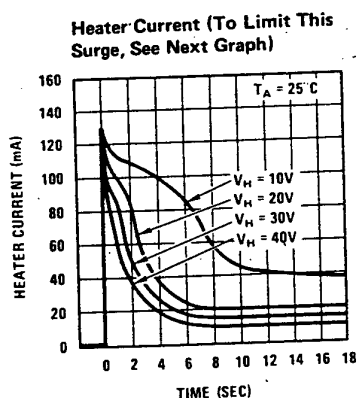
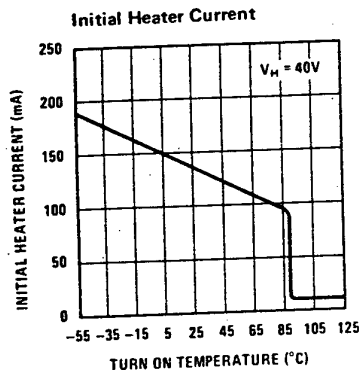
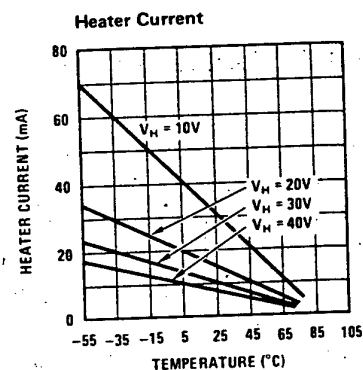
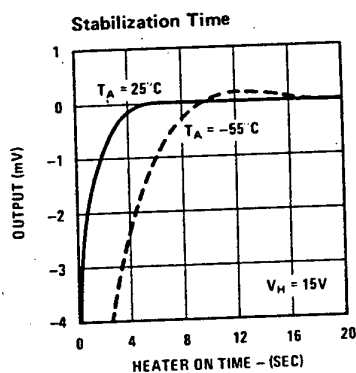
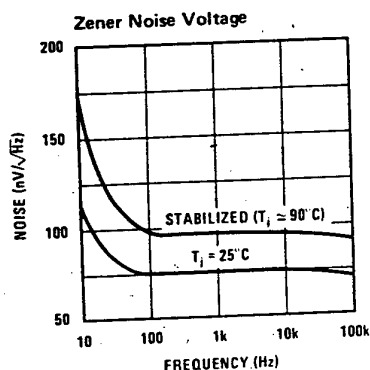
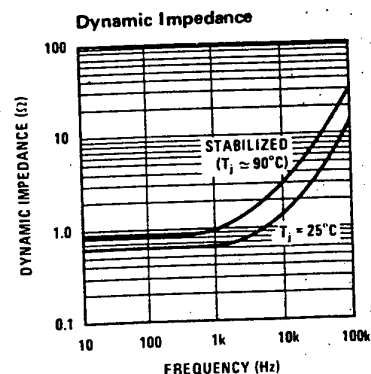
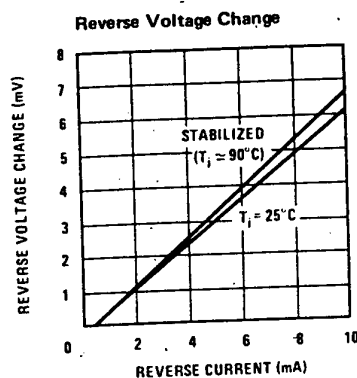
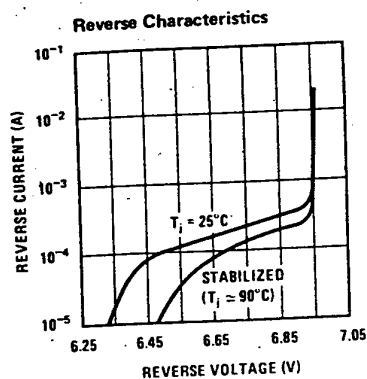
PARAMETER	CONDITIONS	LM199/LM299			LM399			UNITS
		MIN	TYP	MAX	MIN	TYP	MAX	
Reverse Breakdown Voltage	$0.5 \text{ mA} \leq I_R \leq 10 \text{ mA}$	6.8	6.95	7.1	6.6	6.95	7.3	V
Reverse Breakdown Voltage Change With Current	$0.5 \text{ mA} \leq I \leq 10 \text{ mA}$		6	9		6	12	mV
Reverse Dynamic Impedance	$I_R = 1 \text{ mA}$		0.5	1		0.5	1.5	Ω
Reverse Breakdown Temperature Coefficient	$-55^\circ\text{C} \leq T_A \leq 85^\circ\text{C}$ $85^\circ\text{C} \leq T_A \leq 125^\circ\text{C}$		0.00003	0.0001				%/°C
	LM199		0.0005	0.0015				%/°C
	$-25^\circ\text{C} \leq T_A \leq 85^\circ\text{C}$		0.00003	0.0001				%/°C
	LM299					0.00003	0.0002	%/°C
	$0^\circ\text{C} \leq T_A \leq 70^\circ\text{C}$							%/°C
	LM399							
RMS Noise	$10 \text{ Hz} \leq f \leq 10 \text{ kHz}$		7	20		7	50	μV
Long Term Stability	Stabilized, $22^\circ\text{C} \leq T_A \leq 28^\circ\text{C}$, 1000 Hours, $I_R = 1 \text{ mA} \pm 0.1\%$		20			20		ppm
Temperature Stabilizer Supply Current	$T_A = 25^\circ\text{C}$, Still Air, $V_S = 30\text{V}$ $T_A = -55^\circ\text{C}$		8.5	14		8.5	15	mA
Temperature Stabilizer Supply Voltage	(Note 3)	9		40	9		40	V
Warm-Up Time to 0.05%	$V_S = 30\text{V}$, $T_A = 25^\circ\text{C}$		3			3		Seconds
Initial Turn-on Current	$9 \leq V_S \leq 40$, $T_A = 25^\circ\text{C}$, (Note 3)		140	200		140	200	mA

Note 1: The substrate is electrically connected to the negative terminal of the temperature stabilizer. The voltage that can be applied to either terminal of the reference is 40V more positive or 0.1V more negative than the substrate.

Note 2: These specifications apply for 30V applied to the temperature stabilizer and $-55^\circ\text{C} \leq T_A \leq +125^\circ\text{C}$ for the LM199; $-25^\circ\text{C} \leq T_A \leq +85^\circ\text{C}$ for the LM299 and $0^\circ\text{C} \leq T_A \leq +70^\circ\text{C}$ for the LM399.

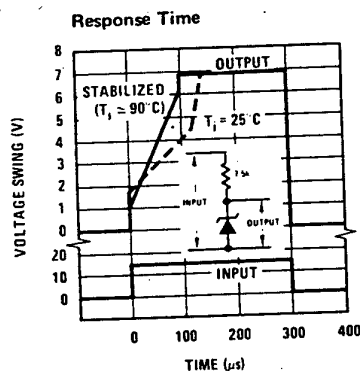
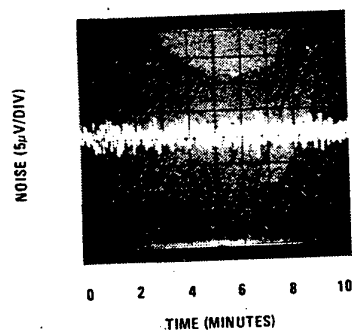
Note 3: This initial current can be reduced by adding an appropriate resistor and capacitor to the heater circuit. See the performance characteristic graphs to determine values.

Typical Performance Characteristics



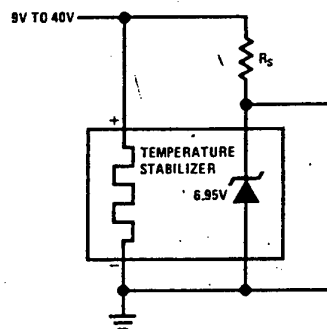
* Heater must be bypassed with a $2 \mu\text{F}$ or larger tantalum capacitor if maximum value resistors are used. Otherwise, 30% to 50% smaller values must be used. If heater oscillates, resistor value may be too small.

Low Frequency Noise Voltage

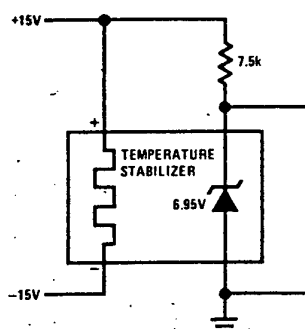


Typical Applications

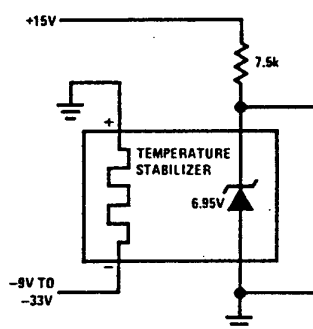
Single Supply Operation



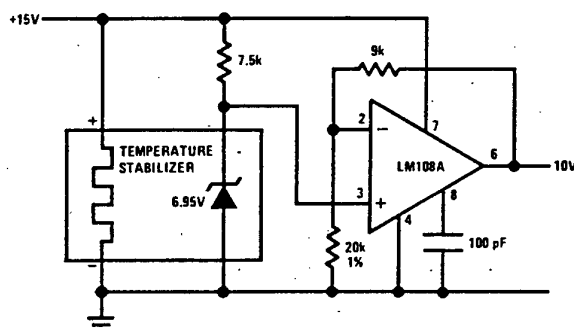
Split Supply Operation



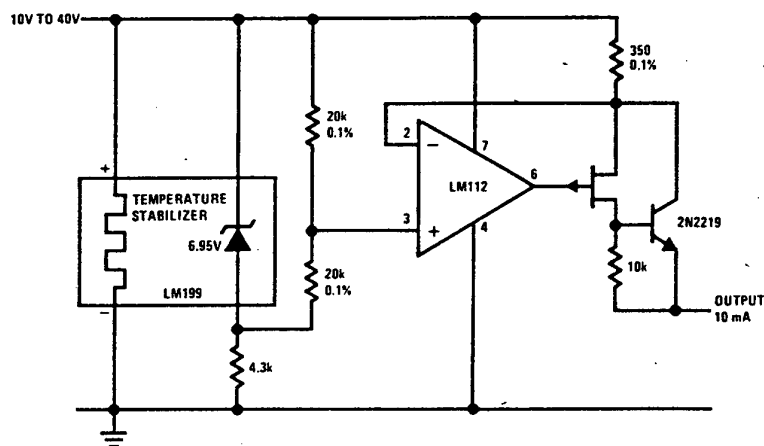
Negative Heater Supply with Positive Reference



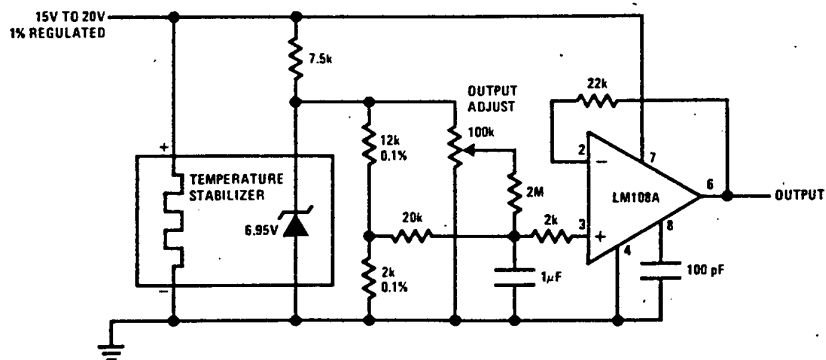
Buffered Reference With Single Supply



Positive Current Source



Standard Cell Replacement

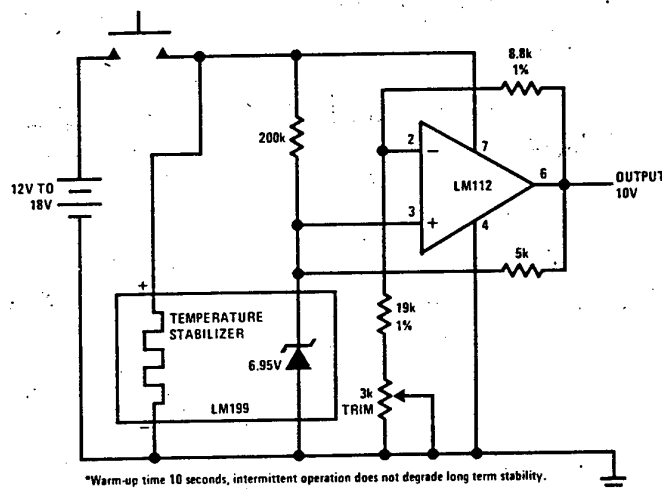
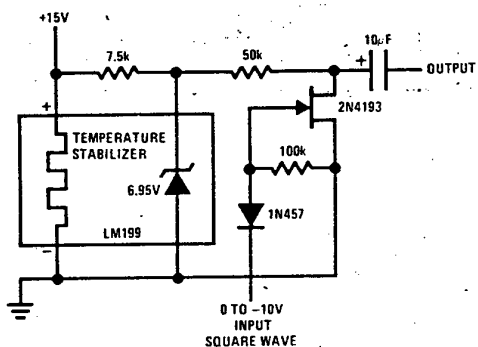


LM199/LM299/LM399

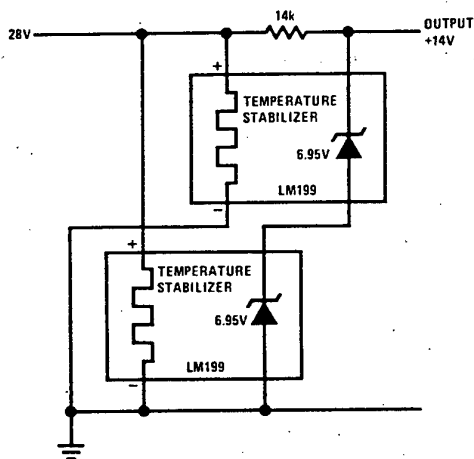
2

The circuit diagram shows a precision current source. A $-25V$ supply is connected to the non-inverting input (pin 3) of the LM112 op-amp through a $7.5k\Omega$ resistor. The inverting input (pin 2) is connected to the output (pin 6) of the LM112. The output (pin 6) is also connected to the base of a 2N2905 PNP transistor. The emitter of the 2N2905 is connected to the $-25V$ supply. The collector of the 2N2905 is connected to the output terminal, which is labeled "OUTPUT 10-20 mA". A 300Ω resistor is connected between the output terminal and the $-25V$ supply. The LM112 op-amp is configured with pin 1 to ground, pin 4 to $-25V$, and pin 7 to ground. A $6k\Omega$ resistor is connected between the output (pin 6) and the base of the 2N2905. The LM199 temperature stabilizer is connected with its non-inverting input (pin 3) to the $-25V$ supply through a $20k\Omega$ resistor. Its inverting input (pin 2) is connected to the output (pin 6) of the LM112. The output (pin 6) of the LM112 is also connected to the base of the 2N2905. The LM199 is powered by a $6.95V$ source (indicated by a triangle symbol) and has its other pins connected to ground or $-25V$ as shown.

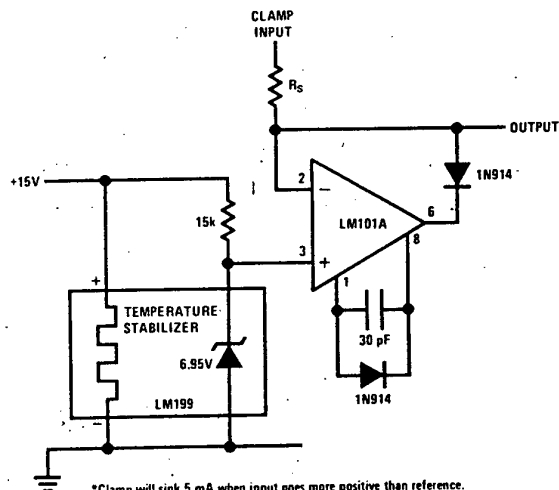
Square Wave Voltage Reference



14V Reference

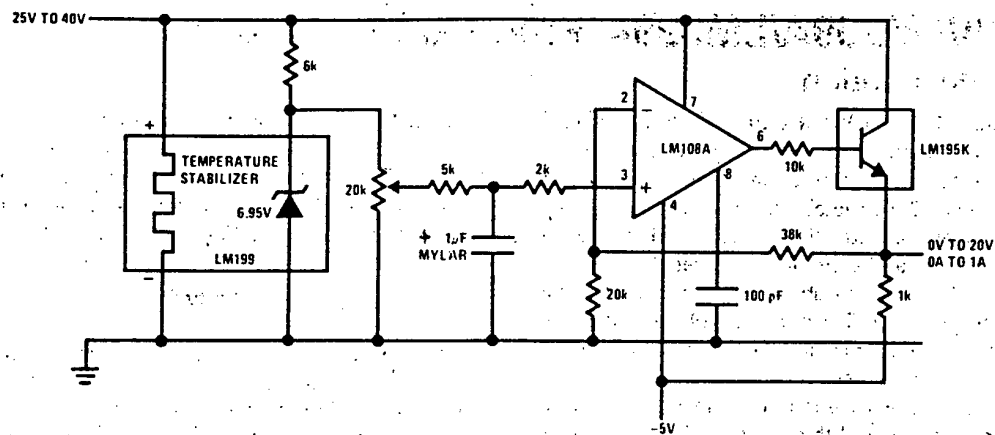


Precision Clamp*

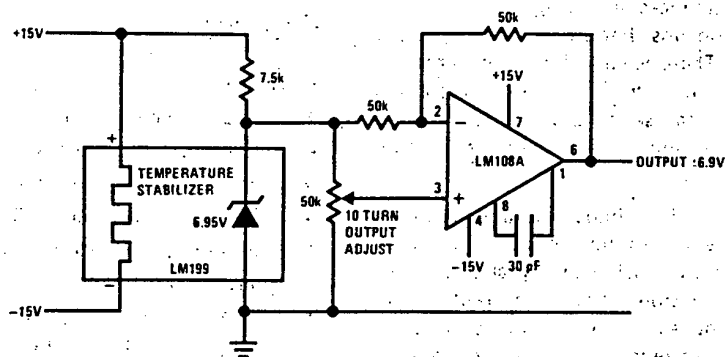


Typical Applications (Continued)

0V to 20V Power Reference



Bipolar Output Reference



LM199/LM299/LM399

2

TTL
MSI

TYPES SN5490A, SN5492A, SN5493A, SN54LS90, SN54LS92, SN54LS93, SN7490A, SN7492A, SN7493A, SN74LS90, SN74LS92, SN74LS93 DECADE, DIVIDE-BY-TWELVE, AND BINARY COUNTERS

BULLETIN NO. DL-S 7611807, MARCH 1974—REVISED OCTOBER 1976

'90A, 'LS90...DECADE COUNTERS

'92A, 'LS92...DIVIDE-BY-TWELVE
COUNTERS

'93A, 'LS93...4-BIT BINARY
COUNTERS

TYPES	TYPICAL POWER DISSIPATION
'90A	145 mW
'LS90	45 mW
'92A, '93A	130 mW
'LS92, 'LS93	45 mW

description

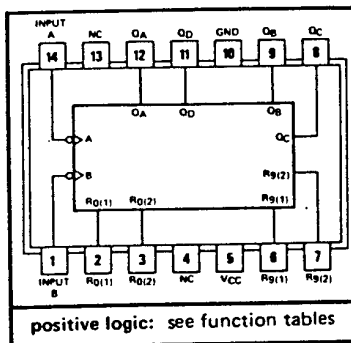
Each of these monolithic counters contains four master-slave flip-flops and additional gating to provide a divide-by-two counter and a three-stage binary counter for which the count cycle length is divide-by-five for the '90A and 'LS90, divide-by-six for the '92A and 'LS92, and divide-by-eight for the '93A and 'LS93.

All of these counters have a gated zero reset and the '90A and 'LS90 also have gated set-to-nine inputs for use in BCD nine's complement applications.

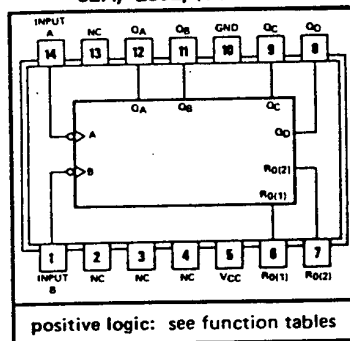
To use their maximum count length (decade, divide-by-twelve, or four-bit binary) of these counters, the B input is connected to the Q_A output. The input count pulses are applied to input A and the outputs are as described in the appropriate function table. A symmetrical divide-by-ten count can be obtained from the '90A or 'LS90 counters by connecting the Q_D output to the A input and applying the input count to the B input which gives a divide-by-ten square wave at output Q_A .

SN54', SN54LS' ... J OR W PACKAGE
SN54', SN74LS' ... J OR N PACKAGE

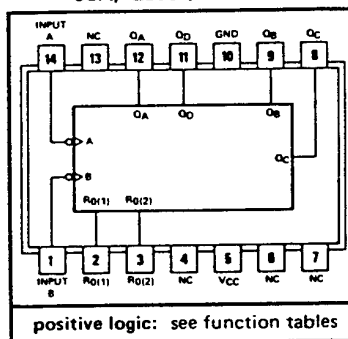
'90A, 'LS90 (TOP VIEW)



'92A, 'LS92, (TOP VIEW)



'93A, 'LS93 (TOP VIEW)



TYPES SN5490A, '92A, '93A, SN54LS90, 'LS92, 'LS93, SN7490A, '92A, '93A, SN74LS90, 'LS92, 'LS93 DECADE, DIVIDE-BY-TWELVE, AND BINARY COUNTERS

'90A, 'LS90
BCD COUNT SEQUENCE
(See Note A)

COUNT	OUTPUT			
	Q _D	Q _C	Q _B	Q _A
0	L	L	L	L
1	L	L	L	H
2	L	L	H	L
3	L	L	H	H
4	L	H	L	L
5	L	H	L	H
6	L	H	H	L
7	L	H	H	H
8	H	L	L	L
9	H	L	L	H

'90A, 'LS90
BI-QUINARY (5-2)
(See Note B)

COUNT	OUTPUT			
	Q _A	Q _D	Q _C	Q _B
0	L	L	L	L
1	L	L	L	H
2	L	L	H	L
3	L	L	H	H
4	L	H	L	L
5	H	L	L	L
6	H	L	L	H
7	H	L	H	L
8	H	L	H	H
9	H	H	L	L

'92A, 'LS92
COUNT SEQUENCE
(See Note C)

COUNT	OUTPUT			
	Q _D	Q _C	Q _B	Q _A
0	L	L	L	L
1	L	L	L	H
2	L	L	H	L
3	L	L	H	H
4	L	H	L	L
5	L	H	L	H
6	H	L	L	L
7	H	L	L	H
8	H	L	H	L
9	H	L	H	H
10	H	H	L	L
11	H	H	L	H

'93A, 'LS93
COUNT SEQUENCE
(See Note C)

COUNT	OUTPUT			
	Q _D	Q _C	Q _B	Q _A
0	L	L	L	L
1	L	L	L	H
2	L	L	H	L
3	L	L	H	H
4	L	H	L	L
5	L	H	L	H
6	L	H	H	L
7	L	H	H	H
8	H	L	L	L
9	H	L	L	H
10	H	L	H	L
11	H	L	H	H
12	H	H	L	L
13	H	H	L	H
14	H	H	H	L
15	H	H	H	H

'90A, 'LS90
RESET/COUNT FUNCTION TABLE

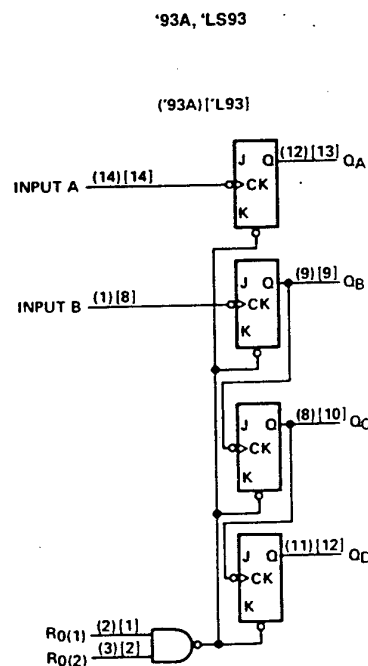
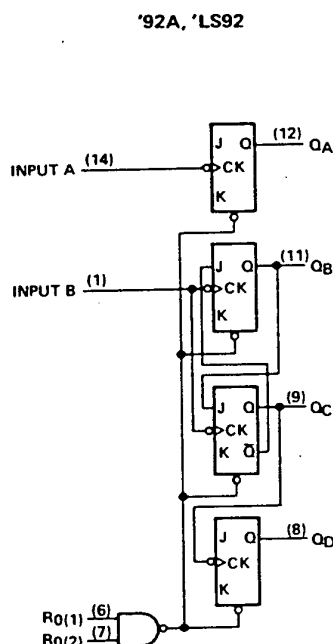
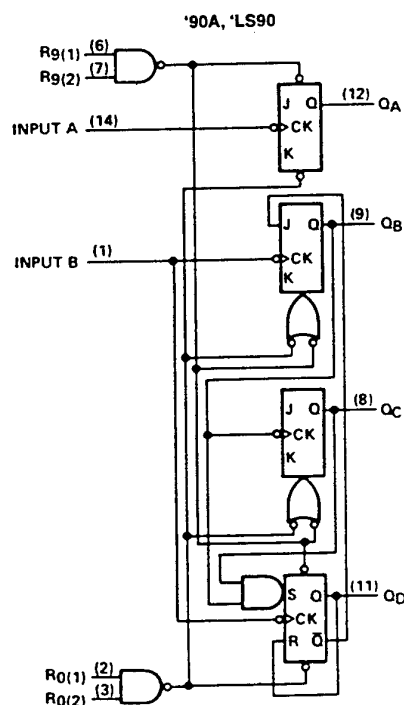
RESET INPUTS				OUTPUT			
R ₀ (1)	R ₀ (2)	R ₉ (1)	R ₉ (2)	Q _D	Q _C	Q _B	Q _A
H	H	L	X	L	L	L	L
H	H	X	L	L	L	L	L
X	X	H	H	H	L	L	H
X	L	X	L	COUNT			
L	X	L	X	COUNT			
L	X	X	L	COUNT			
X	L	L	X	COUNT			

'92A, 'LS92, '93A, 'LS93
RESET/COUNT FUNCTION TABLE

RESET INPUTS		OUTPUT			
R ₀ (1)	R ₀ (2)	Q _D	Q _C	Q _B	Q _A
H	H	L	L	L	L
L	X	COUNT			
X	L	COUNT			

- NOTES: A. Output Q_A is connected to input B for BCD count.
B. Output Q_D is connected to input A for bi-quinary count.
C. Output Q_A is connected to input B.
D. H = high level, L = low level, X = irrelevant

functional block diagrams



The J and K inputs shown without connection are for reference only and are functionally at a high level.

CCD442A

2048 x 2048 Element

Full Frame Image Sensor

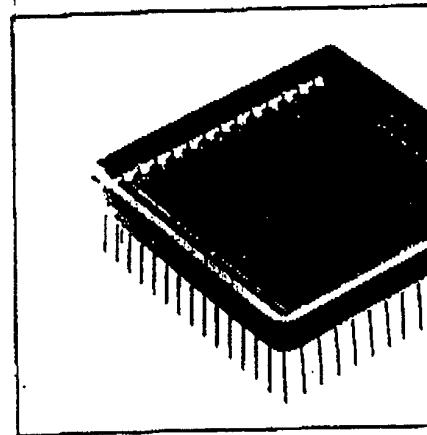
FEATURES

- 2048 x 2048 Photosite Array
- 15µm x 15µm Pixel
- 30.72mm x 30.72mm Image Area
- Near 100% Fill Factor
- Multi-Pinned Phase (MPP) Option
- Readout Noise Less Than 7 Electrons at 250k pixels/sec
- Dynamic Range 10000:1
- Three Phase Buried Channel NMOS

GENERAL DESCRIPTION

The CCD442A is a 2048 x 2048 element solid state Charge Coupled Device (CCD) Full Frame area image sensor which is intended for use in high resolution scientific, industrial, and commercial electro-optical systems. The CCD442A is organized as a matrix array of 2048 horizontal by 2048 vertical CCD photosites. The pixel pitch and spacing is 15µm. For dark reference the top and bottom eight rows and the left and right eight columns are covered by a light shield. The available imaging area is thus 2032 rows by 2032 columns.

The imaging array may be operated in one of three modes, Buried Channel or Multi-Pinned Phase (MPP). The Buried Channel operation offers low noise performance and excellent charge transfer efficiencies. An additional implant under one vertical phase creates a virtual well which collects the photo-electrons with all Vertical clocks low during integration. This MPP mode decreases dark current down to 25 pA/cm² @ 25°C. Excellent low noise performance is achieved by use of the buried channel CCD structure and a dual stage low noise output amplifier with an output conversion of 3µV/e.



Device processing is done using 2.5 micron design rules. The single metal, triple-poly process allows a photosite layout with smaller pixel geometries and fewer array blemishes.

FUNCTIONAL DESCRIPTION

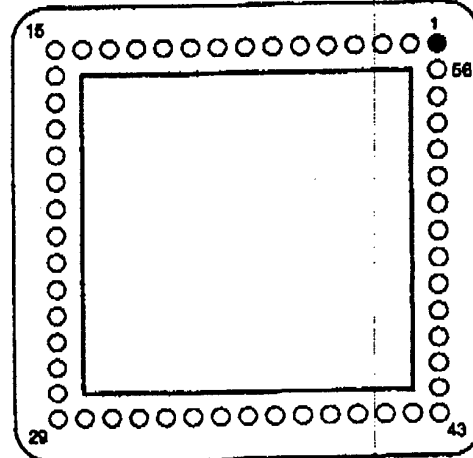
The CCD442A consists of the following functional elements illustrated in the block diagram.

Image Sensing Elements: Incident photons pass through a transparent polycrystalline silicon gate structure creating electron hole pairs. The resulting photo-electrons are collected in the photosites during the integration period. The amount of charge accumulated in each photosite is a linear function of the localized incident illumination intensity and integration period.

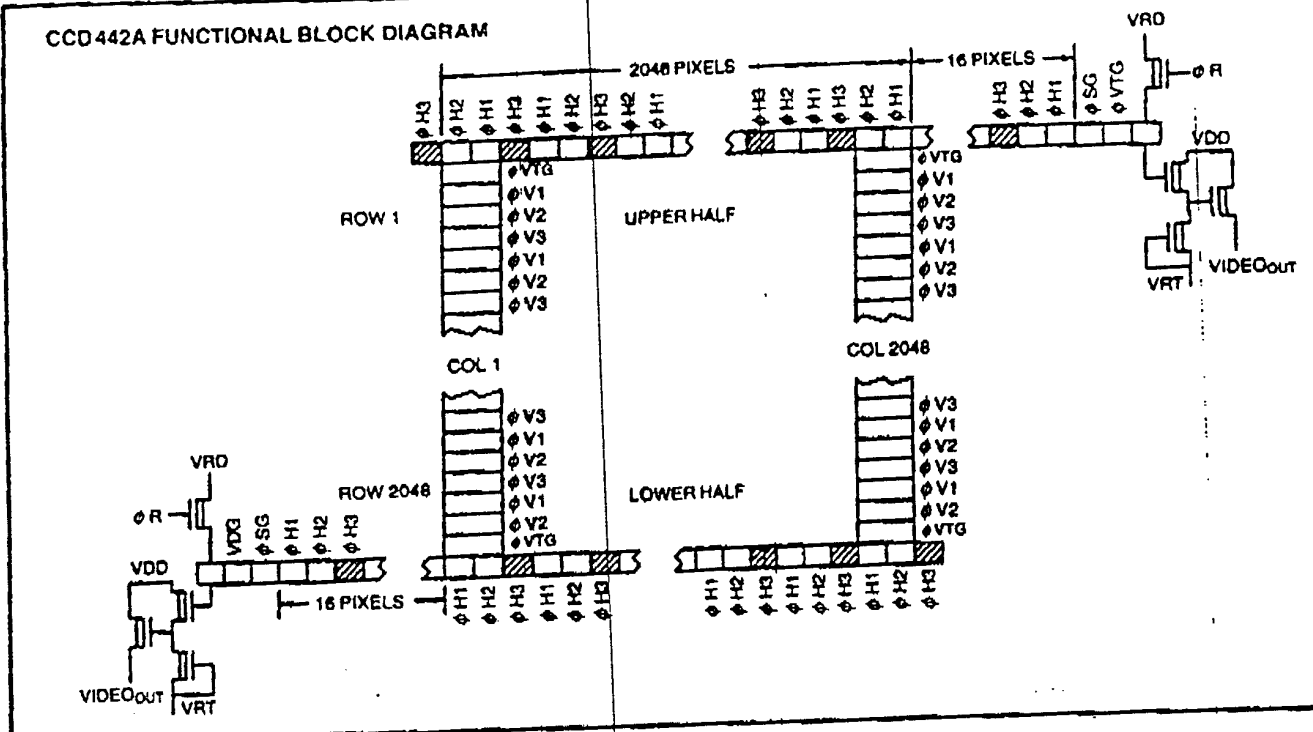
PIN NUMBER/NAME

1 NC	20 φV2 _U	39 φH3 _L
2 NC	21 φV1 _U	40 φVTG _L
3 V _{SS}	22 V _{SS}	41 V _{SS}
4 φR _U	23 φVTG _L	42 NC
5 VRD _U	24 φSG _L	43 NC
6 VRD _U	25 VOG _L	44 NC
7 VIDEO _{OUT}	26 V _{SS}	45 V _{SS}
8 VDD _U	27 V _{SS}	46 V _{SS}
9 φH1 _U	28 V _{SS}	47 φV3 _L
10 φH2 _U	29 NC	48 φV1 _L
11 φH3 _U	30 NC	49 φV2 _L
12 φVTG _U	31 V _{SS}	50 V _{SS}
13 V _{SS}	32 φR _L	51 φVTG _U
14 NC	33 VRD _L	52 φSG _U
15 NC	34 VRT _L	53 VOG _U
16 NC	35 VIDEO _{OUT}	54 V _{SS}
17 V _{SS}	36 VDD _L	55 V _{SS}
18 V _{SS}	37 φH1 _L	56 V _{SS}
19 φV3 _U	38 φH2 _L	

PIN CONNECTIONS



CCD442A FUNCTIONAL BLOCK DIAGRAM



The photosite structure is made up of contiguous CCD elements with no voids or inactive areas. In addition to sensing light, these elements are used to shift image data vertically. Consequently, the device needs to be shuttered during readout.

Vertical Charge Shifting: The Full Frame architecture of the CCD442A provides video information as a single sequential readout of 2048 lines containing 2048 photosite elements. At the end of an integration period the ϕV_1 , ϕV_2 , and ϕV_3 clocks, are used to transfer charge vertically through the CCD array to the horizontal readout register. Vertical columns are separated by a channel stop region to prevent charge migration.

The imaging area is divided into an Upper and Lower half. Each 1024×2048 half may be clocked independently or together. Horizontal Transport registers along the top and bottom permit simultaneous readout of both halves. The CCD442A may be clocked such that the full array is readout the Upper or Lower Transport registers. The package pinouts are arranged so that the device may be rotated 180° without timing changes.

The Vertical Transfer Gate (ϕVTG) is the final array gate before charge is transferred to the serial horizontal shift registers. For simplified operation ϕVTG may be tied to ϕV_3 .

Horizontal Charge Shifting: ϕH_1 , ϕH_2 , and ϕH_3 are polysilicon gates used to transfer charge horizontally to the output amplifier. The horizontal transport register is twice the size of the photosite to allow for vertical binning. The array can be operated normally at full resolution or some lower resolution with binning.

The transfer of charge into the horizontal register is the result of a vertical shift sequence. This register has 16 additional register cells between the first pixel of each line and the output amplifier. The output from these locations contain no signal and may be used as a dark level reference.

The last clocked gate in the Horizontal registers is twice as large as the others and can be used to horizontally bin charge.

This gate requires its own clock which should be tied to ϕH_1 for normal full resolution readout. The output video is available following the high to low transition of ϕSG .

The reset FET in the horizontal readout, clocked appropriately with ϕR , allows binning of adjacent pixels.

Output Amplifier: The CCD442A has one output amplifier at the end of the horizontal transport registers. They are dual FET floating diffusion amplifiers with a reset MOSFET tied to the input gate.

Charge packets are clocked to a precharged capacitor whose potential changes linearly in response to the number of electrons delivered. This potential is applied to the input gate of an NMOS amplifier producing a signal at the output V_{out} pin. The capacitor is reset with ϕR to a precharge level prior to the arrival of the next charge packet except when horizontally binning. It is reset by use of the reset MOSFET.

The output amplifier drain is tied to VDD. The source (Video Out) is connected to an external load resistor to ground. The source constitutes the video output from the device.

Multi-Pinned Phase: MPP is a CCD technology which significantly reduces the dark current generation rate. CCDs are endowed with this capability by the addition of an implant during the semiconductor manufacturing process.

This implant creates a virtual well in the array which allows charge integration while maintaining pixel integrity with the Vertical clocks in the low state. Leaving the Vertical clocks in the low state during the integration cycle is the method used to implement MPP mode.

A drawback to utilizing the MPP mode is reduced full well capacity. The virtual well created by the MPP implant does not hold as much charge as the normal buried channel operating mode which leaves one Vertical clock in the high state during integration. The CCD442A may be operated in the conventional buried channel mode with increase in charge capacity over the MPP mode.

DEFINITION OF TERMS

Charge-Coupled Device—A charge-coupled device is a monolithic silicon structure in which discrete packets of electron charge are transported from position to position by sequential clocking of an array of gates.

Vertical Transport Clocks ϕV_1 , ϕV_2 , ϕV_3 —The clock signals applied to the vertical transport register.

Horizontal Transport Clocks ϕH_1 , ϕH_2 , ϕH_3 —The clock signals applied to the horizontal transport registers.

Reset Clock ϕR —The clock applied to the reset switch of the output amplifier.

Dynamic Range—The ratio of saturation output voltage to RMS noise in the dark. The peak-to-peak random noise is 4–6 times the RMS noise output.

Saturation Exposure—The minimum exposure level that produces an output signal corresponding to the maximum photosite charge capacity. Exposure is equal to the product of light intensity and integration time.

Responsivity—The output signal voltage per unit of exposure.

Spectral Response Range—The spectral band over which the response per unit of radiant power is more than 10% of the peak response.

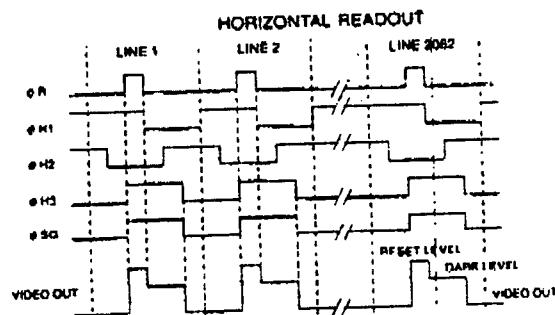
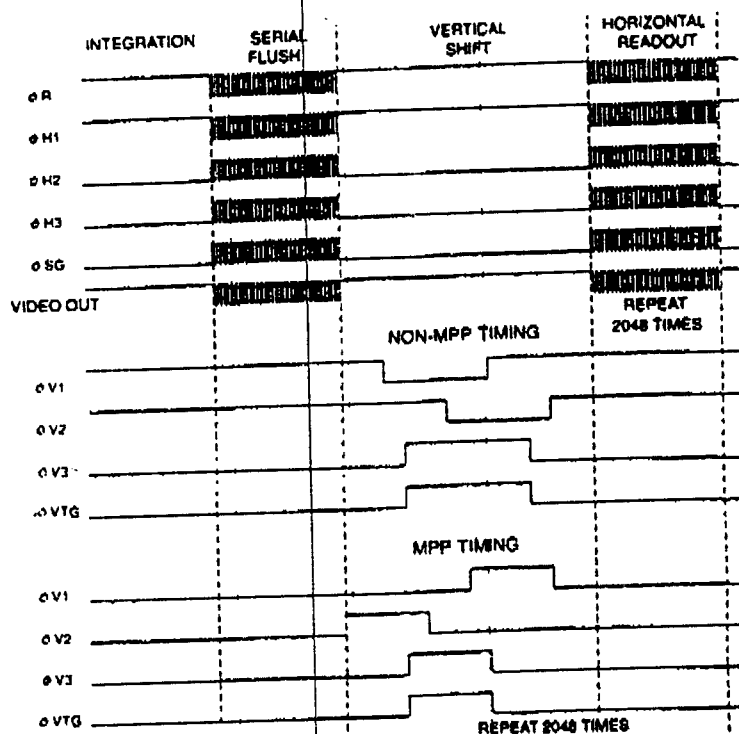
Photo-Response Non-Uniformity—The difference of the response levels between the most and the least sensitive regions under uniform illumination (excluding blemished elements) expressed as a percentage of the average response.

Dark Signal—The output signal in the dark caused by thermally generated electrons. Dark signal is a linear function of integration time and an exponential function of chip temperature.

Vertical Transfer Gate ϕVTG —Gate structures adjacent to the end row of photosites and the horizontal transport registers. The charge packets accumulated in the photosites are shifted vertically through the array. Upon reaching the end row of photosites, the charge is transferred in parallel via the transfer gates to the horizontal transport shift registers whenever the transfer gate voltage goes high.

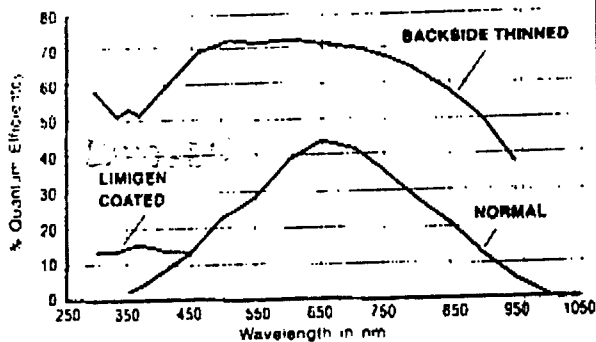
Pixel—Picture element or sensor element also called photoelement or photosite.

TIMING DIAGRAM



CCD442A

TYPICAL CCD QUANTUM EFFICIENCY



QUANTUM EFFICIENCY ENHANCEMENTS

On a custom basis, our large area CCDs can be backside thinned for increased QE. The CCD is bump mated to a fanout and thinned to approximately 15 microns. The incident illumination enters through the backside of the array. Since no photons are absorbed in the polysilicon gate structures, the QE increases. We can also coat frontside illuminated devices with a fluorescent dye that absorbs UV light and fluoresces in the visible range. This provides CCD response at wavelengths less than 400nm.

TYPICAL DC VOLTAGES

SYMBOL	PARAMETER	RANGE			UNIT	REMARKS
		MIN	NOM	MAX		
VDD	DC Supply Voltage		20.0	25	V	
V _{RD}	Reset Drain Voltage	12	13.0	16	V	
V _{OG}	Output Gate Voltage		1.0		V	
V _{SS}	Substrate Ground		0.0		V	

TYPICAL CLOCK VOLTAGES

SYMBOL	PARAMETER	HIGH	LOW	UNIT	REMARKS
V ϕ H _(1,2,3)	Horizontal Multiplexer Clock	+5.0	-5.0	V	
V ϕ V _(1,2,3)	Vertical Array Clocks	+3.0	-8.0	V	
V ϕ R	Reset Gate Clock	+8.0	0.0	V	
V ϕ VTG	Array Transfer Gate Clock	+3.0	-8.0	V	

Note ϕ H=400pF, ϕ V=60,000pF

PERFORMANCE SPECIFICATIONS

SYMBOL	PARAMETER	RANGE			UNIT	REMARKS
		MIN	NOM	MAX		
V _{SAT}	Saturation Output Voltage	300		1200	mV	Note 1
	Full Well Capacity	100,000		400,000	e-	
	Output Amp Sensitivity		3.0		μ V/e-	
PRNU	Photo-Response Non-Uniformity Peak-to-Peak			10	%V _{SAT}	
DSNU	Dark Signal Non-Uniformity Peak-to-Peak			1.0	mV	
DC	Dark Current	0.025		2.0	nA/cm ²	Note 2
R	Responsivity		1.0		V/ μ J/cm ²	
V _{ODC}	Output DC Level		14.0		V	Note 3
Z	Suggested Load Register	1.0	5.0	20	k Ω	Note 3

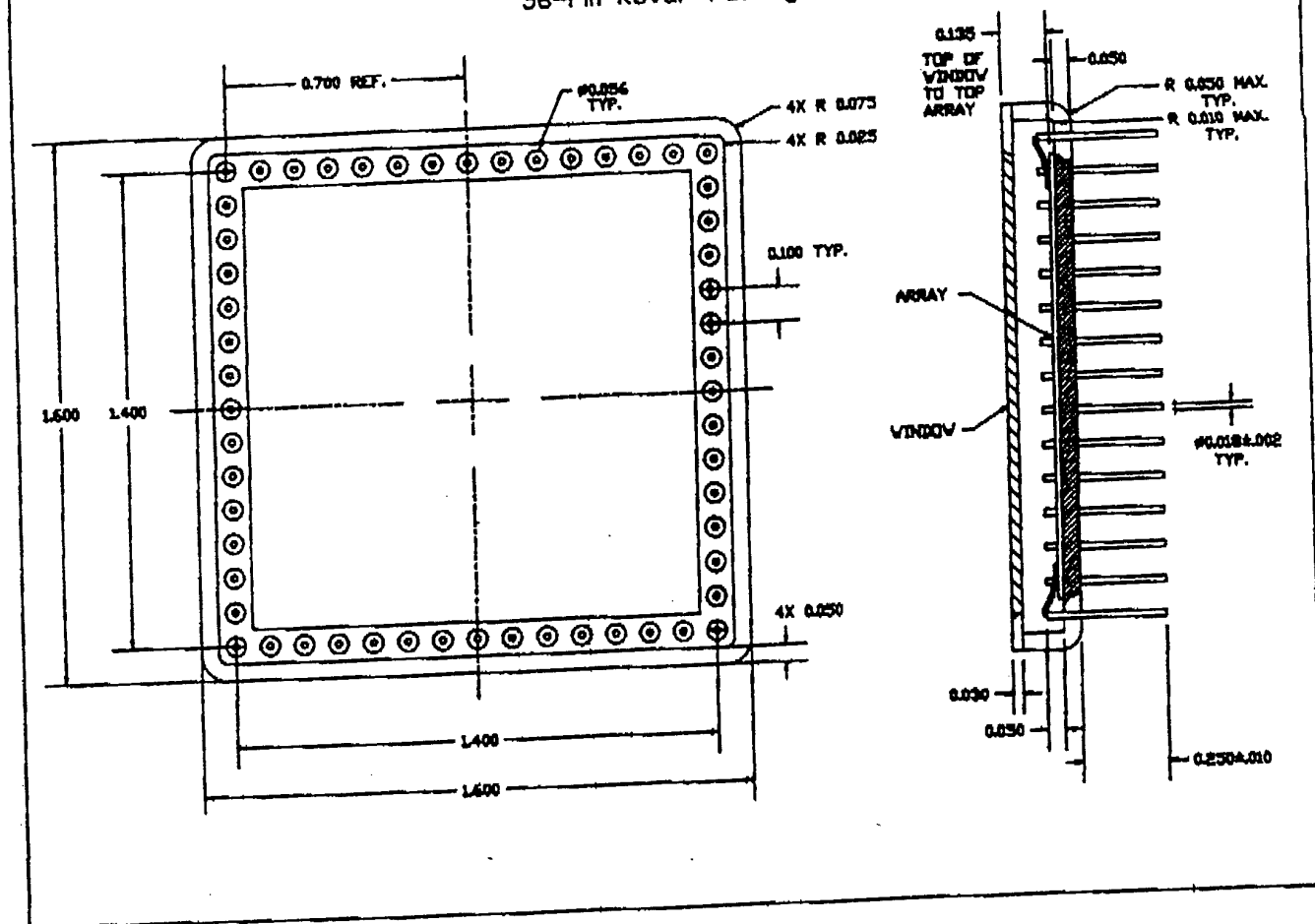
Note 1 Maximum well capacity is achieved operating in Buried Channel Mode; minimum capacity is in MPP mode.

Note 2 Values shown are for 25 C. Dark current doubles for every 4 - 6 C.

Note 3 Standard test conditions are nominal MPP clocks and DC operating voltages. 1MHz Horizontal Data Rate, 6 μ Sec Vertical Shift Cycle.

CCD442A

CCD442 PACKAGE OUTLINE
56-Pin Kovar Package



COSMETIC GRADING

Device grading helps to establish a ranking for the image quality that a CCD will provide. Blemishes are characterized as spurious pixels exceeding 10% of V_{sat} with respect to neighboring elements. Blemish content is determined in the dark, at various illumination levels and for different device temperatures.

The CCD442A is available in various standard grades, as well as custom selected grades. Consult the factory for available grading information and custom selections.

WARRANTY

WARRANTY
Within twelve months of delivery to the end customer, Loral Fairchild will repair or replace, at our option, any Loral Fairchild camera product if any part is found to be defective in materials or workmanship. Contact factory for assignment of warranty return number and shipping instructions to ensure prompt repair or replacement.

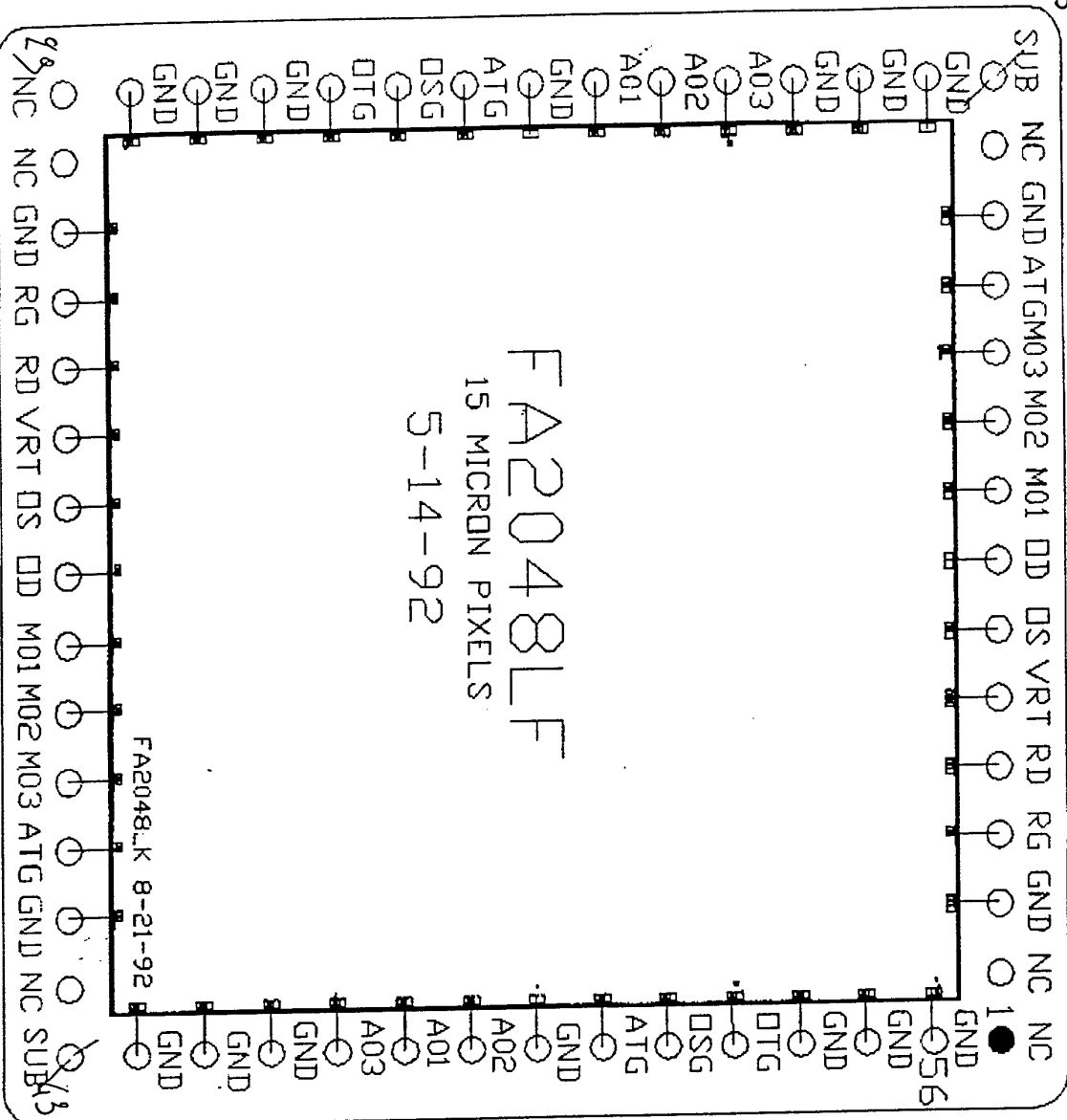
CERTIFICATION

CERTIFICATION
Loral Fairchild Division certifies that all products are carefully inspected and tested at the factory prior to shipment and will meet all requirements of the specification under which it is furnished.

LORAL
Fairchild Imaging Sensors

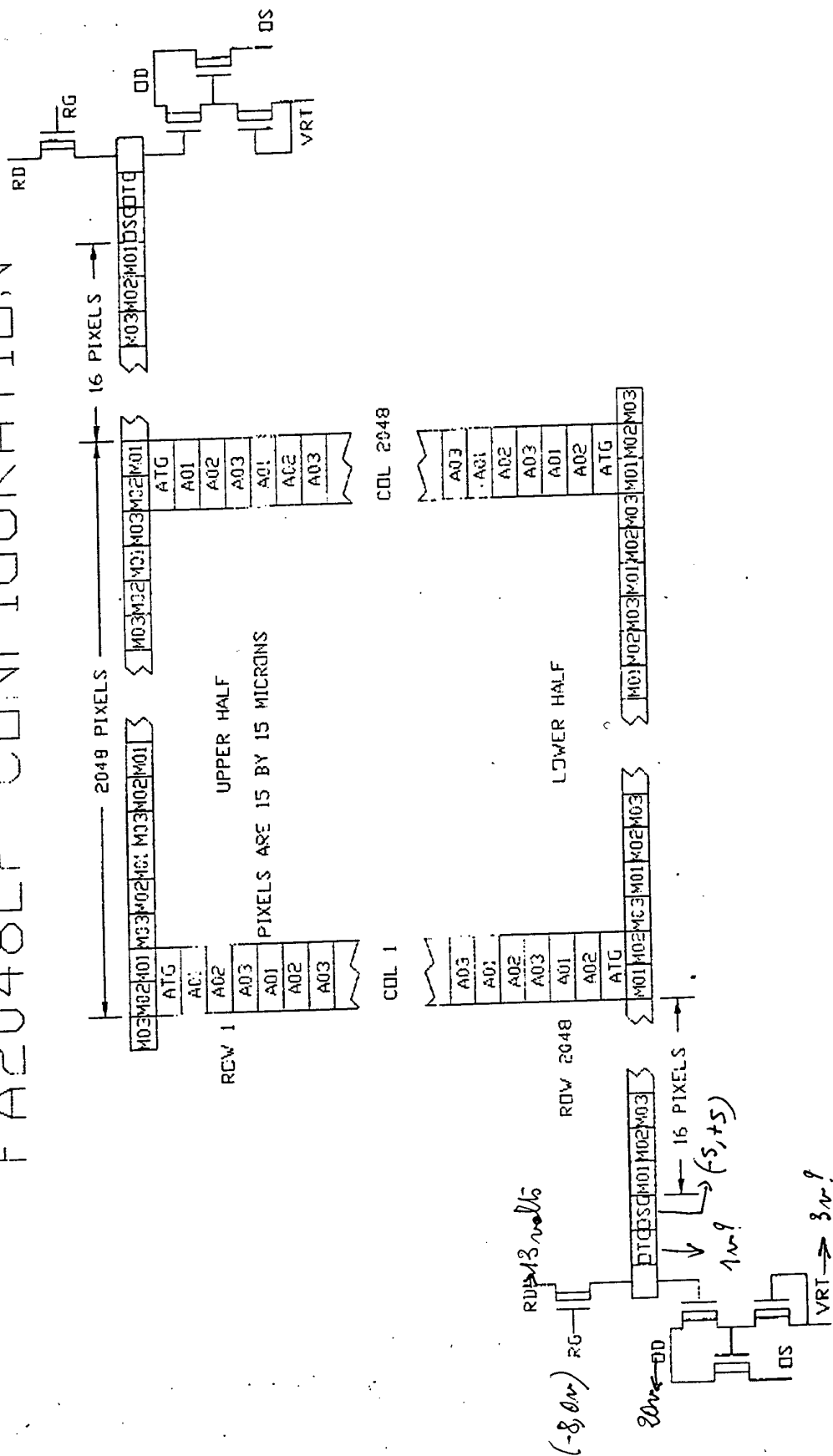
Loral Fairchild cannot assume responsibility for use of any circuitry described other than circuitry embodied in a Loral Fairchild product. No other circuit patent licenses are implied.

45



1	NC	29	NC
2	NC	30	NC
3	GND	31	GND
4	RG	32	RG
5	RD	33	RD
6	VRT	34	VRT
7	DS	35	DS
8	DD	36	DD
9	M01	37	M01
10	M02	38	M02
11	M03	39	M03
12	ATG	40	ATG
13	GND	41	GND
14	NC	42	NC
15	SUB	43	SUB
16	GND	44	GND
17	GND	45	GND
18	GND	46	GND
19	A03	47	A03
20	A02	48	A01
21	A01	49	A02
22	GND	50	GND
23	ATG	51	ATG
24	DSG	52	DSG
25	DTG	53	DTG
26	GND	54	GND
27	GND	55	GND
28	GND	56	GND

FA2048LF CONFIGURATION



93-4007

**Appendix 3 : 87C750 based
sequencer software**

In order to write this software, I used an 8051 public domain assembler which was found on the Internet. It can be found at several nodes, including for example at [csd4.csd.uwm.edu](http://csd4.csd.uwm.edu/pub/8051) in the directory /pub/8051. Its author can be reached at markh@csd4.csd.uwm.edu. It seemed to me one of the best packages in its category. We also purchased a Ceibo DS750 development system in order to program, simulate and emulate the 87C750.

In order for the reader to see its main performances, as well as for reference purpose, I included the documentation obtained with this package for reference.

CAS: The Free Full-Featured 8051 Assembler

Located at the FTP site `csd4.csd.uwm.edu` in `/pub/8051/assem`.

This is a 1-pass assembler for the MCS-51 family of microprocessors with the following features:

- (1) Separately assembleable files.
There are two stages of assembly:
 - (a) Pass 1: Creation of object files
 - (b) Pass 1 1/3: Linking of object files
- (2) Relative addressing
- (3) Unnamed segments
- (4) Conditional assembly
With a C-like syntax, e.g.

```
if (Condition) {  
    Assembly instructions...  
} else {  
    Assembly instructions...  
}
```
- (5) Multiple statements per line with C-like syntax.
- (6) C-like expression syntax.
- (7) Command-line options similar to those of the UNIX C compiler.
Compatible with UNIX's make.
- (8) An extensive archive of real-life assembly language programs, including a multi-tasking library.

Plus, if you don't want to learn all the elaborate ins and outs of this tool right away, it is just as easy to use the first time out as any minimal assembler. Read the start-up notes for details. It has been tested and verified against all the software contained in the rest of the 8051 archive. With enough demand and support, I may add in the following features in future versions of this assembler:

- (9) More compatibility between the 8051 disassembler and assembler.
- (10) A powerful macro facility, to allow significant customizations on the 8051 language.
- (11) The ability to include "Pre-assembled" macros.

This is probably the most elaborate 8051 assembler you will find anywhere in the public domain. It's yours with the source for free. Be on the lookout for future software developments...

START-UP NOTES

- (1) If you use semi-colons for comments, make sure you precede the comments by at least 2 consecutive semi-colons. Usually, this should amount to nothing more than performing a substitution ; -> ;; with your

editor. A short program (semi.c) has been provided that will do the same.

(2) If you are using an 8051 extension (like the 8052 or 8051fa), you will have to make an include file to define the special function registers and bits and values for that processor. Example include files have been provided: 8052.h and 8051fa.h.

(3) The software in the data directory has been provided as an illustration of modular programming using this assembler.

(4) About the 8051 source in the rest of the archive:

Most of the 8051 assembly software and documentation preceded the assembler provided with this package, therefore the files are not completely modular. The one exception is the source contained in `assem/data`, which is a modularized version of the source in the data directory. There is a corresponding makefile for `assem/data` in that directory.

To assemble all the other sources:

<code>crc.asm</code>	in the <code>crc</code> directory
<code>data.asm</code>	in the <code>data</code> directory
<code>debug.asm</code>	in the <code>debug</code> directory
<code>drive.asm</code>	in the <code>drive</code> directory
<code>lcd.asm</code>	in the <code>kernel</code> directory
<code>int.asm</code>	in the <code>kernel</code> directory

just run the assembler on the file indicated, for instance:

`cas crc.asm`

to assemble `crc.asm`. All the other assembly language modules have names ending in `.lib`, and are combined with the source module (`crc.asm` in the example above) using the assembler's file inclusion feature.

(5) If your source file was written for another non-minimal assembler, make sure your directives are converted to conform to this assembler. For reference, the directives accepted by this assembler are:

<code>include "FILE"</code>	--- File inclusion
<code>seg Type</code>	--- Setting current segment and/or location

seg Type at Loc
seg Type org Loc
at Loc
org Loc

Type = code, xdata, or data.

LABEL equ Val
LABEL Type Val
or
LABEL:
LABEL set Val
LABEL = Val

--- Defining new labels
Type = code, xdata, data, sfr,
bit.

<digit>:

--- Defining and using numeric
labels

<digit>f
<digit>b

global LABEL equ Val
global LABEL Type Val
global LABEL
public LABEL equ Val
public LABEL Type Val
public LABEL

--- Declaring global labels.

extern equ LABEL, ..., LABEL
extern Type LABEL, ..., LABEL

--- Declaring external labels
Type = code, xdata, data, sfr, or
bit

ds Val
rb Val
rw Val

--- Memory allocation

db Val, ..., Val

--- Memory formatting. db/byte
can accept strings byte

Val, ..., Val
dw Val, ..., Val
word Val, ..., Val

if (Val) <Statement>
if (Val) <Statement> else <Statement>

--- Conditional assembly

{ <Statement> ... <Statement> }

--- Statement grouping

<Statement>; <Statement> --- Multiple statements on a line.

Documentation of the assembler

CAS -- The 8051 C-Assembler

(0) Introduction

(a) Features

This is a free full-featured one-pass 8051 assembler, it could very well be the first one-pass assembler for the popular MCS-51 family of microprocessors.

What you get are the following features:

- * Separately assembleable files. There are two stages of assembly:
 - Pass 1: Creation of object files
 - Pass 1 1/3: Linking of object files
- * Segmentation
 - RELATIVE ADDRESSING supported for all segment types
- * Conditional assembly, with a C-like syntax. Example:

```
if (Condition) {
    Assembly instructions...
} else {
    Assembly instructions...
}
```
- * Multiple statements per line with C-like syntax.
- * C-like expression syntax.
- * Command-line options similar to those of *NIX C compilers.
- * An extensive archive of real-life assembly language programs, including a multi-tasking library and an 8051 disassembler.

Plus, if you don't want to learn all the elaborate ins and outs of this tool right away, it is just as easy to use the first time out as any minimal assembler.

You simply will not find anything this extensive anywhere in the public domain. But it's yours, here, for free.

Also under works: a compatible 8051 simulator kit for software developers.

What makes this kit unique is that you can (and usually must) link in your own C code to define any arbitrary 8051 environment at all. This gives you the flexibility to simulate the 8051 in your favorite embedded application and to even simulate the I/O on a desktop. A Standard Environment file is included with the package.

(b) Design Philosophy ... everything is done in one pass.

A clean distinction is made between the two phases of assembly: (a) creating segments and formatting image files, (b) mapping segments and resolving references to variable addresses.

An assembly language program will normally consist of a set of assembly language modules (or source files). Each will typically be named with the suffix ".s". In addition, there will also be a set of files, with names ending in ".h" whose purpose is to provide common points of reference for declarations of objects in or related to modules. They are incorporated in *.s files using the "include" directive.

The first stage of assembly will create OBJECT files, whose names end in ".o": one for each assembly language module. For instance, a module named Kernel.s will be assembled to the object file Kernel.o.

The second stage will take all the object files that have been created and LINK them together. This process will consist mainly of completing the definitions of variables defined in one module and used in another, and in mapping the memory segments defined in each module onto a memory image.

These two stages correspond roughly to the first and second pass of a traditional two-pass assembler. But there are now two major differences:

(a) the second stage can now be deferred. It is possible to assemble object files only, and defer the linking phase. Furthermore, it is possible to use the SAME object file in more than one project.

(b) the second stage is now considerably shortened compared to the second pass of a traditional two-pass assembler because object files tend to be much smaller than source files and because the assembler no longer has to process the assembly language itself by the second stage.

(1) Command line arguments

The cas assembler's command line basically follows that of a typical C compiler. In the examples:

- (a) `cas -c kernel.s`
- (b) `cas -c math.s data.s stdio.s kernel.s`
- (c) `cas math.s data.s stdio.s kernel.s`
- (d) `cas -o data.hex math.o data.s stdio.o kernel.s`

- (a) will assemble the file kernel.s, creating kernel.o.
- (b) will assemble all the files listed, creating .o files in the process.
If a .o file is listed with the -c option, it is ignored.
- (c) will assemble all the files listed, as in (b), and then link all the corresponding .o files. The output file will take the same base name as the first file listed, and will have the suffix .hex. Therefore, the output in this example will be math.hex.
- (d) will do the same as (c), but will name the output file data.hex.
If a .o file is listed in either of these two command lines it will be

ignored during assembly, but will be used during linking.

(2) Directives

The following is a summary of the directives available in this language.

(a) FILE INCLUSION -- include "FILE"

This command will read the contents of the file named (FILE) into the current location of the current file. By convention, include files should have names ending in ".h" or ".i" and should only consist of declarations.

Include files generally serve two purposes: to provide a place to store related constant definitions and declarations, to declare the globally visible objects of an assembly language module.

(b) Setting current SEGMENT and LOCATION -- seg, at, org

At any point in scanning a *.s assembly language file, the assembler will recognize a current segment and current location. The latter can be referred to by the user as \$.

To see how these items can be set, look at the following examples:

```
seg code
seg xdata at 0x8000
seg xdata org 0x8000
org 50
at 50
```

The first example sets the current segment to the type "code". The current location is left unspecified. THIS IS HOW RELATIVE ADDRESSING IS INITIATED.

The actual address of the segment's start will not be determined when the object file is created, but is deferred until the object file is linked.

Why do things this way? One simple reason: MODULARITY. You can now define your own assembly language module, and convert it into an object file ready to be linked in with the rest of whatever program might be using it. You don't have to worry about the exact address where your memory segments will be located each time you include this module in a new program. This makes it possible to create reusable libraries of common assembly language functions.

The second and third example do exactly the same things because "at" and "org" are synonymous. The latter is included only for compatibility with other assembly language programs and for familiarity's sake, but I strongly recommend you using the former. It simply reads nicer.

The effect of this operation is to set the current segment to "xdata" and the current location to 0x8000.

The last two examples are equivalent to one another and set the current location to 50 without changing the current segment.

At the very start of assembly, the current segment is set to the first segment ("code"), and the address is left indefinite. When different modules are linked together, the linker will attempt to take all the segments of each type and place them in non-overlapping areas of memory, shifting the relative segments around as needed to accomplish this goal.

What if you want to control the placement of objects, say to exclude addresses 0 to 4000 hex? An easy way is to simply write up a module to the effect:

```
seg code at 0
ds 4000h
```

assemble it separately and link it in with any program where you want to reserve this address space.

The linker tries to place your segments in exclusive areas in as tight a fit as possible. So this module will result in the address space 0 to 4000 being excluded from the rest of your program.

The segments types supported by this 8051 assembler are the following:

- * code --- the 8051 code address space, ranges from 0 to ffff hex.
- * xdata -- the external data address space, same range.
- * data --- the internal data/register space. Ranges from 0 to ff.
Only addresses under 80 hex can be used in mnemonics involving direct addressing.

Other segment types are internally used by the assembler. They are:

- * sfr ---- the Special Function Register space -- ranges from 80 to ff.
- * bit ---- the bit addressible address space. These comprise the individual bits in registers 20(hex) to 2f(hex), and the sfr addresses (hexadecimal) 80, 88, 90, 98, ..., f0, f8.

Defining a new segment with one of these types will result in an error.

(c) Defining new LABELS -- LABEL equ Exp, LABEL Type Exp,
LABEL:

LABEL set Exp, LABEL = Exp

These operations are defined as follows:

LABEL equ Exp

defines a constant value LABEL and sets it to the value Exp.

LABEL Type Exp

defines a constant address "LABEL" of the indicated type and sets it to the address given by "Exp". The types recognized

by this assembler are: code, xdata, data, sfr, and bit.

LABEL:

sets a constant address "LABEL" to the current address in the current segment.

LABEL set Exp

defines a variable, LABEL, and sets it to the value Exp.

LABEL = Exp

the same thing as "set".

The following assembly language fragment is an illustration of these operations:

```
seg code at 0
Start: ds 0x4000
Size equ $ - Start
End code Start + Size
```

The first statement sets the current segment and location to "code" and 0.

The next statement is preceded by the label, "Start:". This is equivalent to the statement:

Start code \$.

What it does is define "Start" as a code address, and sets it to the current location (which is 0). Following this is an instruction to reserve 4000(hex) units (bytes) of storage. After this operation, the current location is now 0x4000.

The third instruction sets the numerical constant "Size" to 0x4000 - 0, or just 0x4000. The final directive defines a code address with the name "End" and sets it to the address Start + Size (or just 0x4000).

Variables differ from constants in that they can be redefined. Constants cannot be redefined.

(d) Numeric labels

One can also define anonymous numeric labels, as in the following example:

```

1: cjne A, #0, 1f
   inc A
   movx @DPTR, A
   inc DPTR
   mov A, @R1
   inc R1
   jz 2f
   sjmp 1b
1: setb C
   ret
2: clr C
   ret

```

Each occurrence of "1:" stands for a unique anonymous label, likewise for "2:". Any number may be used in this way to denote an anonymous label.

When a label is referenced by the number followed by an "f", then the first matching numeric label IN THE CURRENT SEGMENT forward of the current location is being referred to. In the example above, 1f and 2f refer respectively to the occurrences of 1: and 2: toward the end of the example.

When a label is referenced by the number followed by a "b", then the first matching numeric label IN THE CURRENT SEGMENT behind the current location is being referred to. In the example above, 1b refers to the 1: at the top of the example.

Thus, this segment is equivalent to the following:

```

X1: cjne A, #0, Y1
   inc A
   movx @DPTR, A
   inc DPTR
   mov A, @R1
   inc R1
   jz Y2
   sjmp X1
Y1: setb C
   ret
Y2: clr C
   ret

```

This feature saves you from the burden of defining needless names for labels that really serve as nothing more than place-holders.

(e) Declaring GLOBAL labels -- global, public

Any constant directive:

```
LABEL equ Exp
LABEL Type Exp
LABEL:
```

can be prefixed by "global" or "public" to result in:

```
global LABEL equ Exp
global LABEL Type Exp
global LABEL:
```

or

```
public LABEL equ Exp
public LABEL Type Exp
public LABEL:
```

What this does is to make these labels visible to modules other than the one where these labels are defined. By default, all labels are visible only in the file where they are used.

(f) Declaring EXTERNAL labels -- extern Type LABEL, ..., LABEL
extern equ LABEL, ..., LABEL

For each global label defined in a *.s module file, a corresponding external declaration should be made in whatever other module this label is to be used. Typically, one will make these and other related declarations in a *.h file and then INCLUDE this file in whatever module needs the declarations. The type must match the type of the label being referenced, if it is an address, or it must be "equ" if the label referenced was a numeric constant.

For example if one declared global labels in a module Kernel.s as follows:

```
public STACK_BASE data 0x80
```

```
...
```

```
seg code
```

```
public Spawn:
```

```
....
```

```
public Resume:
```

```
...
```

one would generally make the corresponding declarations:

```
extern data STACK_BASE
```

```
extern code Spawn, Resume
```

in a header file (say, Kernel.h), and then include this file in any source

module where the addresses STACK_BASE and Spawn might be needed.

(g) Memory ALLOCATION -- ds, rb, rw

The following operations can be used in any segment. They are generally used to allocate space for objects and so are generally used in conjunction with "LABEL:" type definitions. These are examples:

```
seg code at 0
BASIC_SEG: ds 0x4000
```

```
seg xdata
Byte: ds 1
ByteArray: rb 5
WordArray: rw 5
```

The first example reserves 0x4000 units (bytes) in the current segment for the variable BASIC_SEG and then increments the current location by 0x4000. Basically, this operation behaves as if the assignment "\$ = \$ + 0x4000" had just been carried out.

Both "ds" and "rb" are exactly equivalent, but the latter more descriptively states: reserve single-byte units. So the second example reserves 1 byte for the variable "Byte", and 5 bytes for "ByteArray".

NO MEMORY IMAGE IS GENERATED FOR ANY SPACE SKIPPED BY ds/rb/rw.

The third example is equivalent to:

```
WordArray: rb 10
```

Each unit following a "rw" is a word, which consists of two bytes.

(h) Memory FORMATTING - db, dw

These operations can be used in the code segment only. They are the only directives that can generate memory images. The only other operations that generate memory image output are the 8051 mnemonics, which likewise are restricted to the code segment only.

Two purpose served by these operations is mainly to initialize data, examples:

```
ByteArray: db 'a', 'b', 'c', 'd', 'e'
String:   db "This is a string", 0
```

In the following examples:

```
db 0x20, "String", 'c'
dw 0x1234, 0x5678
```

the first operation lays out the byte 0x20 and equivalent character codes for 'S', 't', 'r', 'i', 'n', 'g', and 'c' in that order. The current location is then incremented by 8 to the location following the last item.

The second operation is equivalent to the following:

```
db 0x12, 0x34, 0x56, 0x78
```

It formats 2-byte word units into memory.

Both of the operations: db, and dw can be followed by a comma-separated series of numeric values or addresses. In addition, db can accept strings, as shown in the examples above.

(i) **CONDITIONAL assembly** -- if (Ex) ST, if (Ex) ST else ST

These statements are used to selectively assemble different sets of statements. For example

```
if (STAND_ALONE) {
    at 0x03
    mov R0, #SP_IE0
    acall Pause
    reti
} else {
    at 0x4003
    pop PSW
    mov R0, #SP_IE0
    acall Pause
    reti
}
```

will assemble the first set of statements (at 0x03 ... reti) if the label STAND_ALONE is anything other than 0, and the second set (at 0x4003...reti) if the label is 0.

An example with the exact same effect could be written as:

```
if (STAND_ALONE) SEG equ 0; else SEG equ 0x4000
at SEG + 3
if (!STAND_ALONE) pop PSW
mov R0, #SP_IE0
acall Pause
reti
```

Both the if and else part of the conditional will accept only one statement. If more than one statement needs to be included, as in the first example, then they can be grouped within curly braces.

(j) Statement GROUPING -- { ... }, multiple statements on a line.

Any sequence of statement included within a matching set of curly brackets is treated as a single statement. It can then be used in the body of any conditional just like any single statement can.

SPECIAL NOTES ON STATEMENT FORMATTING:

(A) ALL STATEMENTS (a) THROUGH (h) MUST END IN SEMICOLONS.

However, this semicolon can be elided if it is the last item on a line. This allows compatibility with more traditional one-statement-on-a-line type assemblers. So normally, you don't have to even concern yourself with this if you adhere to one-statement per line style.

(B) A BASIC STATEMENT ((a) THROUGH (h)) MUST BE WRITTEN ALL ON ONE LINE

It cannot be split up into two or more lines.

(C) ALL COMMENTS ARE IN C++ STYLE.

Many assemblers use the semicolon to initiate comments. I have decided against this feature in favor of making this assembler more compatible with C++ syntax. Comments occur in the following two forms:

(a) Anything included between a matching pair /* ... */

(b) Anything included between a // and end of line.

However, for increased compatibility, I also allow the following format:

(c) Anything included between a ;; and end of line.

My personal style is to precede comments with a ;;, so none of this impinges on the software included in the archive with the assembler.

There is a short C-program included that will blindly convert all single semicolons to double semicolons. Since I've observed that semicolons rarely occur inside string or character constants in actual 8051 programs, this should ALMOST always be sufficient to resolve any incompatibilities with your older assembly language programs.

(n) What goes in a *.s file, what goes in a *.h file?

Generally speaking, declarations should be placed in a *.h header file. The design of this assembler (especially with it being a one-pass

assembler) is intended to support this usage. Any of the following is a declaration:

- (c) Defining new LABELS -- LABEL equ Exp, LABEL Type Exp
- (f) Declaring EXTERNAL labels -- extern Type LABEL, ..., LABEL
extern equ LABEL, ..., LABEL

Declarations only meant to be accessed within one module should be made inside that module, instead of out in a header file.

The following should be used only in *.s files, as they are generally

(a) used to create memory images, (b) used to define non-global objects, or (c) used to define address values:

- (a) FILE INCLUSION -- include FILE
- (b) Setting current SEGMENT and LOCATION -- seg, at, org
- (c) Defining new LABELS -- LABEL:
- (d) Numeric labels
- (e) Declaring GLOBAL labels -- global
- (g) Memory ALLOCATION -- ds, rb, rw
- (h) Memory FORMATTING - db, dw

The last two items are generally used in many different contexts, and so can be used anywhere:

- (i) CONDITIONAL assembly -- if (Ex) ST, if (Ex) ST else ST
- (j) Statement GROUPING -- { ... }

(3) Expressions

(a) Operators

The syntax is the same as in C. The following operations are defined:

BIT-WISE: ~, &, ^, |, <<, >>

BOOLEAN: !, &&, ||, <, <=, >, >=, ==, !=

CONDITIONAL: ? :

ARITHMETIC: prefix + and -, +, -, *, /, %

CONVERSIONS: high, low, by

BIT CONVERSION: .

The operator precedences are all the same as in C.

The latter two groups, not defined in C, are described in more detail below. The operator high, and low have the same precedence as all the other

prefix operators (+, -, !, and ~). The operators "by" and "." have the lowest precedence of all infix operators, so for example

$A * B \text{ by } C$

is resolved as:

$A * (B \text{ by } C)$

and

$A.B + C$

as:

$(A.B) + C$

Parentheses may be used to enclose expressions as in C, for example:

$A + ((B \ll 2) \& (C \gg 3))$

(b) CONVERSIONS ... high X, low X, H by L

The following examples illustrate these operations:

high 1234h	(result: 12h .. the upper byte of the word 1234h)
low 1223h	(result: 34h .. the lower byte of the word 1234h)
12h by 34h	(result: 1234h)

(c) BIT-CONVERSION ... Dir.Pos

This is an 8051-specific operation related to the bit-addressing structure of the processor. The first argument represents a direct data register (of type "data" and value < 80h, or type "sfr" and value >= 80h). The second represents a bit position (0, through 7).

The register, Dir, must be bit addressible. These include only:

data; 20h - 2fh
sfr: 80h, 88h, 90h, 98h, 0a0h, 0a8h, 0b0h, 0b8h,
0c0h, 0c8h, 0d0h, 0d8h, 0e0h, 0e8h, 0f0h, 0f8h

The sfr registers and bit positions generally have meanings defined by the manufacturer of the 8051 processor and vary between different versions of the 8051. They are not generally free to be defined by the programmer for arbitrary use. Most of them control or monitor the internal 8051 peripherals.

(d) LOCATION COUNTER -- \$

A variable address that denotes the current location within the current segment. NOTE:

$\text{dw } \$, \$ - 2, \$ - 4$

IS EQUIVALENT TO:

dw \$; dw \$ - 2; dw \$ - 4
which is equivalent to:
1: dw 1b; dw 1b; dw 1b

The location counter advances in the middle of a dw or db.

(e) NUMERIC CONSTANT

This assembler accepts both C numeric syntax, as well as the Intel numeric syntax. The relation between the (extended) C notation and Intel notation is illustrated below:

HEXADECIMAL: 0xa44f = 0a44fh
 0x23 = 23h
DECIMAL: 23 = 23
 23 = 23d
OCTAL: 034 = 34q
 056 = 56o
BINARY: 0b1001 = 1001b

Upper case may be used anywhere lower case is used, so the above can be written as:

HEXADECIMAL: 0XA44F = 0A44FH
 0X23 = 23H
DECIMAL: 23 = 23
 23 = 23D
OCTAL: 034 = 34Q
 056 = 56O
BINARY: 0B1001 = 1001B

(f) LABELS

Labels may consist of any sequence of letters, the `_`, and digits, not starting in a digit. As with numbers, labels are CASE INSENSITIVE. So all of the following are equivalent:

PPC, PPc, Ppc, pPC

(4) Referencing Expressions

At any time during assembly, a label may be in one of 3 states:

(a) DEFINED and ABSOLUTE:

This is either a numeric label, or a label denoting an address whose actual value is known.

(b) DEFINED and RELATIVE:

This is a label denoting an address whose location within its segment is known, but with the segment being relative.

(c) UNDEFINED:

This is a label that is either defined elsewhere in another file, or defined later on in the file currently undergoing processing.

The following restrictions hold when using expressions:

- * Only ABSOLUTE labels can be used in any of the directives:
at/org,
ds/rb, rw
if (...)
- * Only DEFINED labels can be used on the right-hand side of any of the following directives:
Label equ Exp,
LABEL Type Exp
LABEL set Exp, LABEL = Exp
- * Any expression can be used with any image generating statement:
Mnemonics
db, dw

If the expression's value is not known at the time of assembly, then the corresponding location in the image is zeroed out. If the expression's value becomes known by the time the file is processed, the assembler will go back and fill in the zero with the appropriate value(s).

(5) Bugs (or "features")

(a) There is no way to tell the assembler to locate relatively addressed data registers in the directly addressible space. Consequently you may receive numerous errors during the linking phase telling you that such and such registers cannot be directly addressed.

There are basically 2 ways to resolve this: (1) give the registers absolute addresses, (2) try listing the files in which these registers are defined first. The linker maps relative segments from the files in the order you list those files.

In the makefile of the sample program provided (in 8051/assem/data), the linking phase is done with the command line:

```
cas -o math.o data.o stdio.o kernel.o
```

This ordering resolves the problem.

(b) The assembler won't recognize UNIX-style newlines on a DOS. Therefore, a conversion utility (nl.c) has been provided.

- (c) No run-time checks are made against the object files processed.
A corrupt object file will crash the assembler during the linking phase.

Documentation of the debugger :

8051 DISASSEMBLER, notes on use.

(0) COMMAND LINE, INPUT FORMAT

This is a disassembler for the Intel 8051-family of processors. It is currently set to read up to and including the 8052.

The typical command line for this disassembler is:

```
das <asm.hex>asm.s
```

The input file (read from standard input) is assumed to be in Intel Hex Format. The output is placed in standard output (or in a file when redirection is used). A third file:

entries

is used to specify a list of entry points for das to use.

Currently, it will only accept input in Intel Hex format located between the addresses 0000 and 4000 (hex). It will read entry points from a file: "entries" and recursively disassemble all the addresses "reachable" from those listed as entry points. A sample entries file, input file and output file are provided with the software. The format of each address in the file is (starting at column 1):

```
HHHH<cr>
```

with the H's representing the digits of the address in hexadecimal.

Typically entry points are assigned to 0000, the starting address, and the address of each interrupt-handler that is used.

(1) OUTPUT FORMAT

The output of the disassembler is not exactly re-assembleable. When the disassembler searches for reachable segments of code, it will fail to resolve indirect jumps (jmp @A + DPTR, jmp @A + PC), and will fail to recognize virtual jumps created by manipulating the stack pointer, e.g.

```
push DPL
```

```
push DPH
```

```
ret      ;;; An indirect jump to the address pointed to by DPTR.
```

In addition, if the original program contained data tables, these would normally be passed by untouched.

The disassembler will regard anything in the allocated code space that was not accessible as being data. If necessary, you will have to visually inspect the disassembled code for indirect jumps and stack pointer manipulations to determine what addresses are being indirectly referenced. Then you could add these addresses to the entry point list in the file "entries" and disassemble again. This, in turn, may reveal additional indirect addresses, so that there is an iterative process implied here to completely disassemble code.

For assistance, a list of locations where indirect jumps occur will be listed in the following format:

Indirect jump at <Address>

In the sample program provided, there were no indirect jumps. However, since the assembly-language software implements a multitasking kernel, it heavily manipulates the stack pointer. Consequently, the addresses listed in the entries file are not just the starting address (0000), and interrupt handler(s) (0023), but also derived addresses: 0067 (found on the first iteration). and 0131 (found on the second).

Also, any addresses referenced that lie outside the range of the Intel Hex Format file will be listed as external references. These are listed in the following form:

REF: <Address>

Data segments will be clearly marked as such, beginning with the header:

DATA at <Address>

following which will be a hexadecimal listing of the "data" in the format illustrated below:

```
34 aa ef 00 00 00 00 00 34 ee ff 00 00 00 00 00 |4      4      |
```

A list of up to 16 bytes in hexadecimal notation is generated, and then between vertical bars is the list of corresponding ASCII characters. Control characters and meta characters (those ranging from 128 to 255) are "blanked" out. The character listing allows you to easily recognize embedded string constants wherever they occur.

Labels are generated for SFR's corresponding to the 8052 processor, and Bit labels are likewise generated. These are listed in the tables SFRs, and Bits contained in the disassembler source, and can be modified to suit whatever version of the 8051 you are using.

In the output, addresses are labeled in the format illustrated below:

C3458

The initial letter indicates the number of times this address is referenced with the following key:

B = 1 reference, C = 2 references, D = 3, E = 4, etc.

the next 4 items represent that actual address of the label in hexadecimal.

(2) ERRORS

The disassembler will duly note where disassembled instructions overlap, or where it cannot further process input.

No entry points listed.

Bad hexadecimal digit in input.

The input file "entries" was not found, or contained corrupt data.

Address out of range 0 - 4000h in input.

The input file contains a portion of a program lying outside the address range 0 to 4000 (hex). The disassembler cannot go beyond 4000h. This is a fatal error, disassembly stops.

Bad format string, PC = HHHH.

Something's wrong with your compiler or operating system, or the source file has been corrupted. Report this promptly to me, if you should see this.
:)

Bad hexadecimal digit in input.

Bad checksum.

Unexpected EOF

The input file contained a portion not strictly in Intel Hex Format.

Entry into ARG at <Address>.

An entry point is accessing a point that has already been disassembled as the interior of an instruction.

OP into ARG at <Address>.

An attempt was made to disassemble an instruction starting at an address already contained in the interior of another instruction. The disassembler is generating overlapping instructions.

ARG into OP at <Address>.

An attempt was made to disassemble an argument to an instruction at an address already disassembled as the first byte of another instruction.

ARG into ARG at <Address>.

An attempt was made to disassemble an argument to an instruction at an address already disassembled as an argument of another instruction.

The presence of this error indicates that something is wrong with your compiler or operating system, or the source file has been corrupted. Report this promptly to me, as well, if you should see this error.

Too many entries, PC = <Address>.
The number of pending points for the disassembler to process exceeded its capacity.

(3) USER SUPPORT and GUARANTEE

This software has been rigorously verified with respect to the description given above. It will also conform to the description of the binary coding of the 8051 processor supplied by Intel.

Since this was a hobby project mainly intended for myself and not for a wider audience, you will be on your own in using this software. However, I am interested in hearing any feedback you have. Send comments to my e-mail address, markh@csd4.csd.uwm.edu. If you want to make upgrades, I encourage you to try your hand at it. That's exactly why I'm distributing this package free.

Documentation of the Object file format utility

OBJECT FILE FORMAT, and OBJECT FILE DISPLAY UTILITY

(0) COMMAND LINE, INPUT FORMAT

This utility will display the contents of an object file created by the assembler. The command line is typically of the form:

ds file.o >file.x

The contents of an object file are as follows:

HEADER, IMAGE, FILES, SEGMENTS, GAPS, SYMBOLS,
EXPRESSIONS, RELOCATIONS

(a) HEADER

Bytes	Item
2	55aa ... the MAGIC number
4	Location of first item past the end of the memory IMAGE.
4	Number of FILES listed.
4	Number of SEGMENTS listed.
4	Number of GAPS listed.
4	Number of SYMBOLS listed.
4	Number of unresolved EXPRESSIONS listed.
4	Number of RELOCATION items listed.
4	Check sum of the previous 7 items.

(b) IMAGE

This is a binary listing of each of the code segments contained in this object file, all concatenated together. This binary format is listed by the display program 16 bytes per line, each byte written in hexadecimal form. It is also displayed in character form to the right-hand side, in order to show off any embedded strings.

The size of the image is given in two ways: (a) it is the sum of the sizes of all the code segments listed minus all the sizes of all the gaps listed, and (b) it is the number of bytes contained between the end of the header and the end of the image (which is indicated in the header).

(c) FILES

Source and header file names are added to the object file as needed to help the linker determine where any errors may be originating from. Each file is listed in the following form:

Bytes	Item
1	L = length of file name
L	file name

(d) SEGMENTS

The segments 0 through 4 are only used internally by the assembler. They have the types respectively code, xdata, data, sfr, and bit. Their sizes are 0, bases are 0, and they are absolute. These segments serve as default segments for absolute addresses.

The remaining segments (if there are any) are listed in the following format:

```
## Line File Rel Type Base Size Loc
```

which indicate, respectively, the segment's number, the line and file where it is first defined, whether it is relative, the segment's type (0 = code, 1 = xdata, 2 = data, 3 = sfr, and 4 = bit, 3 and 4 are never seen). The Base is 0 for relative segments, else it points to the actual starting address of the segment. The Size is the number of bytes in the segment, and (for code segments) Loc is the place in the object file where its image is found.

Each segment is formatted in the object file as follows:

Bytes	Item
2	Line
2	File
2	Bits 0-7 = Type
	Bit 8 = 1 for relative segments, 0 else.
2	Size
2	Base
4	Loc

The file number indicated indexes the appropriate entry in the file list previously described.

(d) GAPS

Gaps are subsets of segments that are reserved but that have no corresponding memory image. No memory output is generated for gaps. They are displayed in the following format:

Seg Off Size

where Seg, and Off are the segment where the gap occurs and its offset within the segment. Size is the size of the gap.

Two constraints will apply for valid object files: (1) every gap will be located entirely within its segment ($\text{Off} + \text{Size} < \text{segment's size}$), and (2) only code segments contain gaps.

Each gap is formatted in the object file as follows:

Bytes	Item
2	Segment index
2	Offset
2	Size

(e) SYMBOLS

Each symbol is displayed in the following format:

Scope Var Type Value Name

where ## is the symbol's index, Scope is defined in terms of the GLOBAL and DEFINED attributes as follows:

Scope	GLOBAL	DEFINED
undefined		
local		X
external	X	
global	X	X

Var is marked if the symbol is a variable. Variables and global/externals are disjoint. Its type and value will be displayed as one of the following:

ADDR Seg Offset
NUM Value

The first case indicates a symbol defined as an address. Its segment index and offset are then listed. The second case indicates numeric value. The actual value is then listed. Offset and Value are listed in hexadecimal format. Neither the Offset nor Value are applicable if the symbol is not defined (external or undefined). Seg is inapplicable if the symbol is totally undefined.

Following this is the symbol's name.

Each symbol is formatted in the object file as follows:

Bytes	Item
1	Bit 0: 1 if variable, 0 if constant Bit 1: 1 if address, 0 if numeric Bit 2: DEFINED Bit 3: GLOBAL 2 Segment index
2	Offset/Value
2	L = Length of symbol name
L	Symbol name

(f) EXPRESSIONS

These items represent expressions that could not be evaluated from the information presented in the source files. Expression evaluation is deferred to the linker if the expression contains relative addresses, or externally defined symbols.

They are displayed as follows:

```
## Line File Tag Args...
```

where the first 3 items are respectively the expressions's index, and the line and file where it is defined. Tag and args will take on one of the sets of values:

Description	Tag	Args
Number	0	Value
Address	1	Seg Offset
Symbol	2	Symbol Name
Unary operation	3	Unary A
Binary operation	4	Binary A B
Conditional	5	A B C

where A, B, and C are indexes of previously defined expression occurring earlier in the list, Symbol Name is the name of the symbol which the expression is set to, Seg and Offset are the segment and offset of an address expression, Value is the numeric value of a numeric expression, and Binary and Unary are symbolic number for operators defined as follows:

Unary Operators	
Operator	Value
~	27
!	28
high	29
low	30
+	31
-	32

Binary Operators	
Operator	Value

+	31
-	32
*	33
/	34
%	35
<	36
<=	37
>	38
>=	39
==	40
!=	41
&&	42
	43
&	44
^	45
	46
<<	47
>>	48
.	49
by	50

The format of an expression in the object file is as follows:

Bytes	Item
2	Line
2	File
1	Tag

with the following cases:

(i) Numeric (Tag = 0)	Number
Bytes	Item
2	Value
(i) Address (Tag = 1)	Segment:Offset
Bytes	Item
2	Segment Index
2	Offset
(i) Symbol (Tag = 2)	Symbol
Bytes	Item
2	Symbol Index
(i) Unary operation (Tag = 3)	Op A
Bytes	Item
1	Operator
2	Expression A's index.
(i) Binary operation (Tag = 4)	A Op B
Bytes	Item
1	Operator
2	Expression A's index.
2	Expression B's index.

(i) Conditional operation (Tag = 5) A ? B : C

Bytes	Item
2	Expression A's index.
2	Expression B's index.
2	Expression C's index.

(g) RELOCATIONS

A relocation item indicates an unevaluated expression, some information about its addressing mode, and the segment and offset where it is to be patched. These are items that the assembler could not properly determine because external references were used or relative addressing was used. All relocations refer to positions within code segments, since only code segments have memory images.

A relocation item is displayed as follows:

Line File Tag Exp Seg: Off

where Line and File determine the corresponding point in the source file, Seg and Off refer to the location where the patch is to be applied, Exp is the item to be patched, and Tag is the type of relocation defined as follows:

Tag	Description
b	Byte-sized constant
w	Word-sized constant
B	Bit address
D	Direct register address Data address < 80h or SFR address >= 80h.
R	Relative code address
P, Q	Paged code address.
L	Absolute (long) code address.

Relocation items are formatted in the object file as follows:

Bytes	Item
2	Line
2	File
1	Tag
2	Expression index
2	Segment index
2	Offset

List of reported errors

Diagnostics

The assembler will only print out a screenful of diagnostics (24) before it stops. When applicable, the file and line number where the error occurred are given. An attempt is made to indicate the line where the error actually occurred, rather than where its effects were first seen.

Assembly is stopped at certain points if errors have been detected, since it would make no sense to go on past the given point.

The following is an alphabetized listing of diagnostics classified by whether they are fatal or non-fatal. Assembly always stops after the former but will continue after the latter (until the error limit is reached).

Internal errors should be reported to me, see the README file at the FTP site `csd4.csd.uwm.edu` in the directory `/pub/8051` or `/pub/micro/8051` for more information. If you get an internal error, you should display the object file using the object file display program provided with the assembler. If you want me to resolve the bugs (if any exist) I would need to see this as well.

Alternatively, the source for the assembler is yours. If you want to make modifications, try your hand at it, and if something significant results let everyone know by placing it in the public domain.

(a) Fatal Errors

These diagnostics will cause assembly to stop immediately, they usually represent serious inconsistencies or serious points of confusion for the assembler that makes further processing impossible or pointless.

Address <Decimal Number> out of range

The location counter points to the indicated address, which lies outside the address space of the current segment. These address spaces are as follows:

Type	Low	High
code	0	65535 (or ffff hexadecimal)
xdata	0	65535 (or ffff hexadecimal)
data	0	255 (or ff hexadecimal)

Attempting to write to a non-code segment.

This happens if you try to use any image generating statement (db/byte, dw/word, or any opcode) in a data or xdata segment. Only code segments will accept these statements.

Bad opcode initialization.

This is an internal error you should never see. It indicates an internal inconsistency in the assembler's opcode tables. These tables are used to process the 8051 mnemonics.

Cannot fit segment.

The linker tries its best to fit all the relatively addressed segment in the tightest fitting space it can. It may fail to pack them all in, if they can only fit real tightly, or your segments may actually not fit at all. Use

the object file display program that accompanies this assembler to look at your object files in some detail.

Cannot open <FILE>

An file was lost sometime during processing, or the file named probably does not exist, or is not readable.

Cannot open object file for <FILE>.

The object file for the source file indicated was somehow lost or made inaccessible between the time it was created and the time it was processed by the linker.

Cannot open output file.

For some unexpected reason, the assembler could not create an .hex output file. If you're running this on a UNIX, then you're probably assembling output into a directory where you have no write access. Also possible: you ran out of disk space.

Cannot reopen <FILE>

A failure to reopen the current file after processing an include file occurred. This indicates either an internal flaw in the assembler or that something happened to the file during assembly.

Corrupt object file <NAME>.

A rudimentary check are made on object files. The file indicated didn't pass that check. It's was either corrupted, or the assembler itself has an internal flaw that made it produce corrupt object files.

Could not restore <FILE>'s position.

A failure to resume processing the current file past the most recent include statement occurred. This indicates either an internal flaw in the assembler or that the file undergoing processing was corrupted during assembly.

Could not save <FILE>'s position.

A transient error occurred while setting up to include another file. This is either a serious system flaw, or an internal flaw in the assembler.

Expression too complex ... aborting.

If you see this, you must be Lieutenant Commander Data. An expression has to be pretty complicated before it confuses the assembler. Break it up by symbolically defining subexpressions first.

Internal error (0).

An inconsistency was detected by the linker in the object file undergoing

processing. A relocation item of undefined type was found.

Internal error (1).

An inconsistency was detected by the linker in the object file undergoing processing. An undefined expression was found.

Internal error (2).

A flaw in the assembler made it lose track of all the code segments, even though at least one was defined.

Internal error (3).

A flaw in the assembler made it think that a ds/rb, or rw statement was reserving memory outside the segment where it was defined.

Internal error (4).

A flaw in the assembler caused it to try and relocate two or more items to the same address. This can also be caused by a corrupt object file.

Internal error (5).

A flaw in the assembler caused it to have trouble resolving an expression during linking. This can also be caused by a corrupt object file.

Internal error (6).

A flaw in the assembler made it fail to sort the list of items to be relocated. A corrupt object file can cause this too.

Internal error (7).

A flaw in the assembler made it think it was trying to patch up a hole of size other than 1 or 2 bytes. This can also be caused by a corrupt object file.

Internal error (8).

The assembler is expecting an object file larger than what it sees. This is the result of either an internal flaw or corrupt object file.

Invalid object file <NAME>.

The first type bytes of all object files are 55 and aa (hexadecimal). The object file undergoing processing doesn't show this. It's was either corrupted, or the assembler itself has an internal flaw that made it produce corrupt object files.

Invalid option: -<OPTION>

Only -c and -o are recognized.

Missing ';' after include statement.

The include statement (like all other statements) needs to be followed by a semicolon, or otherwise be the last statement on the line.

Missing filename in 'include'

The include statement has the form: include "FILE", with the file name enclosed in double-quotes, as indicated.

Out of memory.

The assembler's resources exceeded the system's available space.

If this happens during linking it may point to the high degree of interconnectivity between modules thereby indicating a need to make the interfaces between modules more clean-cut. If it happens during assembly it may point to a need to break the assembly-language file into smaller modules.

Overlapping segments: <FILE> [<LINE>] and <FILE> [<LINE>].

The segments in the indicated places have overlapping ranges of addresses. Move one or both of them, if possible.

Overlapping segments at [<LINE>] <FILE> and [<LINE>] <FILE>.

This one is actually an internal error.

Statement too complex

If you see this, you must be Lieutenant Commander Data. An statement has to be pretty complicated with many layers of if's and else's and brackets '{' and '}' before it confuses the assembler. There is no direct recourse here other than to try and simplify your program.

Too many nested include files.

Up to 5 levels of nested includes are allowed.

Too many segments.

Up to 32 segments can be defined per file. Any occurrence of the org/at or seg statements will start a new segment. The first 5 segments are defined by and used internally by the assembler, which leaves you with 27.

Undefined segment type.

Only the types: code, xdata, or data can be used with the seg statement.

Unexpected EOF.

Attempted to read past the end of an object file. The object file is corrupt

Use: <CAS> -c File... to assemble.

At least one input file should be listed with the -c option.

Use: <CAS> -o Output File... to link.

At least one source or object file should be listed with the -o option.

Use: <CAS> [-o Output] File... to link.

At least one source or object file should be listed with the -o option or with no options.

(b) Non-Fatal Errors

These are errors the assembler will attempt to recover from and continue processing after.

Address <NAME> redeclared as number: <FILE> <FILE>

The symbol indicated was declared as a global number in the first indicated files and as a global address in the second.

Address cannot appear in: x? x: x.

Addresses cannot be used as conditions of conditionals in the combination

address? x: x. Only numeric values can be used there.

Address cannot be used with infix operator.

The only allowable combinations of addresses involving infix operations are pretty much like in C:

number + address

address + number address - number

In addition, the combination reg.bit

can be used for bit addressable registers "reg" and values of 0 to 7 for "bit".

Address cannot be used with prefix operator.

Prefix operators other than + (-, !, ~, high, low) can only be used with numeric values, not addresses.

Address of wrong type in reg.pos

Only data or sfr addresses can be used in the place referred to ("reg").

Address type mismatch.

Rudimentary type-checking is done for operations involving bit addresses, data and sfr addresses, or code addresses. Type checking is not done on xdata addresses, as there are not 8051 operations that directly involve such addresses.

Addresses of different types cannot be subtracted.

Addresses must be of the same type in order to be subtracted. They can be of different segments, however, as long as the segments have the same type.

Attempting to make local label global.

Local (numeric) labels cannot be made global.

Attempting to redefine <NAME>

The label indicated was used in the context LABEL:, but is already defined,

Bad hexadecimal character.

A hexadecimal character takes the form: \xHK or \XHK, where H and K are hexadecimal digits. Exactly two hex digits must be seen. An \x by itself without any following hexadecimal digits is interpreted as just x.

Binary number has non-binary digits.

A number with a binary prefix (0b, 0B) or binary suffix (b, B) can only contain the digits 0 and 1.

Bit address out of range.

Bit addresses range only from 0 to ff hex (0 to 255 decimal).

Bit position out of range.

The operation reg.pos only allows values of 0 to 7 for "pos".

Byte value out of range (hex value).

An immediate byte value in the range -80 (hex) to ff (hex) (or -128 to 255) is required in this context.

Cannot use @A+A

Only @A+DPTR, and @A+PC can be used with the 8051 language.

Cannot use @A+AB.

Only @A+DPTR, and @A+PC can be used with the 8051 language.

Cannot use @A+C.

Only @A+DPTR, and @A+PC can be used with the 8051 language.

Cannot use @A+Rn

Only @A+DPTR, and @A+PC can be used with the 8051 language.

Cannot use @A.

The only valid combinations using @ are @R0, @R1, @DPTR, @A+DPTR, and @A+PC.

Cannot use @AB.

The only valid combinations using @ are @R0, @R1, @DPTR, @A+DPTR, and @A+PC.

Cannot use @C.

The only valid combinations using @ are @R0, @R1, @DPTR, @A+DPTR, and @A+PC.

Cannot use @PC.

The only valid combinations using @ are @R0, @R1, @DPTR, @A+DPTR, and @A+PC.

Cannot use PC as a register.

PC only appears in the combination @A+PC in the statement movc A, @A+PC.

Code address out of range.

Code addresses can only range from 0 to ffff hex (or 0 to 65535 decimal).

Data address out of range.

Only direct data register addresses (0 to 7f hex, or 0 to 127), or special function register addresses can be used here.

Decimal number has non-decimal digits.

The number indicated has hexadecimal digits (a through f, or A through F) but no hexadecimal prefix (0x, 0X, 0h, 0H) or suffix (h, H, x, X).

Division by 0.

x / 0 is not defined.

Empty character constant.

A single quote cannot be quoted as a character ("). Use \" instead. Alternatively, you attempted to define an empty string (") with single quotes.

Expected an argument/prefix operator.

The expression indicated is probably missing a prefix operator, a symbol or constant, or an opening parenthesis '('.

Expected segment type.

The statement: seg Type, must have Type code, xdata, or data.

Expected symbol after 'global'

Only symbols can be declared as global.

Expected symbol in 'extern'

Only symbols can be declared as external.

Expected type or 'equ' after 'extern'.

External symbols must be declared as a type (sfr, bit, code, data, xdata) or as a numeric constant (equ). The valid forms are:

extern (sfr, bit, code, data, xdata, or equ) Symbol, ..., Symbol

Expression type does not match.

The expression occurring on the right hand side of the statement of the form: Symbol Type Value, does not have the indicated type.

Extra ','.

An extra command was inserted after the last operand. Alternatively, the operand following the last comma is missing.

Illegal combination: address + address

Only numeric values can be added to or subtracted from addresses. The rules for address arithmetic are basically like those in C.

Illegal combination: number - address

Numbers can only be subtracted from addresses, not the other way around. The rules for address arithmetic are basically like those in C.

Illegal combination: reg.bit

Only numeric values in the range 0 to 7 can be used in place of "bit", not addresses.

Indirect registers are not bit addressable.

The register referred to ("reg") must be a data address in the range 20-2f. The address referred to lies above 80 (hex). It is possible you may have meant to use a Special Function Register here, not a data address.

Invalid addressing mode: <MNEMONIC>

This mnemonic is not defined for this combination of operands.

Missing '(' in 'if (...)'.
The if statement must have the form: if (x) ..., as it does in C.

Missing ')'

An unmatched closing parenthesis, ')', or possibly an extra opening parenthesis, '(', occurred.

Missing ')' in 'if (...)'.
The if statement must have the form: if (x) ..., as it does in C.

Missing ':'

The conditional indicated is missing a colon. Conditional expressions have the form x? x: x.

Missing ';'.

Semi-colons must be used between two or more statements on the same line. You may have mistyped a colon (:) instead.

Missing '}'.

An opening bracket '{' was never closed off with a '}'.

Missing DPTR or PC after @A+

Only @A+DPTR, and @A+PC can be used with the 8051 language.

Missing a ' in character constant.

No more than one item can be enclosed between single quotes (e.g. 'ab', or 'a\n', though '\n' is okay.) You may have meant to define a string here. Use double quotes.

Modulo by 0

x 684s not defined.

Number <NAME> redeclared as address: <FILE> <FILE>

The symbol indicated was declared as a global address in the first indicated files and as a global number in the second.

Octal number has non-octal digits.

A number with an octal prefix (0, other than those with an appropriate suffix), or an octal suffix (o, O, q, Q) can only contain the digits 0 through 7.

Only code, xdata, or data segments allowed.

The types sfr and bit cannot be used with segments.

Paged address out of range.

Paged jump or call instructions (ajmp, acall) must have destinations that lie on the same code page (every 800 hex bytes, starting at 0 is a code page). This only happens when your program gets beyond a certain size, or when it is located at an odd address (such as 7ff hex). Generally, it's a good idea to start your programs on even multiples of 800 hex. Stand-alone programs have to start at the reset address (0) anyway.

Redeclaring address <NAME> as number.

A symbol already defined as an address is being declared as an external numeric value.

Redeclaring local symbol <NAME> as external.

A symbol already defined as a local symbol is being defined as external. This may also be a name clash between a local and global symbol.

Redeclaring number <NAME> as address.

A symbol already defined as a number is being declared as an external address.

Register address out of range.

Only directly addressable data registers (0 to 7f hex, or 0 to 127) can be used here. This error may have inadvertently been created by the assembler if you defined relatively addresses data registers, and the assembler tried to map them beyond 80 hex. See the Bugs section of the document.

Register in @Rn out of range.
Only registers R0 or R1 can be used as register pointers.

Register in reg.bit not bit addressible.
The register referred to ("reg") must be a data address in the range 20-2f (hexadecimal, 32-47 decimal), or must be one of the Special Function Registers whose address is evenly divisible by 8 (in hexadecimal: 80, 88, 90, ..., f0, f8).

Register must appear after @.
Numeric or address expressions cannot appear after @.

Relative address cannot be used here
Only absolute addresses, or numeric values containing objects that have been previously defined (in terms of other such absolute objects) can appear in the conditional (if (E) ...), or in address setting/modifying statements (org/at, ds/rb, rw).

Relative address out of range.
Some jump instruction (cjne, djnz, sjmp, jz, jnz, jc, jnc, jbc, jb, jnb) require their destinations to be within a certain range (-80 hex to +7f hex) of the first byte following the jump instruction. Generally, you should not use these operations to jump across segment boundaries when one or more of the segments is relatively addressed.

SFR address out of range.
Special function register addresses can only have values in the range (80 to ff hex, or 128 to 255 decimal).

Symbol <NAME> already declared as external number.
The symbol indicated was declared as an external numeric constant but was defined as an address.

Symbol <NAME> already declared as external.
The symbol indicated was already declared as an external/global, and so cannot be defined as a variable, or local constant. If you had intended to define it as a global constant, precede the definition with the word 'global' or 'public'.

Symbol <NAME> already declared as number.
The symbol indicated was used in the context LABEL:, but is already declared as a numeric constant.

Symbol <NAME> already defined as constant.
Constant symbols cannot be redefined as variables.

Symbol <NAME> already defined.
Constant symbols can only be defined once.

Symbol <NAME> cannot be equated to relative address.
The symbol indicated was declared as an external numeric constant. It can only be set to an absolute address or numeric value.

Symbol <NAME> cannot be set to relative address.
Variables can only be set to absolute addresses, or numbers.

Symbol <NAME> redefined: <FILE> <FILE>.
The symbol indicated was defined as a global in both of the files listed.
It can only be defined as a global once.

Symbol <NAME>'s type does not match.
The symbol indicated was declared as an external of one type and defined as a global of a different type.

Syntax error
This statement so totally eluded the assembler's ability to process it, that it didn't even know what you intended. This can also occur if it gets set off-track as a result of previous errors.

Too many arguments specified.
More operands were supplied than is accepted by this mnemonic.

Too many gaps.
Each time you issue a statement to the assembler to reserve space (ds/rb, rw) the assembler internally creates a gap for it. No more than 256 such statements can occur per file. You should probably break this file into smaller modules.

Type mismatch <NAME>: <FILE> <FILE>
The symbol indicated was declared as a global with different types in the files indicated.

Type mismatch: <NAME>.
A symbol already defined as an address of one type is being declared as an address of another type.

Undefined expression
The expression indicated must be defined either as an absolute or relative address, or number in this statement. This occurs in any statement where a new label is being defined.

Undefined label <DIGIT>
A numeric label was used in a forward reference (e.g. 1f, 2f, 3f, ...), but was not defined anywhere in the current segment. Numeric labels can not be referenced across segments.

Undefined local symbol <DIGIT>b

A backward reference was made to a numeric label (e.g. 1b, 2b, 3b, ...) that was not previously defined in the current segment. References to numeric labels cannot be made across segments.

Undefined symbol: <NAME>

If a line number and file are indicated, then either the symbol is being used on the right hand side of a label definition (in which case, its value must be an already known absolute address, or number), you may be using a mnemonic not known to the assembler, may have misspelled one that is known, or you may have tried to write a label-defining statement but made an error somewhere.

If no line and file number are indicated, the symbol indicated was used in the file without ever being defined. It needs to be declared as an external (if it is defined in another file) or needs to be defined before it is used.

Unexpected EOF inside // comment.

A // comment occurred at the end of the current file, and no end of line marker was seen. This indicates your source file is not in text mode.

Unexpected EOF inside ;; comment.

A ;; comment occurred at the end of the current file, and no end of line marker was seen. This indicates your source file is not in text mode.

Unexpected EOF inside comment.

A /* comment was seen, but was never closed with a */.

Unexpected EOF inside string.

A double quote was seen without a matching pair anywhere.

Unresolved external: <NAME>.

A symbol was declared as an external but wasn't defined anywhere. If it's a register or bit address defined for one of the 8051 extensions (8052, or 8051fa), you will have to use one of the include files provided (8052.h or 8051fa.h), or make your own.

Variables cannot be made global.

The keyword 'global' cannot be used with the statements Var = Val, or Var set Val. Variables can only be defined as local symbols.

Word value out of range.

An immediate word value (range -8000 to ffff hex, or -32768 to 65535 decimal) is required here.

8051 RELOCATER, notes on use.

(0) COMMAND LINE, INPUT FORMAT

This is a simple utility to relocate hex files. A typical use would be the following: you assemble a stand-alone program for address 0. However, when loading the program in preparation to blast it into EPROM, you have to upload it to memory address 4000 and then blast it to address 0.

The typical command line for this utility is:

reloc Offset File

For example:

reloc 4000 debug

will take the file debug.hex and generate the output in debug.hx. The input file must be suffixed in .hex, and the output will always have the .hx suffix.

The offset must be a 4-digit hexadecimal number. The source file's addresses are shifted by the amount indicated by this offset. Address calculations are done modulo 10000 hex, so an offset of c000 will effectively shift addresses at or above 4000 back by 4000.

SEMI.C, notes on use.

(0) COMMAND LINE, INPUT FORMAT

This utility will convert standard 8051 comments into a form that can be accepted by this assembler by substituting every semicolon by a double semicolon.

The typical command line for this utility is as follows:

semi <Input >Output

Listing of the sequencer program.

Use this file for preliminary reference. If you need to program a 87C750, check the latest version through ftp at rossini.obs-nice.fr
The main modes are supported in this version. More will be added later.

```
;;  
;; OCA CCD sequencer program - V1.0  
;; Alain Maury - Herve Viot  
;; December 1994 - March 1995  
;;
```

```
;; Definition of the I/O port bits  
;;
```

```
A1 equ P3.0  
A2 equ P3.1  
A3 equ P3.2  
CASC equ P3.7
```

```
H1 equ P1.0  
H2 equ P1.1  
H3 equ P1.2  
OSG equ P1.3  
RG equ P1.4  
CL1 equ P1.5  
CL2 equ P1.6  
STCVT equ P1.7
```

```
com_read equ P0.1  
com_dat equ P0.2
```

```
;; Variables stored in RAM
```

```
;;  
;; These are preloaded inside the functions  
;; or by the serial link at the start of an exposure
```

```
seg data at 0
```

```
RAM equ 20
```

```
C_LN_1 equ RAM           ;; Current Line Number  
C_LN_2 equ RAM+1  
C_CN_1 equ RAM+2         ;; Current Column Number  
C_CN_2 equ RAM+3  
C_NE_1 equ RAM+4         ;; Current Number Erase  
C_NE_2 equ RAM+5  
C_ID_1 equ RAM+6         ;; Delay for vertical inversion  
C_ID_2 equ RAM+7  
C_T_1 equ RAM+8          ;; Timer used for scan application  
C_T_2 equ RAM+9  
C_T_3 equ RAM+10  
C_T_4 equ RAM+11
```

```
;; Variables used for the windowed mode
```

```
C_VS_1 equ RAM+12        ;; Current Vertical Start  
C_VS_2 equ RAM+13  
C_HS_1 equ RAM+14        ;; Current Horizontal Start  
C_HS_2 equ RAM+15
```

```

C_VL_1 equ RAM+16      ;; Current Vertical Length
C_VL_2 equ RAM+17
C_HL_1 equ RAM+18      ;; Current Horizontal Length
C_HL_2 equ RAM+19

```

seg code at 0

INIT:

```

;; Interrupts disabled
MOV IE,#0x00
;; Initialisation of the I/O ports at their reset value
MOV P1,#0x81      ;; H1 and STCVT high
                  ;; camera rest mode
MOV P3,#0x00      ;; vertical lines at zero
;; waiting for an order to come :
START:
ACALL comm
;; comm is the general communication routine between an
;; 87C750 and its master through a synchronous serial link.
;; Acc is now loaded with a code which should be the mode in
;; which the camera is going to be used

```

```

MOV DPTR,#SELECTION
JMP @A+DPTR

```

SELECTION:

```

;;
AJMP START      ;; null transmission, return to start
AJMP STARE      ;; Stare exposure
AJMP BIN        ;; stare with binning
AJMP TEMP       ;; Transmit a single temp. measurement
AJMP TRAP       ;; Test for traps in CCD

```

;; The following are not finished and tested or not yet implemented

```

AJMP SCAN      ;; scan
AJMP S_SCAN    ;; smooth scan
AJMP WIN       ;; windowed mode
AJMP BAFAG     ;; back and forth autoguiding
AJMP GEO       ;; geostationary satellite detection

```

;; CCD clocking primary sequences
;;=====

;; Vertical transfert
;; Set at 20.13 microseconds, maybe too slow

;; During readout, the charges are transfered in
;; the 1->2->3->1 direction

```

VT: SETB A1      ;; A1 high
    ACALL DELAY_V
    SETB A2      ;; A1 and A2 high

```

```

ACALL DELAY_V
CLR A1          ;; A2 high
ACALL DELAY_V
SETB A3         ;; A2 and A3 high
ACALL DELAY_V
CLR A2          ;; A3 high
ACALL DELAY_V
SETB A1         ;; A1 and A3 high
ACALL DELAY_V
CLR A3          ;; A1 high
ACALL DELAY_V
CLR A1          ;; Everybody low
ACALL DELAY_V
RET             ;; return from subroutine

```

```

;; Vertical transfert in the reverse direction
;; -----
;; During reverse readout, the charges are transfered in the
;; 2->1->3->2 direction

```

```

RVT: SETB A2          ;; A2 high
      ACALL DELAY_V
      SETB A1          ;; A2 and A1 high
      ACALL DELAY_V
      CLR A2           ;; A1 high
      ACALL DELAY_V
      SETB A3          ;; A1 and A3 high
      ACALL DELAY_V
      CLR A1           ;; A3 high
      ACALL DELAY_V
      SETB A2          ;; A2 and A3 high
      ACALL DELAY_V
      CLR A3           ;; A2 high
      ACALL DELAY_V
      CLR A2           ;; Everybody low
      ACALL DELAY_V
      RET              ;; return from subroutine

```

```

;; Vertical transfert delay
;; -----
DELAY_V:          ;; Delay during vertical transfert
NOP               ;; 8 NOPs = 2.5 microseconds
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
RET              ;; return from subroutine

```

```

;; Fast horizontal transfert without digitisation
;; -----
;; Used during the beginning of the windowed mode
FHT:

```

```

SETB H2      ;; H1 and H2 high
CLR H1       ;; H2 high
SETB H3      ;; H2 and H3 high
CLR H2       ;; H3 high
SETB H1      ;; H1 and H3 high
CLR H3       ;; H1 high
RET          ;; return from subroutine

```

```

;; Horizontal transfert with digitisation
;;
;; Vertical transfert has been done with H1 high, and
;; transfert is done in the H1>H2>H3>H1>OSG direction
;; the length of this procedure is 14 cycles, or
;; with a master clock of 32 MHz, 5.25 microseconds
HT:

```

```

MOV P1, #10111101B
MOV P1, #10001100B
MOV P1, #10001110B
MOV P1, #10001010B
MOV P1, #10001011B
MOV P1, #10001001B
MOV P1, #10001001B
MOV P1, #10001001B
MOV P1, #11001001B
MOV P1, #11000001B
MOV P1, #11000001B
MOV P1, #11000001B
MOV P1, #11000001B
MOV P1, #00000001B
MOV P1, #10100001B
RET

```

```

;; Horizontal transfert during binning mode
;;
;; The direction of the transfert is of course the same
;; as in normal horizontal transfert, 6 extra cycles are
;; added at the beginning of the procedure, total time = 20
;; cycles = 7.5 microseconds
BHT:

```

```

MOV P1, #10111101B
MOV P1, #10001100B
MOV P1, #10001110B
MOV P1, #10001010B
MOV P1, #10001011B
MOV P1, #10001001B
MOV P1, #10111101B
MOV P1, #10001100B
MOV P1, #10001110B
MOV P1, #10001010B
MOV P1, #10001011B
MOV P1, #10001001B
MOV P1, #10001001B
MOV P1, #10001001B
MOV P1, #11001001B

```

```

MOV P1, #11000001B
MOV P1, #11000001B
MOV P1, #11000001B
MOV P1, #00000001B
MOV P1, #10100001B
RET          ;; return from subroutine

```

```

;; Vertical Line Inversion

```

```

;; -----
;; Used to produce the anti blooming mode

```

```

VLI:
    ORL C,A1          ;; Set carry bit if P1.0 == 1
    MOV P1,#0x03      ;; A1 and A2 set high
    ACALL DELAY_V      ;; Wait a bit
    MOV A2,C          ;; A2 is set to A1 former's value
    CPL C             ;; Carry bit complement
    MOV A1,C          ;; A1 is inverted
    CLR C             ;; Carry is cleared
    RET              ;; return from subroutine

```

```

;; Chip full readout

```

```

;; -----
;; It is not possible to interrupt this routine once
;; it has started readout:

```

```

;; Number of line = 2048 = 0x0800
;; Number of pixels per line = 2064 = 0x0810
    MOV C_LN_1,#0x08      ;; Number of lines
    MOV C_LN_2,#0x00

```

```

Line transfert:

```

```

    ACALL VT          ;; Vertical transfert
    MOV C_CN_1,#0x08   ;; Number of pixels reset
    MOV C_CN_2,#0x10

```

```

pixel transfert:

```

```

    ACALL HT          ;; Pixel digitisation
    DJNZ C_CN_1, pixel_transfert ;; Decrement pixel count
;; redo if !=0
    DJNZ C_LN_1, line_transfert  ;; Redo another line
    RET              ;; return from subroutine

```

```

;; Stare exposure procedure

```

```

;; -----
exposure:

```

```

;; Line inversion frequency =300 Hz, C_ID = 2300 = 0x08FC
    ACALL VLI          ;; Vertical line inversion
    MOV C_ID_1,#0x08
    MOV C_ID_2,#0xFC

```

```

delay_stare:

```

```

    JB com_read, end_expo ;; if com_read==1, end_expo
    DJNZ C_ID_2,delay_stare
    MOV C_ID_2,#0x00
    DJNZ C_ID_1,delay_stare
    AJMP exposure      ;; continue

```

```

end_expo:

```

RET

```
;; Scan exposure procedure
;;
scan_readout:
;; It starts with a vertical transfert, then a sequence of 8
;; readout of 2+256 pixels plus a vertical line inversion ),
;; allows to read the line while doing a periodic line
;; inversion. Once a line has been read, the camera places
;; itself in wait mode, while continuing periodic line
;; inversions.
;; The value sent by the computer in order to synchronise
;; the scan with the motion of the sky across the CCD is
;; a byte representing the total number of line inversion
;; plus 2 bytes giving the remaining number of CPU cycles
;; required to achieve a correct synchronisation.
;; C_T_1          ;; Timers used for scan application
;; C_T_2
;; C_T_3
;; C_T_4
;; In order to use another anti blooming pumping frequency,
;; it is necessary to use split the line in other packets
;; of pixels. In order to make things easily, it is possible
;; to read a line as a single 2064 pixels, or 2x1032, or
;; 3x688, or 4x516, or any other combination leading to a
;; total of 2064 pixels.

begin_scan:
    ACALL VT          ;; vertical transfert
    MOV R0, #0x08      ;; 8 burst of 258 pixels
                      ;; to read 2064 pixels

read_scan_line:
    ACALL HT          ;; first 2 pixels ( 258 = 256 + 2 )
    NOP              ;; Set to compensate for the
    NOP              ;; timing in the following loop
    ACALL HT
    NOP
    MOV B, #0x00      ;; loop to get the other 256 pixels

scan_pix:
    ACALL HT
    DJNZ B, scan_pix
    ACALL VLI          ;; vertical line inversion
    DJNZ R0, read_scan_line ;; Line has been read ?
;; The line has been read, now we wait the required time
;; till the next vertical transfert
;; Line inversion frequency =300 Hz, C_ID = 2300 = 0x08FC
    MOV C_ID_1, #0x08
    MOV C_ID_2, #0xFC

delay_scan:
    JB com_read, end_scan ;; if com_read==1, end_expo
    DJNZ C_ID_2, delay_scan
    MOV C_ID_2, #0x00
    DJNZ C_ID_1, delay_scan
    ACALL VLI          ;; Vertical line inversion
    AJMP begin_scan    ;; continue

end_scan:
    RET                ;; Exit from scan mode
```

```

;; back and forth shifting of charges
;; used to test traps in the CCD
5 shifts one way and the other are made 255 times.

```

```

baf:
    MOV B, #0xFF    ;;preload A with 255
start_baf:
    ACALL VT
    ACALL VT
    ACALL VT
    ACALL VT
    ACALL VT
    ACALL RVT
    ACALL RVT
    ACALL RVT
    ACALL RVT
    ACALL RVT
    DJNZ B, start_baf
    RET

```

```

;; Erase sequence, C_NE ( 3x2048) vertical transferts
;;
erase:
;; C_NE = 2*2048 = 6144 = 0x1800
    MOV C_NE_1, #0x18    ;; C_CE is loaded with CE
reset_NE_2:
    MOV C_NE_2, #0x00
begin_erase:
    ACALL VT
    DJNZ C_NE_2, begin_erase ;; C_NE_2 vertical transferts
    DJNZ C_NE_1, reset_NE_2  ;; C_NE_1 x C_NE_2 vert.
                           ;; transferts
    RET

```

```

;; 2x2 Binning readout sequence
;;
;; This sequence cannot be interrupted when started
bin_readout:
;;Number of lines in binning mode = 1024 = 0x0400
;;Number of pixels per lines in binning mode = 1032 = 0x0408
    MOV C_LN_1, #0x04    ;; Store in C_LN_1
l_t_b_2:
    MOV C_LN_2, #0x00    ;; Store in C_LN_2
line_transfert_b:
    ACALL VT              ;; two vertical transferts
    ACALL VT
    MOV C_CN_1, #0x04    ;; Store in C_CN_1
p_t_b_2:
    MOV C_CN_2, #0x08    ;; Store in C_CN_2
pixel_transfert_b:
    ACALL BHT             ;; binning mode readout
    DJNZ C_CN_2, pixel_transfert_b
                           ;; first loop for C_CN_2

```

```

        DJNZ C_CN_1, p_t_b_2      ;; second loop for horizontal
readout
        DJNZ C_LN_2, line_transfert_b
                                ;; first loop for C_LN_2
        DJNZ C_CN_1, l_t_b_2
                                ;; second loop for line transfert
        RET                      ;; return from subroutine

```

```

;; exposure procedures

```

```

;; -----
;; CCD clocking secondary procedures

```

```

STARE:                      ;; Stare exposure starts right away
        ACALL erase
        ACALL exposure
        ACALL readout
        AJMP START

```

```

SCAN:                      ;; scan
;; The four following variables correspond to the time
;; interval required between the end of the horizontal
;; readouts and the start of the next line transfert
;; It is calculated by the PC

```

```

        ACALL Comm          ;; A is loaded with C_T_1
        MOV C_T_1, A
        ACALL Comm          ;; A is loaded with C_T_2
        MOV C_T_2, A
        ACALL Comm          ;; A is loaded with C_T_3
        MOV C_T_3, A
        ACALL Comm          ;; A is loaded with C_T_4
        MOV C_T_4, A
        ACALL erase
        ACALL scan_readout
        ACALL readout       ;; Empties the CCD
        AJMP START

```

```

S_SCAN:                    ;; smooth scan
;; Not implemented yet
;; The idea is to transfert charges by a third of a pixel to
;; see if it improves the image shape, as well as astrometry

```

```

        AJMP START

```

```

BIN:                      ;; stare with binning

```

```

        ACALL erase
        ACALL exposure
        ACALL bin_readout
        AJMP START

```

```

WIN:                      ;; windowed mode
;; Not finished yet. Should come soon

```

```

:: Initialisation of the windowed mode
  ACALL Comm      ;; A is loaded with C_VS_1
  MOV C_VS_1 , A  ;; Current Vertical Start
  ACALL Comm      ;; A is loaded with C_VS_2
  MOV C_VS_2 , A

  ACALL Comm      ;; A is loaded with C_HS_1
  MOV C_HS_1 , A  ;; Current Horizontal Start
  ACALL Comm      ;; A is loaded with C_HS_2
  MOV C_HS_2 , A

  ACALL Comm      ;; A is loaded with C_VL_1
  MOV C_VL_1 , A  ;; Current Vertical Length
  ACALL Comm      ;; A is loaded with C_VL_2
  MOV C_VL_2 , A

  ACALL Comm      ;; A is loaded with C_HL_1
  MOV C_HL_1 , A  ;; Current Horizontal Length
  ACALL Comm      ;; A is loaded with C_HL_2
  MOV C_HL_2 , A

```

AJMP START

```

BAFAG:      ;; back and forth autoguiding
;; Not implemented yet.
;; It will consist of a windowed exposure of a zone
;; located near the output amplifier, followed by a
;; certain number of reverse vertical shifts at
;; periodic interval to put the charges back to their
;; original location. This operation is performed at
;; periodic intervals with the shutter closed.
;;

```

AJMP START

```

GEO:      ;; geostationnary satellite detection
;; Not implemented yet.
;; This mode is a mixture of scan and short stare
;; exposures every 67 or so seconds. The idea is to drive
;; the telescope at sidereal rate, and drift the charges
;; toward the horizontal register at sidereal rate. This way
;; geostationary sources appear as points whereas
;; stars are drifted. When the stare mode is applied
;; stars integrate as point sources whereas satellites
;; drift. When scan mode is resumed, the satellite appear
;; as a second dot near the first one, shifted in RA.
;; The stare exposure allows to date the images ( the time
;; at which the image is done gives the midpoint of the pair
;; of satellites images, as well as its time ). Automated
;; detection should look for RA aligned pairs of stars of
;; similar brightnesses

```

AJMP START

TRAP:

```
;; Test for traps in CCD
;; Allows for a normal integration, then shifts charges back
;; and forth several times, finally readouts the CCD.
    ACALL erase
    ACALL exposure
    ACALL baf
    ACALL readout
    AJMP START
```

;; Temperature readout

```
;;
;; Function which sets the converter in cascade mode, and
;; starts a conversion which will read the temperature probe
;; Needs to be tested, mainly to add possible delays between
;; the instructions
```

Temp:

```
    SETB CASC      ;; Puts the converter in cascade mode
    SETB STCVT     ;; starts a conversion
    NOP            ;; wait a little
    CLR STCVT      ;; stops the conversion pulse
    CLR CASC       ;; Puts the converter in normal mode
    RET            ;; end
```

```
;; Communication procedure, it will be necessary
;; to test it and adapt its timing, and modify
;; if necessary depending of the clock frequency of
;; the master board 8051.
;; Time out procedure added HV Feb 95
```

Comm:

```
;; test if P0.1==1, stored in Carry bit
    MOV C, com_read
    JNC Comm
```

```
;; Receive counter initialisation to 8
    MOV A, #0x08      ;; A = 8
```

```
;; Data register initialisation to 0
    MOV B, #0x00      ;; B = 0
```

```
;; Receive error flag initialisation to 0
    MOV R0, #0x00
```

start_comm:

```
;; Time_out initialisation to its maximum value
    MOV R1, #0xff
```

wait_end_clock_one:

```
;; test if P0.1 == 0, stored in Carry bit
    MOV C, com_read
    JNC data_receive
    DJNZ R1, wait_end_clock_one

;; If time-out reached, R1 = 0, default flag = 1
    MOV R0, #0x01
    MOV A, #0x01
    SJMP wait_end_clock_one
```

data_receive:

```
;; if com_read == 0, read com_dat
;; get bits already received
    XCH A, B
    MOV C, com_dat

;; Rotate A right through carry
    RRC A

;; Save data already received
    XCH A, B

;; Indication of data reception
    DEC A

;; Exit reception after 8 received bits
    JZ data_valid_test

;; Initialisation of receive time_out to the maximum
    MOV R1, #0xff
```

wait_end_clock_zero:

```
;; test if P0.1 == 1, stored in Carry bit
    MOV C, com_read
    JC start_comm
    DJNZ R1, wait_end_clock_zero

;; if time-out , R1 = 0, flag default = 1
    MOV R0, #0x01
```

data_valid_test:

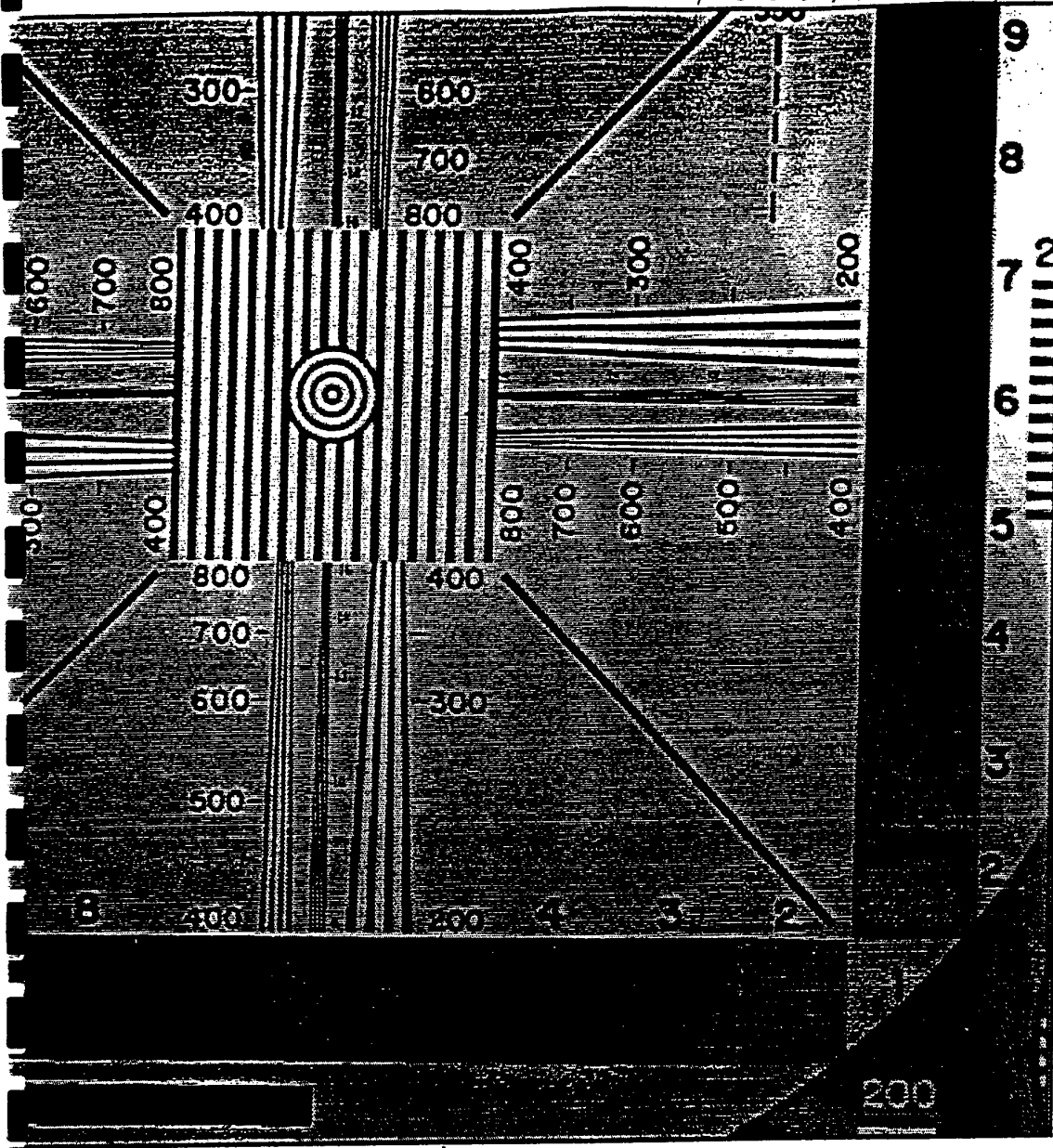
```
;; if alarm_flag = 1, data is not valid
    CJNE R0, #0x01 , end_data_reception

;; Wait a correct message reception
    SJMP comm

end_data_reception:
;; get correct command and return
    XCH A, B
    RET
```

Appendix 4 :
Cosmetic quality of
10 CCD442A CCDs

1 CCD442H-411112

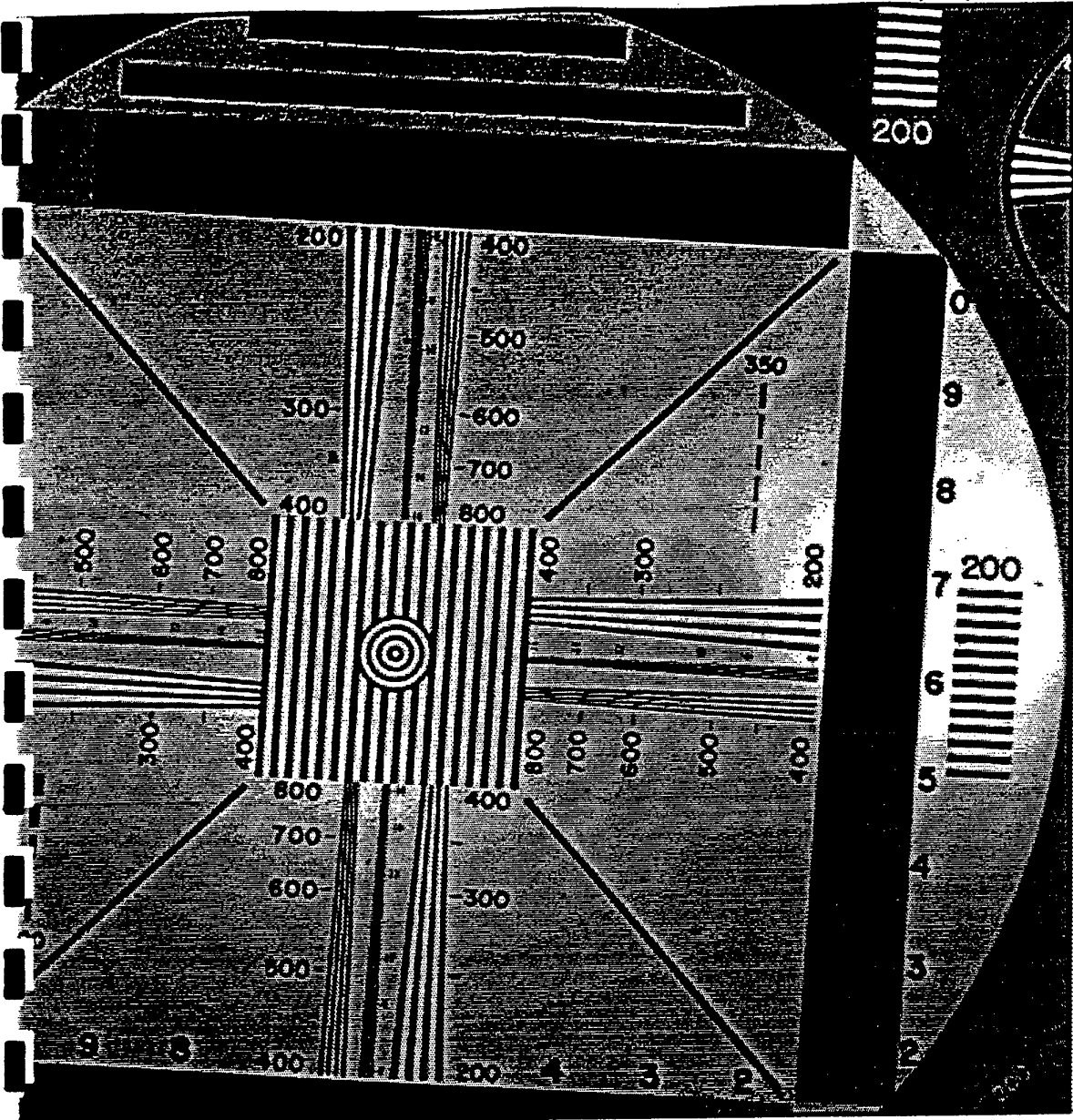


442A
0041
7-94
80°

↑
smear
918
(6)
cy=716

200

SL141702



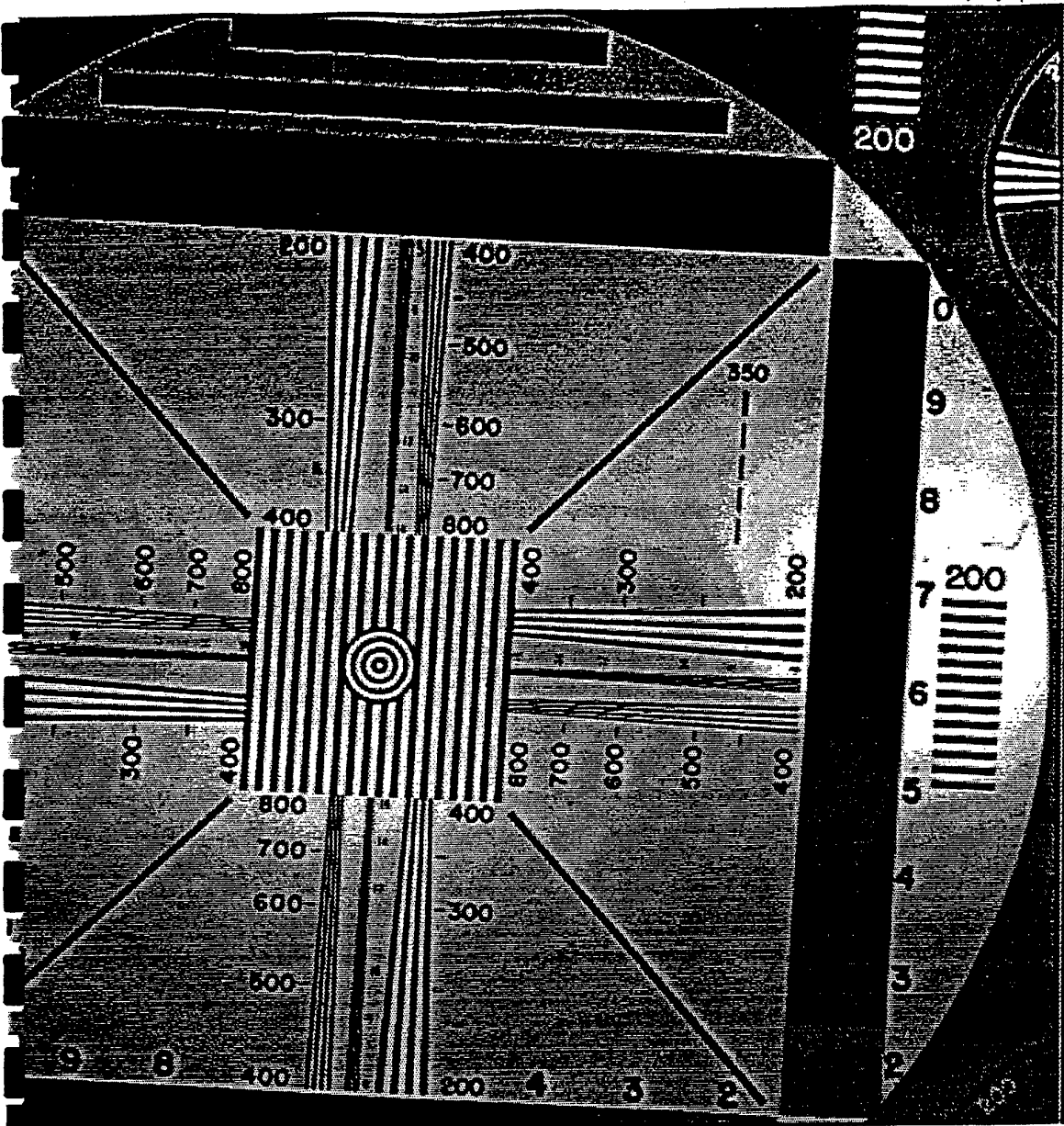
CCD442A

T000062

0°

5-19-94

SL141702



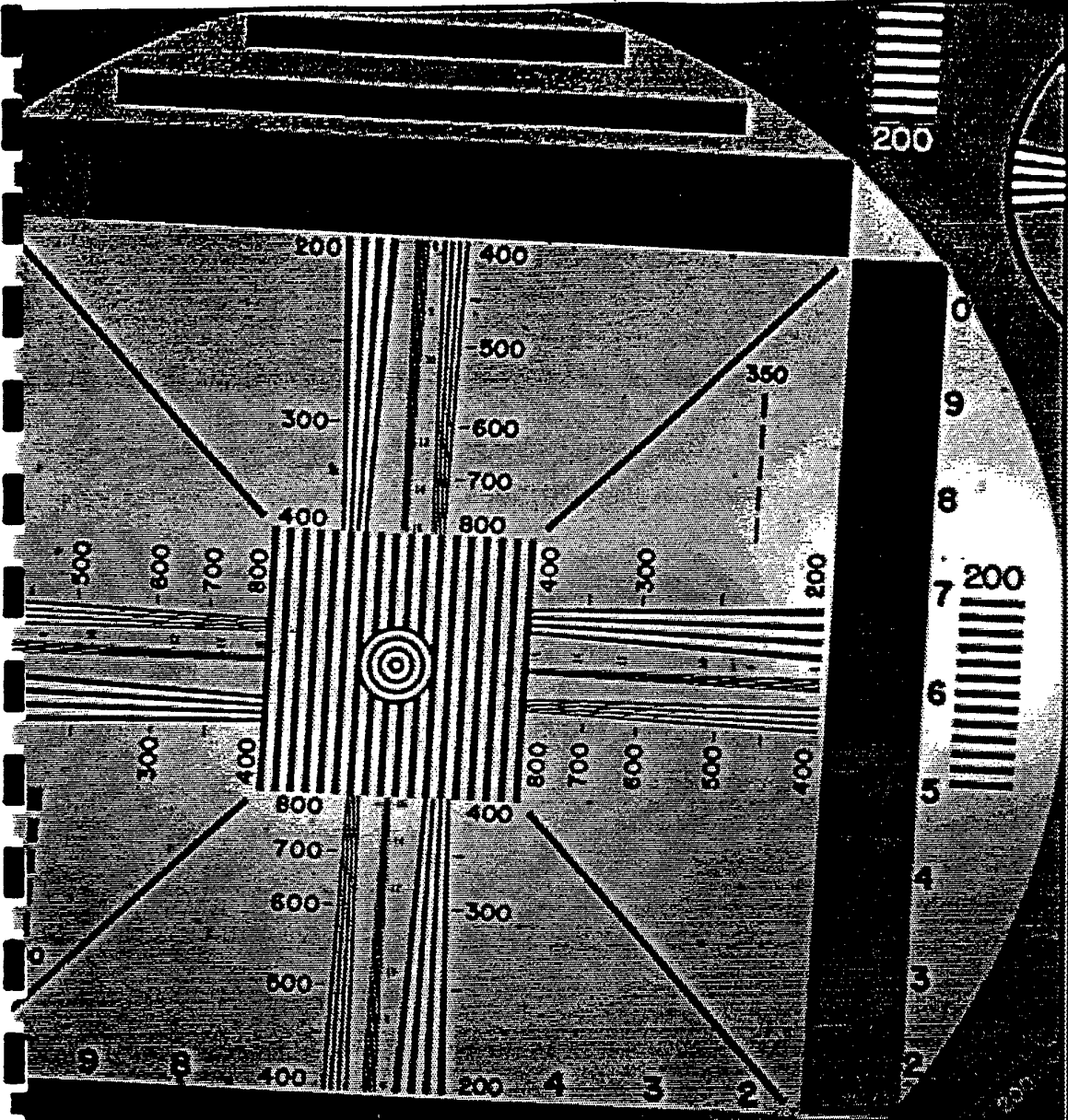
CCD442A

T00006Z

180°

5-19-94

SL141707



↑
1044
(2)

CCD442A

T000064

5-19-94

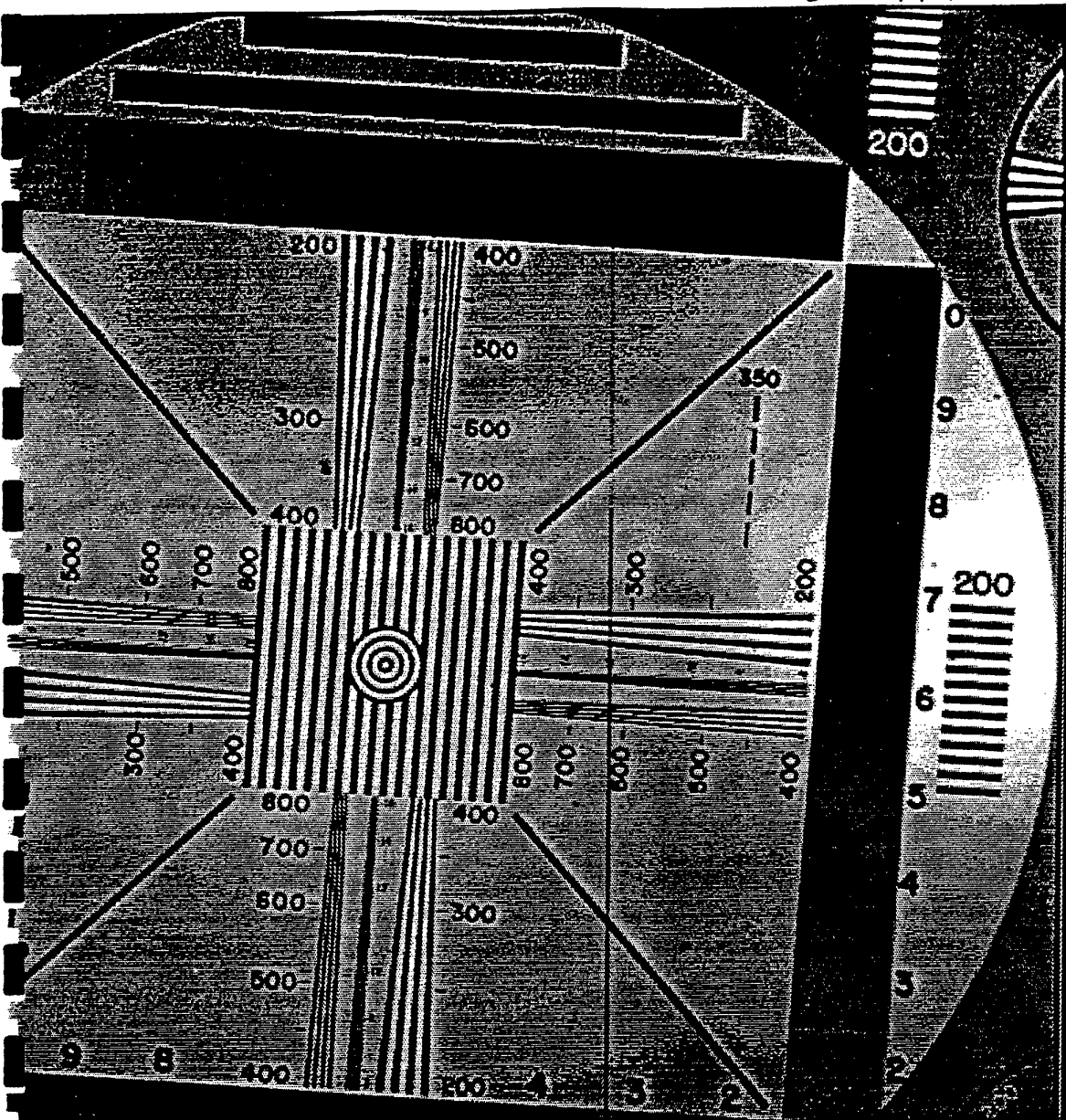
0°

The image displays a circular target or scale, possibly a rangefinder or a measurement device. The target is divided into four quadrants by a vertical and horizontal line. Each quadrant contains radial lines and numerical markings. The top and bottom quadrants have markings from 200 to 800. The left and right quadrants have markings from 400 to 800. A vertical scale on the right side of the target shows numbers 0 through 9. The image is heavily degraded with noise and artifacts.

1980
(4)

CCD4424
T000064
5-19-94
180°

SL141702



↑
1194
(4)

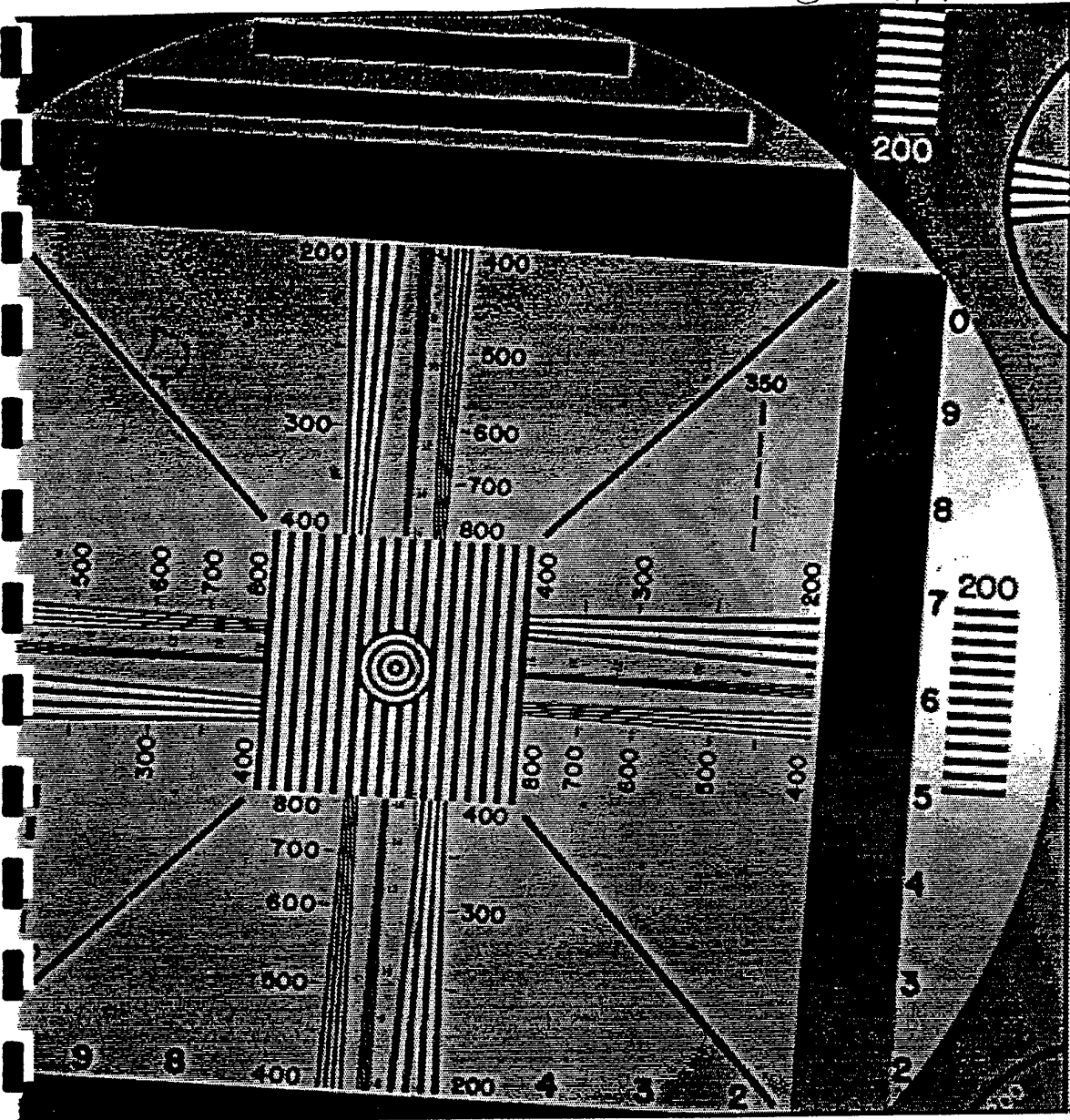
CD442A

000060

-10-94

180°

SL141702



↑
814
(4)

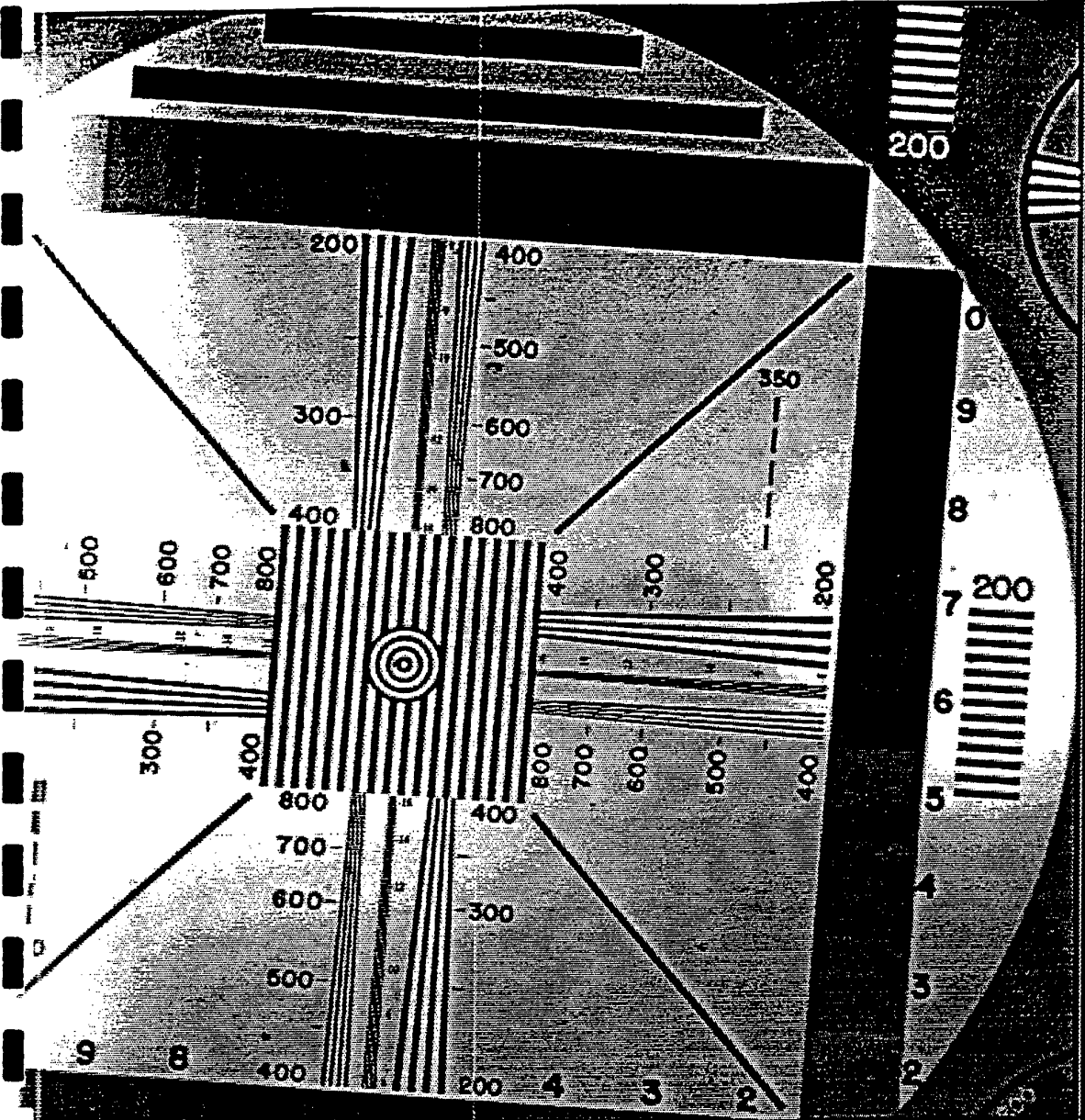
CD 442 A

0000 60

5-10-94

0°

SL141702



↑ 126
(6)
BRIGHT
SIDE

↑ 922
(2)

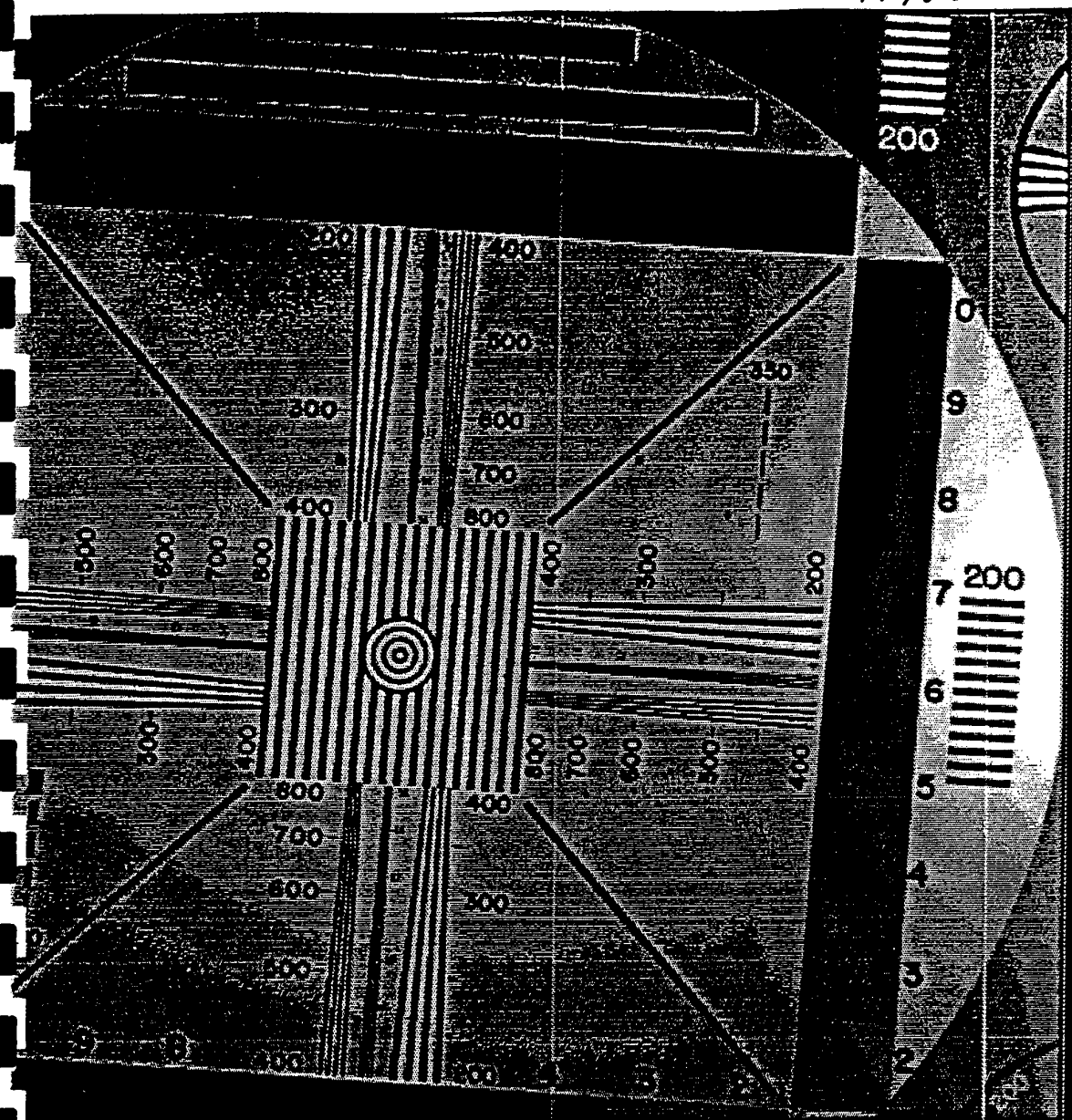
D442A

000059

-10-94

0°

SL141702



↑
1088
(2)

↑
1884
(6)
BRIGHT SIDE

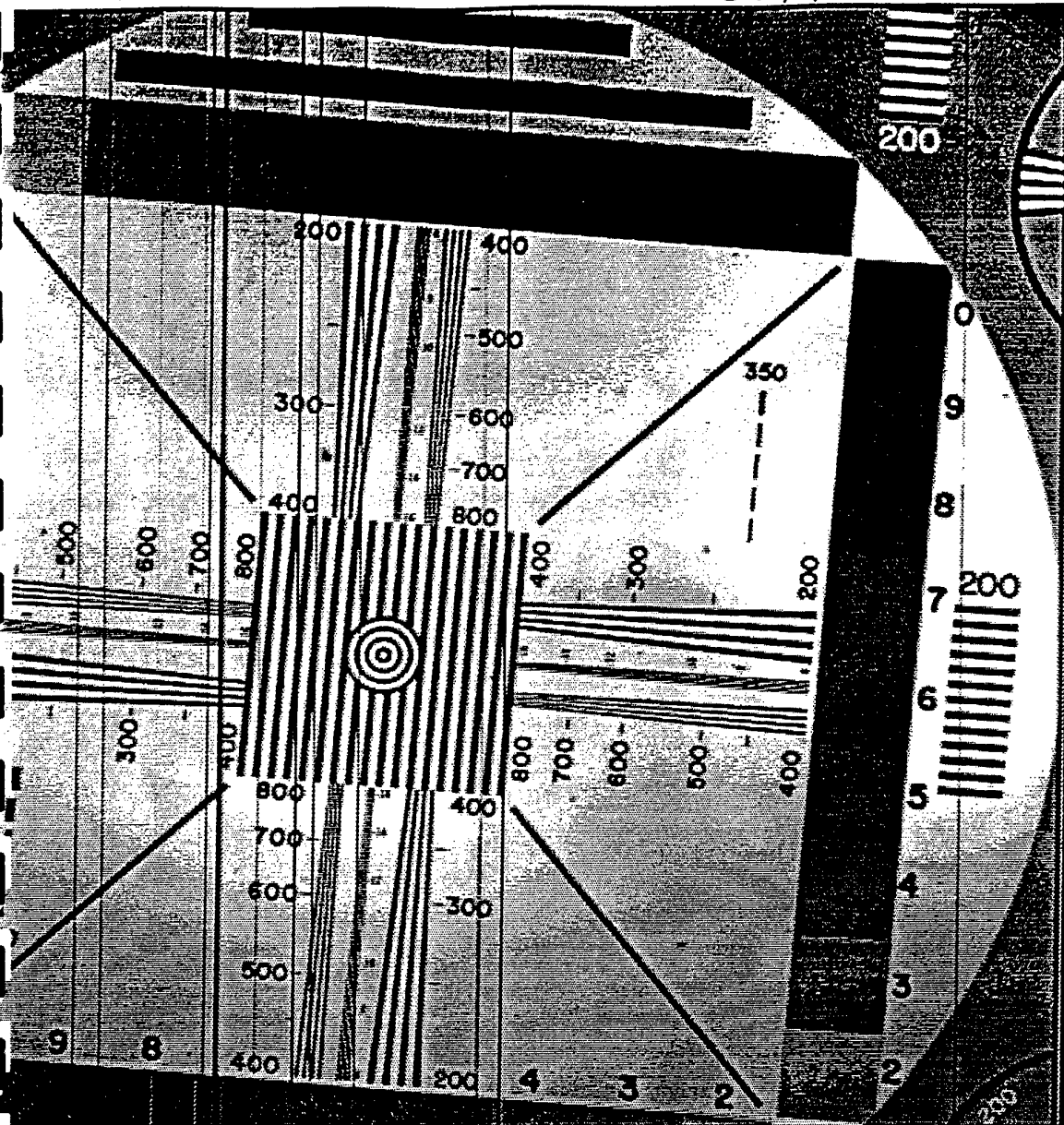
CD442A

T000059

5-10-94

180°

SL141702



similar to
0°

CD442A

T000058

5-10-94

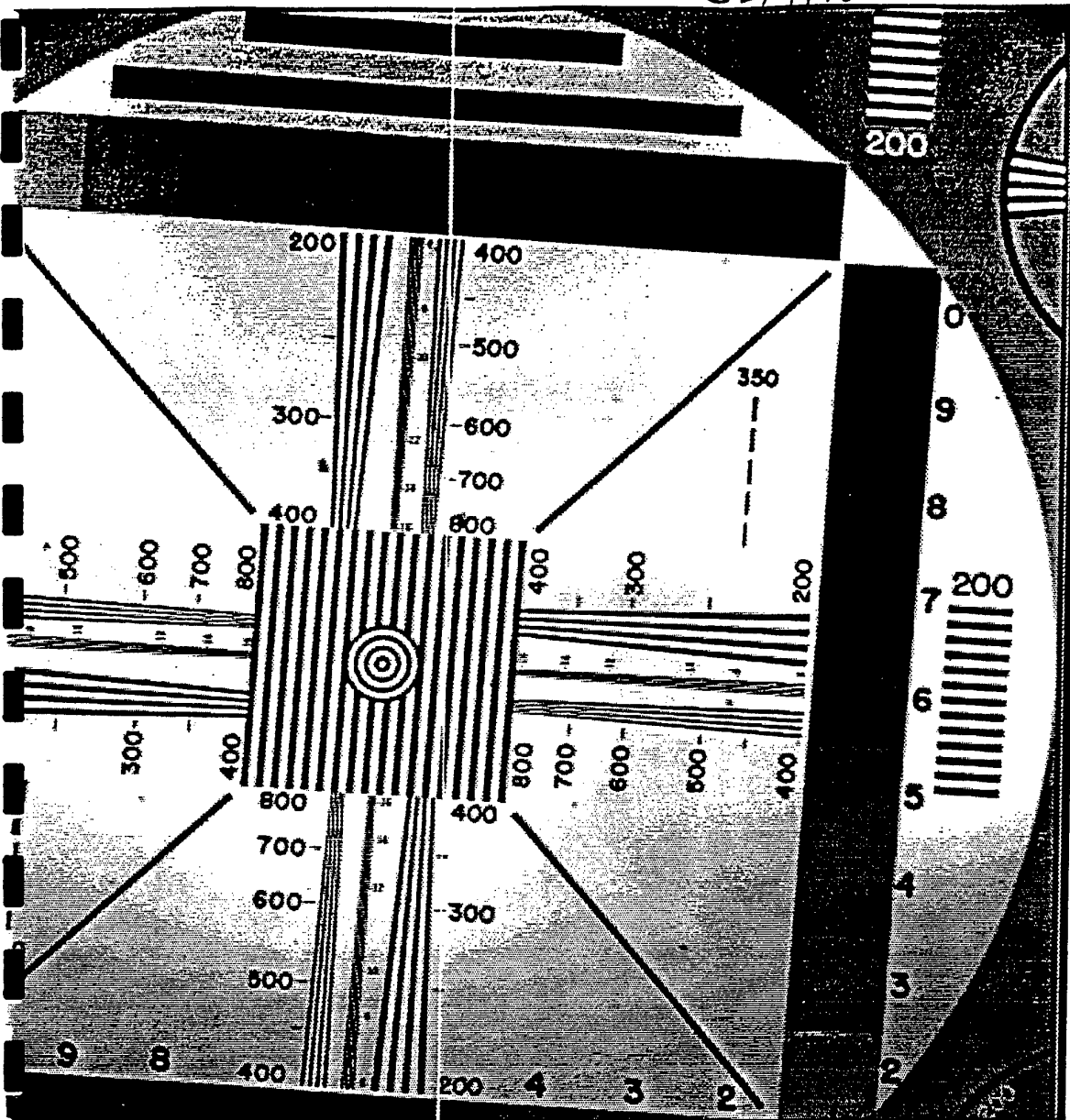
180°

0°

1116
(4)

 180°

SL141702



↑
892
(b)

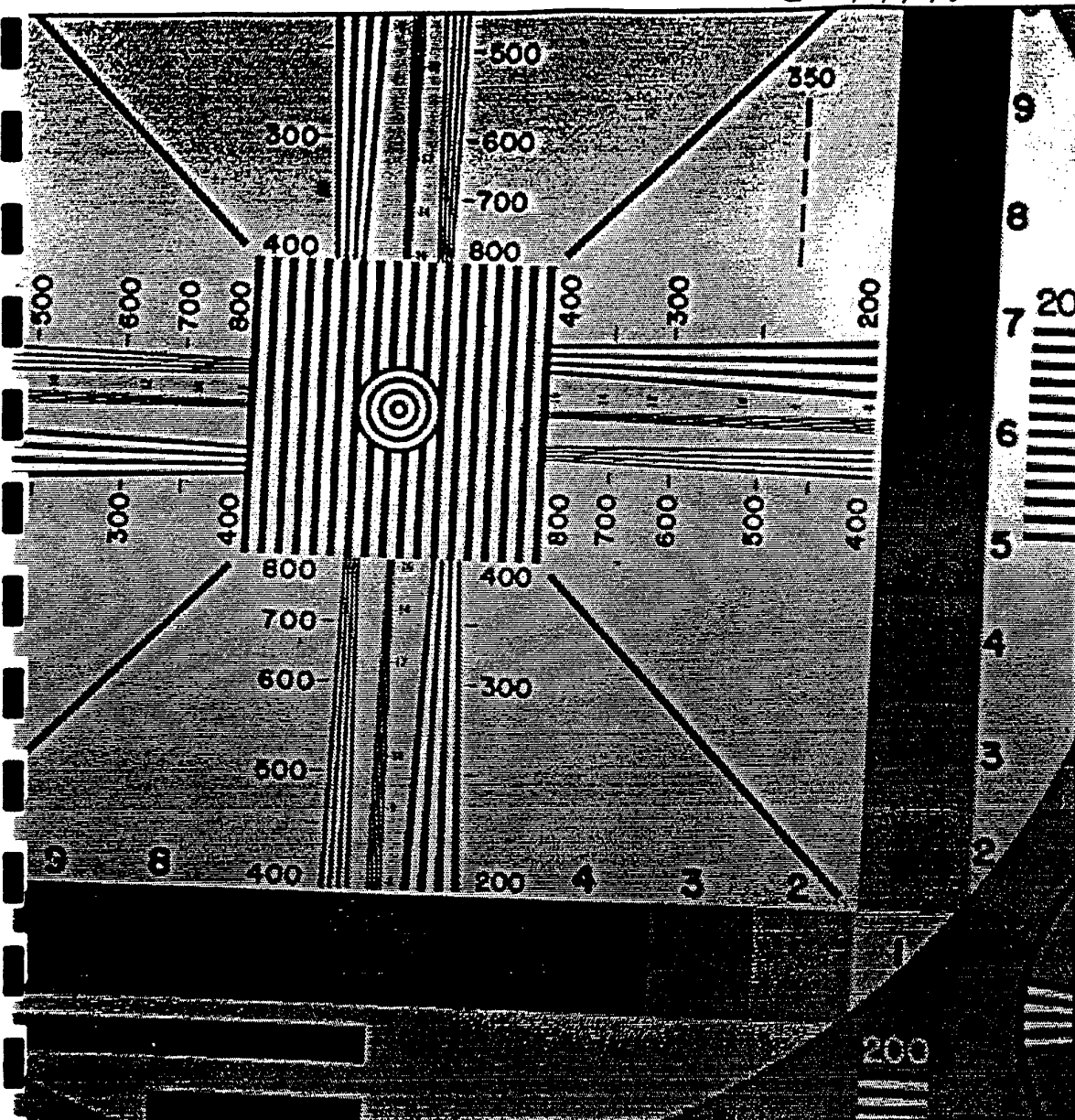
CD 442A

000057

5-10-94

0

SL141702



↑
1226
(2)
Smear

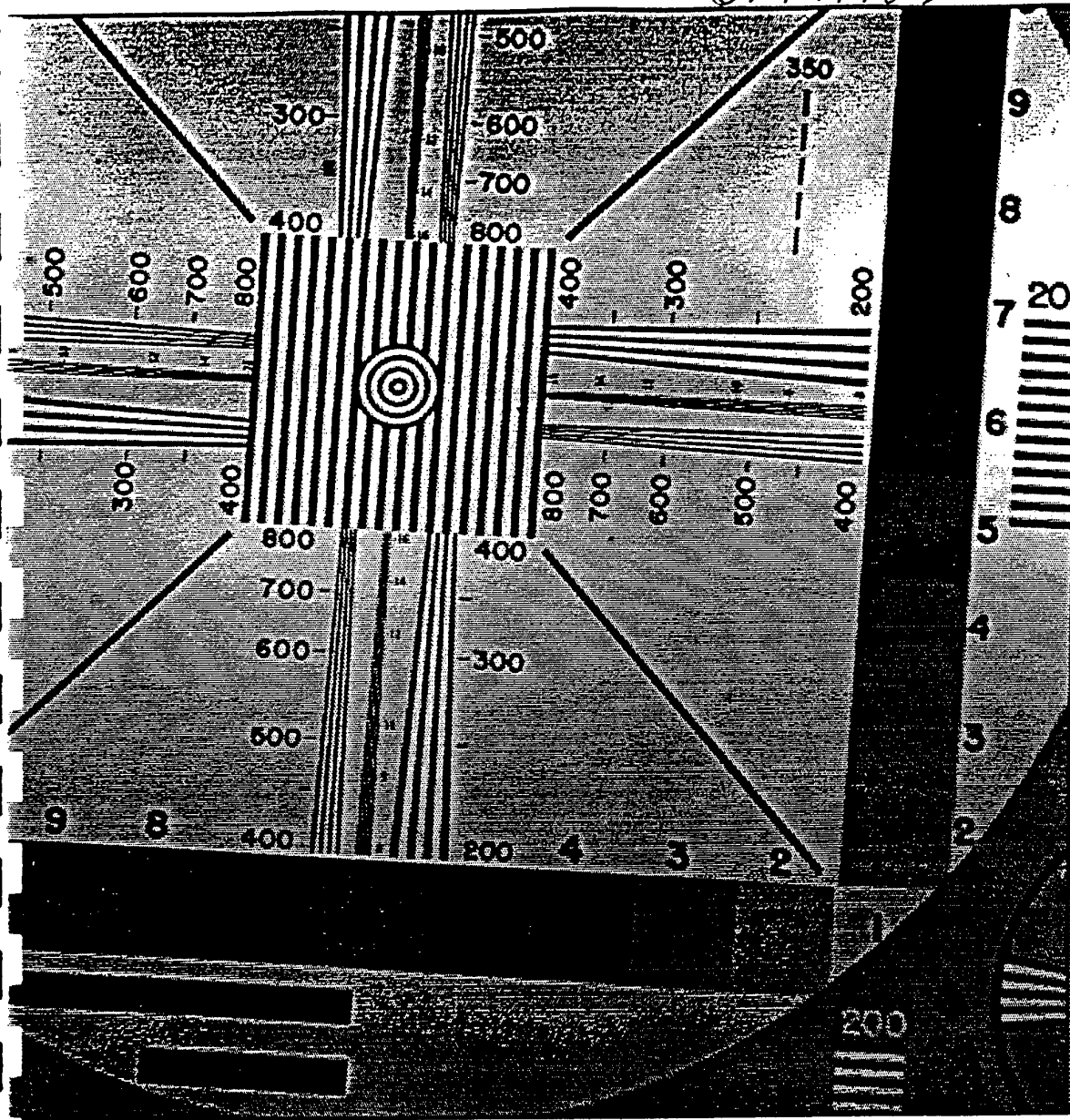
D442A

000049

5-9-94

0°

SL141702



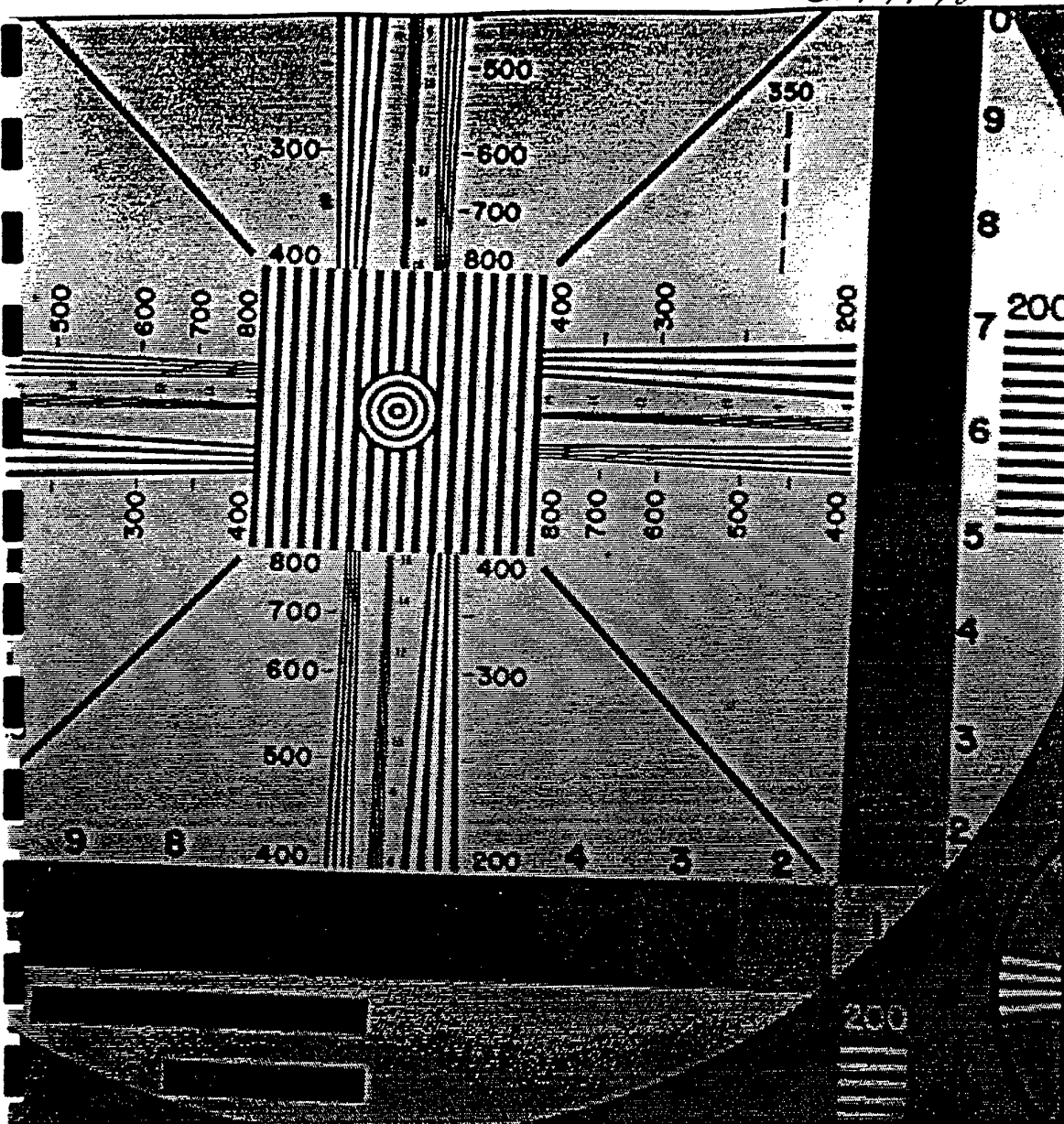
CD 442A

7000049

5-9-94

180°

SL141702



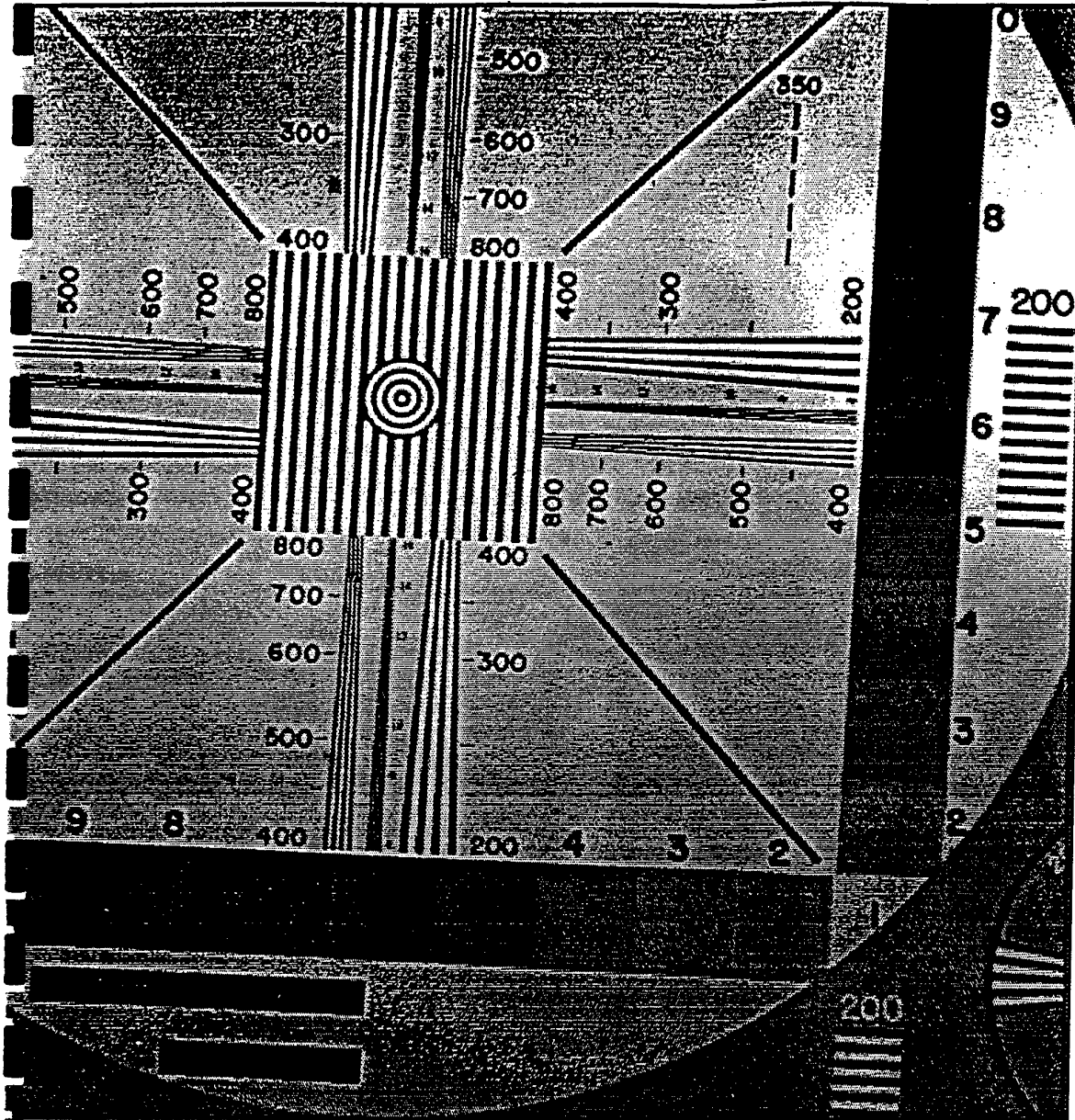
CCD442A

000051

-9-94

0°

SL141702



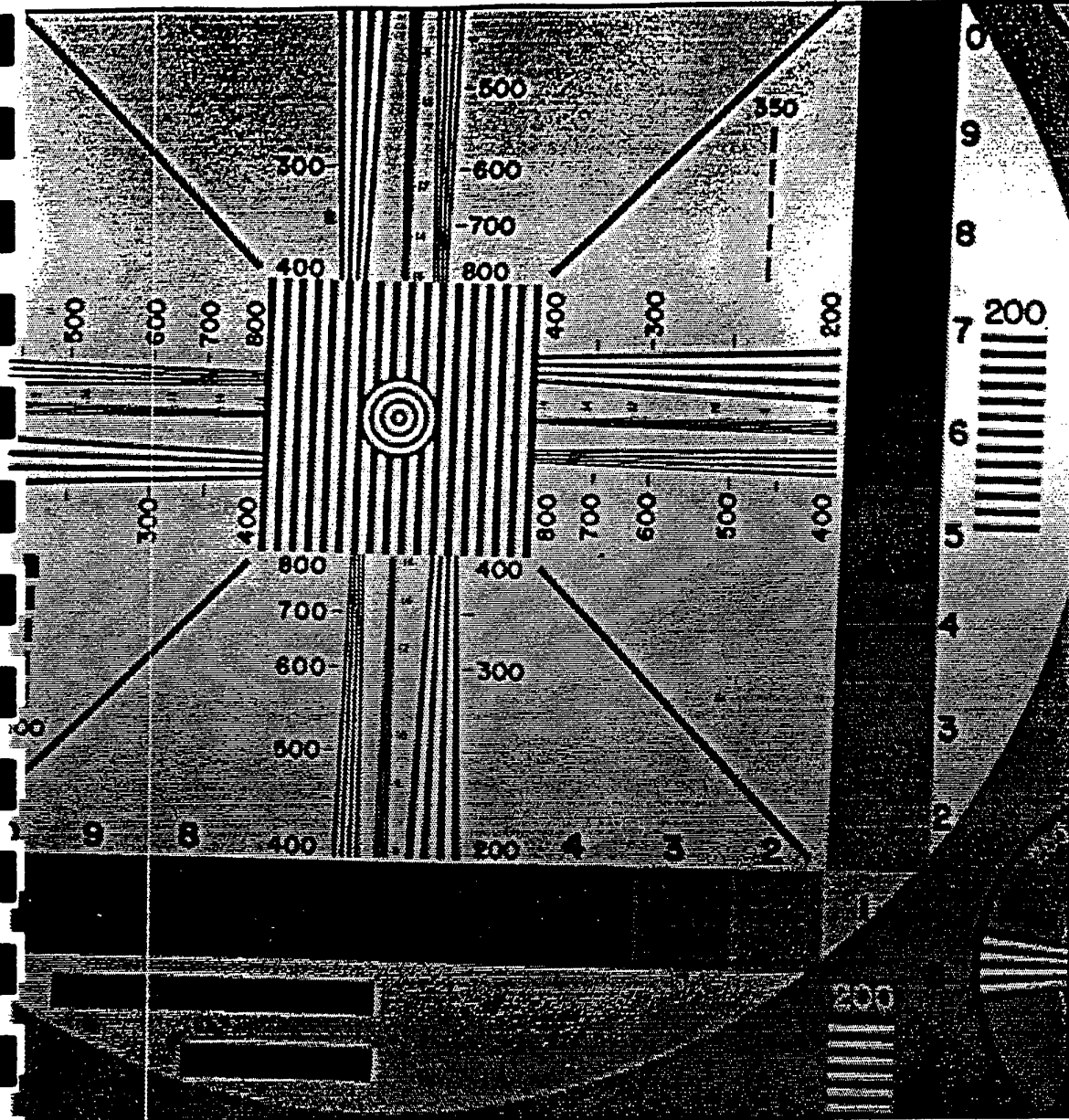
CD442A

000051

-9-94

180°

SL141702



↑
338
(2)

↑↑↑
1392
(2)
y = 1088
smear
1424
(4)
y = 1286
smear
1416
(2)
y = 1240
smear

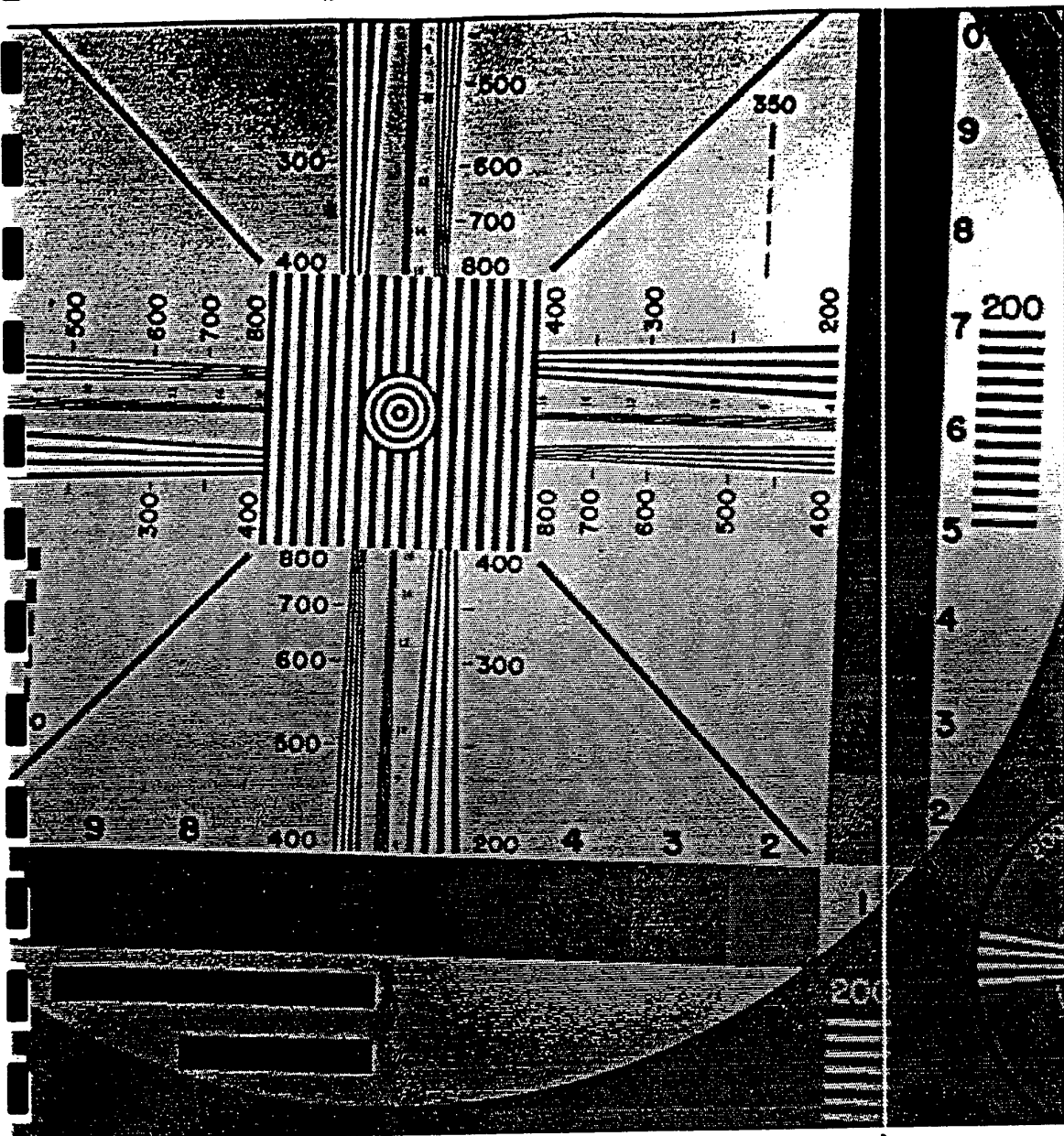
CD442A

T000052

5-9-94

0°

SL141702



612
(4)

622
(2)

646
(2)

1700
(4)

CD442A

000052

5-9-94

180°

Dark-flat field smudge
- see printout