

AD-A223 507 COPY

1

FINAL TECHNICAL REPORT  
VOLUME III

AD-A223 507

INTERACTIVE VISUAL SIMULATION OF  
COMMUNICATION SYSTEMS

DTIC  
ELECTE  
JUL 06 1990  
S D D

April 29, 1988

Contract No. DAAB07-87-C-A023

Prepared for

Commander

U.S. Army CECOM

Fort Monmouth, N. J. 07703-5000

DISTRIBUTION STATEMENT A  
Approved for public release  
Distribution Unlimited

LINKNET

710 Silver Spur Road, Suite 285  
Rolling Hills Estates, California 90274

(213) 373-3384

90 07 6 015

TABLE OF CONTENTS

INTERFACE SOURCE LISTINGS

1.	LN.ASM	.....	1
2.	LN1.ASM	.....	98

Accession For	
NTIS CRA&I	<input checked="" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By <i>perform 50</i>	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

; \*\*\*\*\* LN.ASM \*\*\*\*\*

INCLUDE L3.MAC

STACK SEGMENT 'STACK' STACK
DB 1000 DUP (?)
STACK ENDS

CONST SEGMENT 'DATA' PARA PUBLIC
PUBLIC ENVIRON
ENVIRON DW ?

LOGO DB OFFH,59H,0,0,6,77," "
DB OFFH,59H,0,8,6,77," "
DB OFFH,59H,0,16,6,77," "
DB OFFH,35,11,"LINKNET"
DB OFFH,31,13,"VISUAL SIMULATOR"
DB OFFH,59H,1,1,4,23," "
DB OFFH,59H,27,1,4,23," "
DB OFFH,59H,53,1,4,23," "
DB OFFH,59H,2,2,2,5," "
DB OFFH,59H,10,2,2,5," "
DB OFFH,59H,18,2,2,5," "
DB OFFH,59H,28,2,2,5," "
DB OFFH,59H,36,2,2,5," "
DB OFFH,59H,44,2,2,5," "
DB OFFH,59H,54,2,2,5," "
DB OFFH,59H,62,2,2,5," "
DB OFFH,59H,70,2,2,5," "
DB OFFH,59H,1,17,4,23," "
DB OFFH,59H,27,17,4,23," "
DB OFFH,59H,53,17,4,23," "
DB OFFH,59H,2,18,2,5," "
DB OFFH,59H,10,18,2,5," "
DB OFFH,59H,18,18,2,5," "
DB OFFH,59H,28,18,2,5," "
DB OFFH,59H,36,18,2,5," "
DB OFFH,59H,44,18,2,5," "
DB OFFH,59H,54,18,2,5," "
DB OFFH,59H,62,18,2,5," "
DB OFFH,59H,70,18,2,5," "
DB OFFH,79,25
DB 0DH

ROOT\_CONTROL DW OFFSET ROOT\_STACK\_MESS1
DW OFFSET ROOT\_STACK1
DW OFFSET ROOT\_S\_MESS1
DW OFFSET ROOT\_S1
DW OFFSET ROOT\_S\_MESS1
DW OFFSET ROOT\_S1
DW 000H
DW 34FH
DW 400H
DW 64FH
DB 7
DW OFFSET FUNCTION\_CODE\_STACK



DW 0  
DW OFFSET LINK\_S1  
DW OFFSET LINK\_S\_MESS1  
DW OFFSET LINK\_S\_HELP1  
DB 1,2,5,11  
DW 0  
DW OFFSET NETWORK\_S1  
DW OFFSET NETWORK\_S\_MESS1  
DW OFFSET NETWORK\_S\_HELP1  
DB 1,2,13,15  
DW 8100H  
DW OFFSET DOS\_HANDLER  
DW ?  
DW OFFSET DOS\_HELP1  
DB 1,2,17,20  
DW 4000H  
DW OFFSET EXIT\_S1  
DW OFFSET EXIT\_MESS1  
DW OFFSET EXIT\_HELP1

EXIT\_MESS1 DB OFFH,50H,0FH,OFFH,0,2

DB "NO YES"

DB 0DH

EXIT\_HELP1 DB 0DH,"\$"

EXIT\_S1 DB 2

DB 0FH,70H

DW OFFSET EXIT\_S1+5

DB 1,2,0,1

DW 8200H

DW OFFSET DUMMY\_CALL

DW ?

DW OFFSET NO\_HELP1

DB 1,2,3,5

DW 8200H

DW OFFSET EXIT\_HANDLER

DW ?

DW OFFSET YES\_HELP1

LINK\_S\_MESS1 DB OFFH,50H,0FH,OFFH,0,2

DB "CONFIGURATION SIMULATE POST-PROCESSING\_FILES QUIT"

DB 0DH

LINK\_S\_HELP1 DB 0DH,"\$"

LINK\_S1 DB 4

DB 0FH,70H

DW OFFSET LINK\_S1+5

DB 1,2,0,12

DW 2000H

DW OFFSET CONFIG\_S1

DW OFFSET CONFIG\_S\_MESS1

DW OFFSET CONFIG\_S\_HELP1

DB 1,2,14,21

DW 0

DW OFFSET SIMUL\_S1

DW OFFSET SIMUL\_S\_MESS1

DW OFFSET SIMUL\_S\_HELP1

DB 1,2,23,43  
DW 0  
DW OFFSET ANALY\_S1  
DW OFFSET ANALY\_MESS1  
DW OFFSET LINK\_S\_HELP1  
DB 1,2,45,48  
DW 8200H  
DW OFFSET QUIT\_HANDLER  
DW ?  
DW OFFSET QUIT\_HELP1

NETWORK\_S\_MESS1 DB OFFH,50H,0FH,OFFH,0,2  
DB "CONFIGURATION SIMULATE POST-PROCESSING\_FILES QUIT"  
DB 0DH

NETWORK\_S1 DB 4  
DB 0FH,70H  
DW OFFSET NETWORK\_S1+5  
DB 1,2,0,12  
DW 2000H  
DW OFFSET NETCON\_S1  
DW OFFSET NETCON\_S\_MESS1  
DW OFFSET NETCON\_S\_HELP1  
DB 1,2,14,21  
DW 0  
DW OFFSET NET\_SIMUL\_S1  
DW OFFSET NET\_SIMUL\_S\_MESS1  
DW OFFSET NET\_SIMUL\_S\_HELP1  
DB 1,2,23,43  
DW 1  
DW OFFSET ANALY\_S2  
DW OFFSET ANALY\_MESS1  
DW OFFSET LINK\_S\_HELP1  
DB 1,2,45,48  
DW 8200H  
DW OFFSET QUIT\_HANDLER  
DW ?  
DW OFFSET QUIT\_HELP1

NETWORK\_S\_HELP1 DB 0DH,"\$"

CONFIG\_S\_MESS1 DB OFFH,50H,0FH,OFFH,0,2  
DB "RETRIEVE SAVE EDIT DOS QUIT",0DH  
CONFIG\_S\_HELP1 DW OFFSET CON\_HELP  
DW OFFSET DISPLAY\_CON  
DW OFFSET RECOVER\_CON

CON\_HELP DB 0DH,"\$"

CONFIG\_S1 DB 5  
DB 0FH,70H  
DW OFFSET CONFIG\_S1+5  
DB 1,2,0,7  
DW 8100H  
DW OFFSET R\_CONFIG\_S1  
DW 0  
DW OFFSET R\_CONFIG\_S\_HELP1  
DB 1,2,9,12  
DW 8100H

DW OFFSET S\_CONFIG\_S1  
DW 0  
DW OFFSET S\_CONFIG\_S\_HELP1  
DB 1,2,14,17  
DW 0  
DW OFFSET E\_CONFIG\_S1  
DW OFFSET E\_CONFIG\_S\_MESS1  
DW OFFSET E\_CONFIG\_S\_HELP1  
DB 1,2,19,21  
DW 8100H  
DW OFFSET DOS\_HANDLER  
DW ?  
DW OFFSET DOS\_HELP1  
DB 1,2,23,26  
DW 8300H  
DW OFFSET QUIT\_HANDLER  
DW ?  
DW OFFSET QUIT\_HELP1

NETCON\_S\_MESS1 DB OFFH,50H,0FH,OFFH,0,2  
DB "RETRIEVE SAVE EDIT DOS QUIT",0DH  
NETCON\_S\_HELP1 DW OFFSET NETCON\_HELP  
DW OFFSET DISPLAY\_NETCON  
DW OFFSET RECOVER\_NETCON

NETCON\_HELP DB 0DH,"\$"

NETCON\_S1 DB 5  
DB 0FH,70H  
DW OFFSET NETCON\_S1+5  
DB 1,2,0,7  
DW 8100H  
DW OFFSET R\_NETCON\_S1  
DW 0  
DW OFFSET R\_NETCON\_S\_HELP1  
DB 1,2,9,12  
DW 8100H  
DW OFFSET S\_NETCON\_S1  
DW 0  
DW OFFSET S\_NETCON\_S\_HELP1  
DB 1,2,14,17  
DW 0  
DW OFFSET E\_NETCON\_S1  
DW OFFSET E\_NETCON\_S\_MESS1  
DW OFFSET E\_NETCON\_S\_HELP1  
DB 1,2,19,21  
DW 8100H  
DW OFFSET DOS\_HANDLER  
DW ?  
DW OFFSET DOS\_HELP1  
DB 1,2,23,26  
DW 8300H  
DW OFFSET QUIT\_HANDLER  
DW ?  
DW OFFSET QUIT\_HELP1

SIMUL\_S\_MESS1 DB OFFH,50H,0FH,OFFH,0,2

```

DB "RUN QUIT",0DH
SIMUL_S_HELP1 DB 0DH,"$"
SIMUL_S1 DB 2
DB 0FH,70H
DW OFFSET SIMUL_S1+5
DB 1,2,0,2
DW 8100H
DW OFFSET RUN_S1
DW 0
DW OFFSET RUN_HELP1
DB 1,2,4,7
DW 8300H
DW OFFSET QUIT_HANDLER
DW ?
DW OFFSET QUIT_HELP1

```

```

NET_SIMUL_S_MESS1 DB 0FFH,50H,0FH,0FFH,0,2
DB "RUN OUTPUT DISPLAY QUIT",0DH
NET_SIMUL_S_HELP1 DB 0DH,"$"
NET_SIMUL_S1 DB 3
DB 0FH,70H
DW OFFSET NET_SIMUL_S1+5
DB 1,2,0,2
DW 8101H
DW OFFSET RUN_S1
DW 0
DW OFFSET RUN_HELP1
DB 1,2,4,17
DW 0
DW OFFSET NET_OD_S1
DW OFFSET NET_OD_MESS1
DW OFFSET NET_OD_HELP1
DB 1,2,19,22
DW 8300H
DW OFFSET QUIT_HANDLER
DW ?
DW OFFSET QUIT_HELP1

```

```

E_CONFIG_S_MESS1 DB 0FFH,50H,0FH,0FFH,0,2
DB "SOURCE ENCODER MODULATOR TRANSMITTER_FILTER CHANNEL"
DB 0FFH,0,3
DB "RECEIVER_FILTER DEMODULATOR DECODER QUIT",0DH
E_CONFIG_S_HELP1 DB 0DH,"$"
E_CONFIG_S1 DB 9
DB 0FH,70H
DW OFFSET E_CONFIG_S1+5
DB 1,2,0,5
DW 2000H
DW OFFSET SOURCE_S1
DW OFFSET SOURCE_MESS1
DW OFFSET SOURCE_HELP1
DB 1,2,8,14
DW 2000H
DW OFFSET ENCODER_S1
DW OFFSET ENCODER_MESS1

```

```

DW OFFSET ENCODER_HELP1
DB 1,2,16,24
DW 2000H
DW OFFSET MODULATOR_S1
DW OFFSET MODULATOR_MESS1
DW OFFSET MODULATOR_HELP1
DB 1,2,28,45
DW 2000H
DW OFFSET CHANNEL_S_S1
DW OFFSET CHANNEL_S_MESS1
DW OFFSET CHANNEL_S_HELP1
DB 1,2,47,53
DW 2000H
DW OFFSET CHANNEL_S1
DW OFFSET CHANNEL_MESS1
DW OFFSET CHANNEL_HELP1
DB 1,3,0,14
DW 2080H
DW OFFSET CHANNEL_S_S1
DW OFFSET CHANNEL_S_MESS1
DW OFFSET RECEIVER_HELP1
DB 1,3,16,26
DW 2000H
DW OFFSET DEMODULATOR_S1
DW OFFSET DEMODULATOR_MESS1
DW OFFSET DEMODULATOR_HELP1
DB 1,3,28,34
DW 2000H
DW OFFSET DECODER_S1
DW OFFSET DECODER_MESS1
DW OFFSET DECODER_HELP1
DB 1,3,38,41
DW 8400H
DW OFFSET QUIT_HANDLER
DW 0
DW OFFSET QUIT_HELP1

```

```

E_NETCON_S_MESS1 DB OFFH,50H,0FH,OFFH,0,2
  DB "STORE AND FORWARD MULTIPLE_ACCESS QUIT",0DH
E_NETCON_S1 DB 3
  DB 0FH,70H
  DW OFFSET E_NETCON_S1+5
  DB 1,2,0,16
  DW 2010H
  DW OFFSET SAF_S1
  DW OFFSET SAF_MESS1
  DW OFFSET SAF_HELP1
  DB 1,2,18,32
  DW 2020H
  DW OFFSET MA_S1
  DW OFFSET MA_MESS1
  DW OFFSET MA_HELP1
  DB 1,2,34,37
  DW 8400H
  DW OFFSET QUIT_HANDLER

```

```

DW 0
DW OFFSET QUIT_HELP1

SOURCE_MESS1 DB 0FFH,50H,0FH,0FFH,0,2
DB "RANDOM_BIT_STREAM SINUSOID QUIT",0DH
SOURCE_HELP1 DW OFFSET SOURCE_HELP
DW OFFSET SOURCE_POINTER
DW OFFSET RECOVER_SOURCE

SOURCE_HELP DB 0DH,"$"
SOURCE_S1 DB 3
DB 0FH,70H
DW OFFSET SOURCE_S1+5
DB 1,2,0,16
DW 8201H
DW OFFSET SOURCE_SET
DW 0
DW OFFSET RBS_HELP1
DB 1,2,18,25
DW 2
DW OFFSET SINU_S1
DW OFFSET SINU_MESS1
DW OFFSET SINU_HELP1
DB 1,2,27,30
DW 8500H
DW OFFSET QUIT_HANDLER
DW ?
DW OFFSET QUIT_HELP1

ENCODER_MESS1 DB 0FFH,50H,0FH,0FFH,0,2
DB "CONVOLUTIONAL REED_SOLOMON NONE QUIT",0DH
ENCODER_HELP1 DW OFFSET ENCODER_HELP
DW OFFSET ENCODER_POINTER
DW OFFSET RECOVER_ENCODER

ENCODER_HELP DB 0DH,"$"
ENCODER_S1 DB 4
DB 0FH,70H
DW OFFSET ENCODER_S1+5
DB 1,2,0,12
DW 2
DW OFFSET CC_S1
DW OFFSET CC_MESS1
DW OFFSET CC_HELP1
DB 1,2,14,25
DW 5
DW OFFSET RS_S1
DW OFFSET RS_MESS1
DW OFFSET RS_HELP1
DB 1,2,27,30
DW 8201H
DW OFFSET ENCODER_SET
DW 0
DW OFFSET NONE_EC_HELP1
DB 1,2,32,35
DW 8500H
DW OFFSET QUIT_HANDLER

```

DW ?  
DW OFFSET QUIT\_HELP1

MODULATOR\_MESS1 DB 0FFH,50H,0FH,0FFH,0,2  
DB "PSK\_QAM\_FSK\_NONE\_QUIT",0DH  
MODULATOR\_HELP1 DW OFFSET MODULATOR\_HELP  
DW OFFSET MODULATOR\_POINTER  
DW OFFSET RECOVER\_MODULATOR

MODULATOR\_HELP DB 0DH,"\$"

MODULATOR\_S1 DB 5  
DB 0FH,70H  
DW OFFSET MODULATOR\_S1+5  
DB 1,2,0,2  
DW 1  
DW OFFSET PSK\_S1  
DW OFFSET PSK\_MESS1  
DW OFFSET PSK\_HELP1  
DB 1,2,4,6  
DW 4  
DW OFFSET QAM\_S1  
DW OFFSET QAM\_MESS1  
DW OFFSET QAM\_HELP1  
DB 1,2,8,10  
DW 6  
DW OFFSET FSK\_S1  
DW OFFSET FSK\_MESS1  
DW OFFSET FSK\_HELP1  
DB 1,2,12,15  
DW 8209H  
DW OFFSET MODULATOR\_SET  
DW ?  
DW OFFSET QUIT\_HELP1  
DB 1,2,17,20  
DW 8500H  
DW OFFSET QUIT\_HANDLER  
DW ?  
DW OFFSET QUIT\_HELP1

CHANNEL\_S\_MESS1 DB 0FFH,50H,0FH,0FFH,0,2  
DB "BUTTERWORTH\_CHEBYCHEV\_ELLIPTIC\_FIR\_NONE\_QUIT"  
DB 0DH

CHANNEL\_S\_HELP1 DW OFFSET CHANNEL\_S\_HELP  
DW OFFSET CHANNEL\_S\_POINTER  
DW OFFSET RECOVER\_CHANNEL\_S

CHANNEL\_S\_HELP DB 0DH,"\$"

RECEIVER\_HELP1 DW OFFSET RECEIVER\_HELP  
DW OFFSET RECEIVER\_POINTER  
DW OFFSET RECOVER\_RECEIVER

RECEIVER\_HELP DB 0DH,"\$"

CHANNEL\_S\_S1 DB 6  
DB 0FH,70H  
DW OFFSET CHANNEL\_S\_S1+5  
DB 1,2,0,10  
DW 2003H  
DW OFFSET BW\_S1

DW OFFSET BW\_MESS1  
 DW OFFSET BW\_HELP1  
 DB 1,2,12,20  
 DW 2004H  
 DW OFFSET BW\_S1  
 DW OFFSET BW\_MESS1  
 DW OFFSET BW\_HELP1  
 DB 1,2,22,29  
 DW 2005H  
 DW OFFSET BW\_S2  
 DW OFFSET BW\_MESS2  
 DW OFFSET BW\_HELP1  
 DB 1,2,31,33  
 DW 2006H  
 DW OFFSET FIR\_S1  
 DW OFFSET FIR\_MESS1  
 DW OFFSET FIR\_HELP1  
 DB 1,2,35,38  
 DW 8201H  
 DW OFFSET FILTER\_SET  
 DW 0  
 DW OFFSET NONE\_CS\_HELP1  
 DB 1,2,40,43  
 DW 8500H  
 DW OFFSET QUIT\_HANDLER  
 DW ?  
 DW OFFSET QUIT\_HELP1

CHANNEL\_MESS1 DB OFFH,50H,0FH,OFFH,0,2  
 DB "WHITE\_GAUSSIAN\_NOISE\_PROPAGATION\_MODEL BSC NONE QUIT",0DH

CHANNEL\_HELP1 DW OFFSET CHANNEL\_HELP  
 DW OFFSET CHANNEL\_POINTER  
 DW OFFSET RECOVER\_CHANNEL

CHANNEL\_HELP DB 0DH,"\$"

CHANNEL\_S1 DB 5  
 DB 0FH,70H  
 DW OFFSET CHANNEL\_S1+5  
 DB 1,2,0,19  
 DW 8202H  
 DW OFFSET CHANNEL\_SET  
 DW 0  
 DW OFFSET WGNC\_HELP1  
 DB 1,2,21,37  
 DW 2000H  
 DW OFFSET PM\_S1  
 DW OFFSET PM\_MESS  
 DW OFFSET PM\_HELP1  
 DB 1,2,39,41  
 DW 8203H  
 DW OFFSET CHANNEL\_SET  
 DW 0  
 DW OFFSET BSC\_HELP1  
 DB 1,2,43,46  
 DW 8201H  
 DW OFFSET CHANNEL\_SET

```

DW 0
DW OFFSET NONE_C_HELP1
DB 1,2,48,51
DW 8500H
DW OFFSET QUIT_HANDLER
DW ?
DW OFFSET QUIT_HELP1

PM_MESS DB OFFH,50H,0FH,OFFH,0,2
DB "TRANSMITTER RECEIVER CARRIER_FREQUENCY DISTANCE AREA_TYPE"
DB OFFH,0,3
DB "FOLIAGE_CHARACTERISTIC QUIT"
DB 0DH

PM_HELP1 DW OFFSET PM_HELP
          DW OFFSET PM_DISPLAY
          DW OFFSET DUMMY_CALL

PM_HELP DB 0DH,"$"
PM_S1 DB 7
DB 0FH,70H
DW OFFSET PM_S1+5
DB 1,2,0,10
DW 0
DW OFFSET TX_S1
DW OFFSET TX_MESS1
DW OFFSET BW_HELP
DB 1,2,12,19
DW 0
DW OFFSET RX_S1
DW OFFSET RX_MESS1
DW OFFSET BW_HELP
DB 1,2,21,37
DW 8100H
DW OFFSET PM_CF
DW 0
DW OFFSET BW_HELP
DB 1,2,39,46
DW 8100H
DW OFFSET PM_DS
DW 0
DW OFFSET FIR_HELP
DB 1,2,48,56
DW 0
DW OFFSET AREA_S1
DW OFFSET AREA_MESS
DW OFFSET NONE_CS_HELP1
DB 1,3,0,21
DW 0
DW OFFSET FC_S1
DW OFFSET FC_MESS
DW OFFSET QUIT_HELP1
DB 1,3,23,26
DW 8100H
DW OFFSET QUIT_HANDLER
DW 0
DW OFFSET QUIT_HELP1

```

TX\_MESS1 DB OFFH,50H,OFH,OFFH,0,2  
 DB "ANTENNA\_HEIGHT ANTENNA\_GAIN POWER QUIT"  
 DB 0DH  
 TX\_S1 DB 4  
 DB OFFH,70H  
 DW OFFSET TX\_S1+5  
 DB 1,2,0,13  
 DW 8100H  
 DW OFFSET PM\_AT\_H\_T  
 DW 0  
 DW OFFSET BW\_HELP  
 DB 1,2,15,26  
 DW 8100H  
 DW OFFSET PM\_AT\_G\_T  
 DW 0  
 DW OFFSET BW\_HELP  
 DB 1,2,28,32  
 DW 8100H  
 DW OFFSET PM\_POWER  
 DW 0  
 DW OFFSET BW\_HELP  
 DB 1,2,34,37  
 DW 8100H  
 DW OFFSET QUIT\_HANDLER  
 DW 0  
 DW OFFSET FIR\_HELP

RX\_MESS1 DB OFFH,50H,OFH,OFFH,0,2  
 DB "ANTENNA\_HEIGHT ANTENNA\_GAIN NOISE\_POWER BANDWIDTH "  
 DB "QUIT"  
 DB 0DH  
 RX\_S1 DB 5  
 DB OFFH,70H  
 DW OFFSET RX\_S1+5  
 DB 1,2,0,13  
 DW 8100H  
 DW OFFSET PM\_AT\_H\_R  
 DW 0  
 DW OFFSET BW\_HELP  
 DB 1,2,15,26  
 DW 8100H  
 DW OFFSET PM\_AT\_G\_R  
 DW 0  
 DW OFFSET BW\_HELP  
 DB 1,2,28,38  
 DW 8100H  
 DW OFFSET PM\_NOISE  
 DW 0  
 DW OFFSET BW\_HELP  
 DB 1,2,40,48  
 DW 8100H  
 DW OFFSET PM\_BANDWIDTH  
 DW 0  
 DW OFFSET BW\_HELP

DB 1,2,50,53  
DW 8100H  
DW OFFSET QUIT\_HANDLER  
DW 0  
DW OFFSET FIR\_HELP

AREA\_MESS DB OFFH,50H,OFH,OFFH,0,2  
DB "OPEN SUBURBAN QUIT"  
DB 0DH

AREA\_S1 DB 3  
DB 0FH,70H  
DW OFFSET AREA\_S1+5  
DB 1,2,0,3  
DW 8200H  
DW OFFSET PM\_AREA  
DW 0  
DW OFFSET BW\_HELP  
DB 1,2,5,12  
DW 8201H  
DW OFFSET PM\_AREA  
DW 0  
DW OFFSET BW\_HELP  
DB 1,2,14,17  
DW 8100H  
DW OFFSET QUIT\_HANDLER  
DW 0  
DW OFFSET FIR\_HELP

FC\_MESS DB OFFH,50H,OFH,OFFH,0,2  
DB "NO\_FOLIAGE FOLIAGED QUIT"  
DB 0DH

FC\_S1 DB 3  
DB 0FH,70H  
DW OFFSET FC\_S1+5  
DB 1,2,0,9  
DW 8200H  
DW OFFSET PM\_FOLIAGE  
DW 0  
DW OFFSET BW\_HELP  
DB 1,2,11,18  
DW 8202H  
DW OFFSET PM\_FOLIAGE  
DW 0  
DW OFFSET BW\_HELP  
DB 1,2,20,23  
DW 8100H  
DW OFFSET QUIT\_HANDLER  
DW 0  
DW OFFSET FIR\_HELP

DEMOMULATOR\_MESS1 DB OFFH,50H,OFH,OFFH,0,2  
DB "SOFT HARD NONE QUIT",0DH  
DEMOMULATOR\_HELP1 DW OFFSET DEMOMULATOR\_HELP  
DW OFFSET DEMOMULATOR\_POINTER  
DW OFFSET RECOVER\_DEMOMULATOR

```

DEMODULATOR_HELP DB 0DH,"$"
DEMODULATOR_S1 DB 4
  DB 0FH,70H
  DW OFFSET DEMODULATOR_S1+5
  DB 1,2,0,3
  DW 8201H
  DW OFFSET DEMODULATOR_SET
  DW 0
  DW OFFSET SOFT_HELP1
  DB 1,2,5,8
  DW 8202H
  DW OFFSET DEMODULATOR_SET
  DW 0
  DW OFFSET HARD_HELP1
  DB 1,2,10,13
  DW 8203H
  DW OFFSET DEMODULATOR_SET
  DW ?
  DW OFFSET QUIT_HELP1
  DB 1,2,15,18
  DW 8500H
  DW OFFSET QUIT_HANDLER
  DW ?
  DW OFFSET QUIT_HELP1

DECODER_MESS1 DB 0FFH,50H,0FH,0FFH,0,2
  DB "CONVOLUTIONAL REED SOLOMON NONE QUIT",0DH
DECODER_HELP1 DW OFFSET DECODER_HELP
  DW OFFSET DECODER_POINTER
  DW OFFSET RECOVER_DECODER

DECODER_HELP DB 0DH,"$"
DECODER_S1 DB 4
  DB 0FH,70H
  DW OFFSET DECODER_S1+5
  DB 1,2,0,12
  DW 2
  DW OFFSET CCD_S1
  DW OFFSET CCD_MESS1
  DW OFFSET CCD_HELP1
  DB 1,2,14,25
  DW 8204H
  DW OFFSET DECODER_SET
  DW 0
  DW OFFSET RSD_HELP1
  DB 1,2,27,30
  DW 8201H
  DW OFFSET DECODER_SET
  DW 0
  DW OFFSET NONED_HELP1
  DB 1,2,32,35
  DW 8500H
  DW OFFSET QUIT_HANDLER
  DW ?
  DW OFFSET QUIT_HELP1

```

```

SAF_MESS1 DB OFFH,50H,0FH,OFFH,0,2
DB "TOPOLOGY_NUMBER_OF_NODES_INPUT_TRAFFIC_MATRIX "
DB "REAL_TIME_ANIMATION_QUIT",0DH
SAF_HELP1 DW OFFSET SAF_HELP
DW OFFSET SET_NET_CON
DW OFFSET DUMMY_CALL
SAF_HELP DB 0DH,"$"
SAF_S1 DB 5
DB 0FH,70H
DW OFFSET SAF_S1+5
DB 1,2,0,7
DW 0
DW OFFSET SAFT_S1
DW OFFSET SAFT_MESS1
DW OFFSET SAFT_HELP1
DB 1,2,9,23
DW 8100H
DW OFFSET NON_SET
DW 0
DW OFFSET NON_HELP1
DB 1,2,25,44
DW 8100H
DW OFFSET ITM_SET
DW 0
DW OFFSET ITM_HELP1
DB 1,2,46,64
DW 0
DW OFFSET REAL_TIME_S1
DW OFFSET REAL_TIME_MESS1
DW OFFSET REAL_TIME_HELP1
DB 1,2,66,69
DW 8500H
DW OFFSET QUIT_HANDLER
DW ?
DW OFFSET QUIT_HELP1

```

```

SAFT_MESS1 DB OFFH,50H,0FH,OFFH,0,2
DB "STAR_LOOP_QUIT",0DH
SAFT_HELP1 DB 0DH,"$"
SAFT_S1 DB 3
DB 0FH,70H
DW OFFSET SAFT_S1+5
DB 1,2,0,3
DW 2001H
DW OFFSET LC_S1
DW OFFSET LC_MESS1
DW OFFSET STAR_HELP1
DB 1,2,5,8
DW 0
DW OFFSET LOOP_S1
DW OFFSET LOOP_MESS1
DW OFFSET LOOP_HELP1
DB 1,2,10,13
DW 8500H
DW OFFSET QUIT_HANDLER

```

DW ?  
DW OFFSET QUIT\_HELP1

LOOP\_MESS1 DB OFFH,50H,0FH,OFFH,0,2  
DB "UNIDIRECTIONAL\_LOOP BIDIRECTIONAL\_LOOP QUIT",ODH  
LOOP\_HELP1 DB ODH,"\$"   
LOOP\_S1 DB 3  
DB 0FH,70H  
DW OFFSET LOOP\_S1+5  
DB 1,2,0,18  
DW 2002H  
DW OFFSET LC\_S2  
DW OFFSET LC\_MESS1  
DW OFFSET UD\_LOOP\_HELP1  
DB 1,2,20,37  
DW 2003H  
DW OFFSET LC\_S2  
DW OFFSET LC\_MESS1  
DW OFFSET BD\_LOOP\_HELP1  
DB 1,2,39,42  
DW 8500H  
DW OFFSET QUIT\_HANDLER  
DW ?  
DW OFFSET QUIT\_HELP1

LC\_MESS1 DB OFFH,50H,0FH,OFFH,0,2  
DB "LINK\_CAPACITIES QUIT",ODH

LC\_S1 DB 2  
DB 0FH,70H  
DW OFFSET LC\_S1+5  
DB 1,2,0,14  
DW 8300H  
DW OFFSET LC\_SET  
DW 0  
DW OFFSET LC\_HELP1  
DB 1,2,16,19  
DW 8500H  
DW OFFSET QUIT\_HANDLER  
DW ?  
DW OFFSET QUIT\_HELP1

LC\_S2 DB 2  
DB 0FH,70H  
DW OFFSET LC\_S2+5  
DB 1,2,0,14  
DW 8400H  
DW OFFSET LC\_SET  
DW 0  
DW OFFSET LC\_HELP1  
DB 1,2,16,19  
DW 8500H  
DW OFFSET QUIT\_HANDLER  
DW ?  
DW OFFSET QUIT\_HELP1

REAL\_TIME\_MESS1 DB OFFH,50H,0FH,OFFH,0,2

```

DB "ON OFF QUIT",ODH
REAL_TIME_HELP1 DB ODH,"$"
REAL_TIME_S1 DB 3
  DB OFFH,70H
  DW OFFSET REAL_TIME_S1+5
  DB 1,2,0,1
  DW 8201H
  DW OFFSET REAL_TIME_SET
  DW 0
  DW OFFSET SP_HELP1
  DB 1,2,3,5
  DW 8200H
  DW OFFSET REAL_TIME_SET
  DW 0
  DW OFFSET RR_HELP1
  DB 1,2,7,10
  DW 8600H
  DW OFFSET QUIT_HANDLER
  DW ?
  DW OFFSET QUIT_HELP1

MA_MESS1 DB OFFH,50H,OFFH,OFFH,0,2
  DB "PROTOCOL USER_POPULATION_NUMBER_OF_CHANNELS "
  DB "INPUT_TRAFFIC_RATE"
  DB OFFH,0,3,"REAL_TIME_ANIMATION_QUIT",ODH
MA_HELP1 DW OFFSET MA_HELP
  DW OFFSET SET_NET_CON
  DW OFFSET DUMMY_CALL
MA_HELP DB ODH,"$"
MA_S1 DB 6
  DB OFFH,70H
  DW OFFSET MA_S1+5
  DB 1,2,0,7
  DW 0
  DW OFFSET PRO_S1
  DW OFFSET PRO_S_MESS1
  DW OFFSET PRO_S_HELP1
  DB 1,2,9,23
  DW 8100H
  DW OFFSET UP_SET
  DW 0
  DW OFFSET UP_HELP1
  DB 1,2,25,42
  DW 8100H
  DW OFFSET NOC_SET
  DW ?
  DW OFFSET NOC_HELP1
  DB 1,2,44,61
  DW 8100H
  DW OFFSET ITM_SET1
  DW ?
  DW OFFSET ITR_HELP1
  DB 1,3,0,18
  DW 0
  DW OFFSET REAL_TIME_S1

```

```

DW OFFSET REAL_TIME_MESS1
DW OFFSET REAL_TIME_HELP1
DB 1,3,20,23
DW 8500H
DW OFFSET QUIT_HANDLER
DW ?
DW OFFSET QUIT_HELP1

PRO_S_MESS1 DB OFFH,50H,0FH,OFFH,0,2
DB "ALOHA TREE CSMA QUIT",0DH
PRO_S_HELP1 DB 0DH,"$"
PRO_S1 DB 3
DB 0FH,70H
DW OFFSET PRO_S1+5
DB 1,2,0,4
DW 2001H
DW OFFSET ALOHA_S1
DW OFFSET ALOHA_MESS1
DW OFFSET ALOHA_HELP1
DB 1,2,6,9
DW 8202H
DW OFFSET MA_PRO_SET
DW 0
DW OFFSET TCR_HELP1
DB 1,2,11,14
DW 2003H
DW OFFSET CSMA_S1
DW OFFSET CSMA_MESS1
DW OFFSET CSMA_HELP1
DB 1,2,16,19
DW 8600H
DW OFFSET QUIT_HANDLER
DW ?
DW OFFSET QUIT_HELP1

ALOHA_MESS1 DB OFFH,50H,0FH,OFFH,0,2
DB "MAXIMUM BACKOFF_DELAY QUIT",0DH
ALOHA_HELP1 DW OFFSET MA_HELP
DW OFFSET MA_PRO_SET
DW OFFSET DUMMY_CALL

ALOHA_S1 DB 2
DB 0FH,70H
DW OFFSET ALOHA_S1+5
DB 1,2,0,20
DW 8300H
DW OFFSET AMBD_S1
DW 0
DW OFFSET MA_HELP
DB 1,2,22,25
DW 8600H
DW OFFSET QUIT_HANDLER
DW ?
DW OFFSET QUIT_HELP1

CSMA_MESS1 DB OFFH,50H,0FH,OFFH,0,2

```

DB "MAXIMUM\_BACKOFF\_DELAY PROPAGATION\_RATIO QUIT",ODH  
CSMA\_HELP1 DW OFFSET MA\_HELP  
DW OFFSET MA\_PRO\_SET  
DW OFFSET DUMMY\_CALL

CSMA\_S1 DB 3  
DB 0FH,70H  
DW OFFSET CSMA\_S1+5  
DB 1,2,0,20  
DW 8100H  
DW OFFSET AMBD\_S1  
DW 0  
DW OFFSET MA\_HELP  
DB 1,2,22,38  
DW 8101H  
DW OFFSET AMBD\_S1  
DW 0  
DW OFFSET MA\_HELP  
DB 1,2,40,43  
DW 8600H  
DW OFFSET QUIT\_HANDLER  
DW ?  
DW OFFSET QUIT\_HELP1

SINU\_MESS1 DB 0FH,50H,0FH,0FH,0,2  
DB "SINUSOID\_FREQUENCY SAMPLING\_FREQUENCY QUIT",ODH  
SINU\_HELP1 DB 0DH,"\$"

SINU\_S1 DB 3  
DB 0FH,70H  
DW OFFSET SINU\_S1+5  
DB 1,2,0,17  
DW 8100H  
DW OFFSET SOURCE\_SET  
DW 0  
DW OFFSET SINU\_F\_HELP1  
DB 1,2,19,36  
DW 8110H  
DW OFFSET SOURCE\_SET  
DW 0  
DW OFFSET SINU\_A\_HELP1  
DB 1,2,38,41  
DW 8600H  
DW OFFSET QUIT\_HANDLER  
DW ?  
DW OFFSET QUIT\_HELP1

CC\_MESS1 DB 0FH,50H,0FH,0FH,0,2  
DB "(r=1/2,K=3)\_BINARY (r=1/2,K=7)\_BINARY QUIT",ODH  
CC\_HELP1 DB 0DH,"\$"

CC\_S1 DB 3  
DB 0FH,70H  
DW OFFSET CC\_S1+5  
DB 1,2,0,17  
DW 8300H  
DW OFFSET ENCODER\_SET  
DW 0

DW OFFSET CCR2K3\_HELP1  
DB 1,2,19,36  
DW 8301H  
DW OFFSET ENCODER\_SET  
DW 0  
DW OFFSET CCR2K7\_HELP1  
DB 1,2,38,41  
DW 8600H  
DW OFFSET QUIT\_HANDLER  
DW ?  
DW OFFSET QUIT\_HELP1

RS\_MESS1 DB OFFH,50H,0FH,OFFH,0,2  
DB "(15,9) QUIT",ODH  
RS\_HELP1 DB 0DH,"\$"   
RS\_S1 DB 2  
DB 0FH,70H  
DW OFFSET RS\_S1+5  
DB 1,2,0,5  
DW 8300H  
DW OFFSET ENCODER\_SET  
DW 0  
DW OFFSET RS15\_9\_HELP1  
DB 1,2,7,10  
DW 8600H  
DW OFFSET QUIT\_HANDLER  
DW ?  
DW OFFSET QUIT\_HELP1

CCD\_MESS1 DB OFFH,50H,0FH,OFFH,0,2  
DB "SOFT\_DECISION\_VITERBI HARD\_DECISION\_VITERBI QUIT",ODH  
CCD\_HELP1 DB 0DH,"\$"   
CCD\_S1 DB 3  
DB 0FH,70H  
DW OFFSET CCD\_S1+5  
DB 1,2,0,20  
DW 8300H  
DW OFFSET DECODER\_SET  
DW 0  
DW OFFSET CCD\_SV\_HELP1  
DB 1,2,22,42  
DW 8301H  
DW OFFSET DECODER\_SET  
DW 0  
DW OFFSET CCD\_HV\_HELP1  
DB 1,2,44,47  
DW 8600H  
DW OFFSET QUIT\_HANDLER  
DW ?  
DW OFFSET QUIT\_HELP1

BW\_MESS1 DB OFFH,50H,0FH,OFFH,0,2  
DB "BY\_ORDER BY\_ATTENUATION QUIT",ODH  
BW\_HELP1 DW OFFSET BW\_HELP  
DW OFFSET BW\_SET

DW OFFSET DUMMY\_CALL

BW\_HELP DB ODH,"\$"

BW\_S1 DB 3

DB 0FH,70H

DW OFFSET BW\_S1+5

DB 1,2,0,7

DW 2000H

DW OFFSET ORDER\_S1

DW OFFSET ORDER\_MESS1

DW OFFSET ORDER\_HELP1

DB 1,2,9,22

DW 2040H

DW OFFSET ATTN\_S1

DW OFFSET ATTN\_MESS1

DW OFFSET ATTN\_HELP1

DB 1,2,24,27

DW 8600H

DW OFFSET QUIT\_HANDLER

DW ?

DW OFFSET QUIT\_HELP1

BW\_MESS2 DB OFFH,50H,0FH,OFFH,0,2

DB "BY\_ATTENUATION QUIT",ODH

BW\_S2 DB 2

DB 0FH,70H

DW OFFSET BW\_S2+5

DB 1,2,0,13

DW 2040H

DW OFFSET ATTN\_S1

DW OFFSET ATTN\_MESS1

DW OFFSET ATTN\_HELP1

DB 1,2,15,18

DW 8600H

DW OFFSET QUIT\_HANDLER

DW ?

DW OFFSET QUIT\_HELP1

FIR\_MESS1 DB OFFH,50H,0FH,OFFH,0,2

DB "BY\_ATTENUATION BY\_COEFFICIENTS SAVE\_FIR RETRIEVE\_FIR "

DB "DOS QUIT",ODH

FIR\_HELP1 DW OFFSET FIR\_HELP

DW OFFSET BW\_SET

DW OFFSET DUMMY\_CALL

FIR\_HELP DB ODH,"\$"

FIR\_S1 DB 6

DB 0FH,70H

DW OFFSET FIR\_S1+5

DB 1,2,0,13

DW 2040H

DW OFFSET ATTN\_S1

DW OFFSET ATTN\_MESS1

DW OFFSET ATTN\_HELP1

DB 1,2,15,29

DW 2000H

DW OFFSET COEF\_S1

```

DW OFFSET COEF_MESS1
DW OFFSET COEF_HELP1
DB 1,2,31,38
DW 8100H
DW OFFSET FIR_SAVE1
DW ?
DW OFFSET FIR_S_HELP1
DB 1,2,40,51
DW 8100H
DW OFFSET FIR_RETRIEVE1
DW ?
DW OFFSET FIR_R_HELP1
DB 1,2,53,55
DW 8100H
DW OFFSET DOS_HANDLER
DW ?
DW OFFSET DOS_HELP1
DB 1,2,57,60
DW 8600H
DW OFFSET QUIT_HANDLER
DW ?
DW OFFSET QUIT_HELP1

```

```

IIR_HELP1 DB 0DH,"$"

```

```

PSK_MESS1 DB 0FFH,50H,0FH,0FFH,0,2
DB "BPSK TIME_DOMAIN_BPSK QPSK 8-PSK QUIT",0DH
PSK_HELP1 DB 0DH,"$"
PSK_S1 DB 5
DB 0FH,70H
DW OFFSET PSK_S1+5
DB 1,2,0,3
DW 8300H
DW OFFSET MODULATOR_SET
DW 0
DW OFFSET BPSK_HELP1
DB 1,2,5,20
DW 8309H
DW OFFSET MODULATOR_SET
DW 0
DW OFFSET BPSK_HELP1
DB 1,2,22,25
DW 8301H
DW OFFSET MODULATOR_SET
DW 0
DW OFFSET QPSK_HELP1
DB 1,2,27,31
DW 8302H
DW OFFSET MODULATOR_SET
DW 0
DW OFFSET PSK8_HELP1
DB 1,2,33,36
DW 8600H
DW OFFSET QUIT_HANDLER
DW ?

```

DW OFFSET QUIT\_HELP1

QAM\_MESS1 DB OFFH,50H,0FH,OFFH,0,2

DB "16-QAM 64-QAM QUIT",ODH

QAM\_HELP1 DB ODH,"\$"

QAM\_S1 DB 3

DB 0FH,70H

DW OFFSET QAM\_S1+5

DB 1,2,0,5

DW 8300H

DW OFFSET MODULATOR\_SET

DW 0

DW OFFSET QAM16\_HELP1

DB 1,2,7,12

DW 8301H

DW OFFSET MODULATOR\_SET

DW 0

DW OFFSET QAM64\_HELP1

DB 1,2,14,17

DW 8600H

DW OFFSET QUIT\_HANDLER

DW ?

DW OFFSET QUIT\_HELP1

FSK\_MESS1 DB OFFH,50H,0FH,OFFH,0,2

DB "BFSK 4-FSK QUIT",ODH

FSK\_HELP1 DB ODH,"\$"

FSK\_S1 DB 4

DB 0FH,70H

DW OFFSET FSK\_S1+5

DB 1,2,0,3

DW 8300H

DW OFFSET MODULATOR\_SET

DW 0

DW OFFSET BFSK\_HELP1

DB 1,2,5,9

DW 8301H

DW OFFSET MODULATOR\_SET

DW 0

DW OFFSET QFSK\_HELP1

DB 1,2,11,14

DW 8600H

DW OFFSET QUIT\_HANDLER

DW ?

DW OFFSET QUIT\_HELP1

ATTN\_MESS1 DB OFFH,50H,0FH,OFFH,0,2

DB "SAMPLING\_FREQUENCY PASSBAND\_EDGE\_FREQUENCY "

DB "STOPBAND\_EDGE\_FREQUENCY"

DB OFFH,0,3

DB "PASSBAND\_ATTENUATION STOPBAND\_ATTENUATION QUIT",ODH

ATTN\_HELP1 DW OFFSET ATTN\_HELP

DW OFFSET ORDER\_SET

DW OFFSET DUMMY\_CALL

ATTN\_HELP DB ODH,"\$"

ATTN\_S1 DB 6  
 DB 0FH,70H  
 DW OFFSET ATTN\_S1+5  
 DB 1,2,0,17  
 DW 8100H  
 DW OFFSET FILTER\_SET  
 DW 0  
 DW OFFSET SF\_HELP1  
 DB 1,2,19,41  
 DW 8110H  
 DW OFFSET FILTER\_SET  
 DW 0  
 DW OFFSET CF\_HELP1  
 DB 1,2,43,65  
 DW 8118H  
 DW OFFSET FILTER\_SET  
 DW 0  
 DW OFFSET PA\_HELP1  
 DB 1,3,0,19  
 DW 8120H  
 DW OFFSET FILTER\_SET  
 DW 0  
 DW OFFSET SA\_HELP1  
 DB 1,3,21,40  
 DW 8130H  
 DW OFFSET FILTER\_SET  
 DW 0  
 DW OFFSET SA\_HELP1  
 DB 1,3,42,45  
 LW 8000H  
 DW OFFSET QUIT\_HANDLER  
 DW ?  
 DW OFFSET QUIT\_HELP1

ORDER\_MESS1 DB OFFH,50H,0FH,OFFH,0,2  
 DB "ORDER SAMPLING\_FREQUENCY CUTOFF\_FREQUENCY QUIT",ODH  
 ORDER\_HELP1 DW OFFSET ORDER\_HELP  
 DW OFFSET ORDER\_SET  
 DW OFFSET DUMMY\_CALL

ORDER\_HELP DB ODH,"\$"

ORDER\_S1 DB 5  
 DB 0FH,70H  
 DW OFFSET ORDER\_S1+5  
 DB 1,2,0,4  
 DW 8100H  
 DW OFFSET FILTER\_SET  
 DW 0  
 DW OFFSET ORDER2\_HELP1  
 DB 1,2,6,23  
 DW 8110H  
 DW OFFSET FILTER\_SET  
 DW 0  
 DW OFFSET SFO\_HELP1  
 DB 1,2,25,40  
 DW 8120H

DW OFFSET FILTER\_SET  
DW 0  
DW OFFSET CFO\_HELP1  
DB 1,2,42,45  
DW 8000H  
DW OFFSET QUIT\_HANDLER  
DW ?  
DW OFFSET QUIT\_HELP1

COEF\_MESS1 DB OFFH,50H,0FH,OFFH,0,2  
DB "ORDER SAMPLING\_FREQUENCY CUTOFF\_FREQUENCY "  
DB "COEFFICIENTS QUIT",ODH  
COEF\_HELP1 DW OFFSET COEF\_HELP  
DW OFFSET ORDER\_SET  
DW OFFSET DUMMY\_CALL

COEF\_HELP DB ODH,"\$"

COEF\_S1 DB 5  
DB 0FH,70H  
DW OFFSET COEF\_S1+5  
DB 1,2,0,4  
DW 8100H  
DW OFFSET FILTER\_SET  
DW 0  
DW OFFSET ORDER2\_HELP1  
DB 1,2,6,23  
DW 8110H  
DW OFFSET FILTER\_SET  
DW 0  
DW OFFSET SFO\_HELP1  
DB 1,2,25,40  
DW 8120H  
DW OFFSET FILTER\_SET  
DW 0  
DW OFFSET CFO\_HELP1  
DB 1,2,42,53  
DW 8130H  
DW OFFSET FILTER\_SET  
DW 0  
DW OFFSET CFO\_HELP1  
DB 1,2,55,58  
DW 8000H  
DW OFFSET QUIT\_HANDLER  
DW ?  
DW OFFSET QUIT\_HELP1

OD\_MESS1 DB OFFH,50H,0FH,OFFH,0,2  
DB "BIT\_ERROR\_RATE SIGNAL CONSTELLATIONS "  
DB "FILTER\_WAVEFORMS QUIT",ODH  
OD\_HELP1 DB ODH,"\$"  
OD\_S1 DB 4  
DB 0FH,70H  
DW OFFSET OD\_S1+5  
DB 0,2,0,13  
DW 8100H  
DW OFFSET BERH\_S1

```

DW ?
DW OFFSET BERH_HELP1
DB 0,2,15,35
DW 8100H
DW OFFSET SC_S1
DW ?
DW OFFSET SC_HELP1
DB 0,2,37,52
DW 8100H
DW OFFSET FD_S1
DW ?
DW OFFSET SC_HELP1
DB 1,2,54,57
DW 8000H
DW OFFSET QUIT_HANDLER
DW ?
DW OFFSET QUIT_HELP1

```

```

NET_OD_MESS1 DB OFFH,50H,0FH,OFFH,0,2
DB "STORE AND FORWARD MULTIPLE_ACCESS QUIT",ODH
NET_OD_HELP1 DB ODH,"$"
NET_OD_S1 DB 3
DB 0FH,70H
DW OFFSET NET_OD_S1+5
DB 1,2,0,16
DW 8200H
DW OFFSET SAF_OD_S1
DW 0
DW OFFSET SAF_OD_HELP1
DB 1,2,18,32
DW 8200H
DW OFFSET MA_OD_S1
DW 0
DW OFFSET MA_OD_HELP1
DB 1,2,34,37
DW 8300H
DW OFFSET QUIT_HANDLER
DW ?
DW OFFSET QUIT_HELP1

```

```

SAF_OD_MESS1 DB OFFH,50H,0FH,OFFH,57H,0,7,4FH,24
DB OFFH,30,5,OFFH,50H,70H,"FINISH",OFFH,50H,0FH
DB " : F1"
DB OFFH,30,6,"PACKET DELAY: F2"
DB OFFH,30,7,"THROUGHPUT: F3"
DB OFFH,59H,19,9,6,35,"┐┘└┘"
DB OFFH,20,10,"Display Packet Delay: "
DB OFFH,50H,70H,"ON ",OFFH,50H,0FH
DB OFFH,20,12,"Enter Source Node Number: "
DB OFFH,20,14,"Enter Destination Node Number: "
DB OFFH,59H,19,17,6,35,"┐┘└┘"
DB OFFH,20,18,"Display Throughput: "
DB OFFH,50H,70H,"ON ",OFFH,50H,0FH
DB OFFH,20,20,"Enter Source Node Number: "
DB OFFH,20,22,"Enter Destination Node Number: "

```

```

DB OFFH,29,75,0DH
SAF_OD_MESS3 DB OFFH,50,10
                DB OFFH,50H,70H,"OFF",OFFH,50H,OFH,OFFH,79,25,0DH
SAF_OD_MESS5 DB OFFH,50,18
                DB OFFH,50H,70H,"OFF",OFFH,50H,OFH,OFFH,79,25,0DH
SAF_OD_PD_OFF DB OFFH,50H,7,OFFH,20,12
                DB "
                DB OFFH,20,14
                DB "
                DB "
                DB "
SAF_OD_PB_OFF DB OFFH,50H,7,OFFH,20,20
                DB "
                DB OFFH,20,22
                DB "
                DB "
                DB "
SAF_OD_CTL DB 3
                DB 30,5
                DB 35,5
                DB 0
                DB 50,10
                DB 52,10
                DB 1
                DB 50,18
                DB 52,18
                DB 2

MA_OD_MESS1 DB OFFH,50H,OFH,OFFH,57H,0,7,4FH,24
                DB OFFH,24,7,OFFH,50H,70H,"FINISH",OFFH,50H,OFH
                DB ": F1"
                DB OFFH,24,8,"PACKET DELAY: F2"
                DB OFFH,24,9,"THROUGHPUT: F3"
                DB OFFH,59H,19,11,5,28,"┐┘└┘"
                DB OFFH,20,13,"Display Packet Delay: "
                DB OFFH,50H,70H,"ON ",OFFH,50H,OFH
                DB OFFH,20,15,"Display Throughput: "
                DB OFFH,50H,70H,"ON ",OFFH,50H,OFH
                DB OFFH,79,25,0DH
MA_OD_MESS3 DB OFFH,44,13
                DB OFFH,50H,70H,"OFF",OFFH,50H,OFH,OFFH,79,25,0DH
MA_OD_MESS5 DB OFFH,44,15
                DB OFFH,50H,70H,"OFF",OFFH,50H,OFH,OFFH,79,25,0DH
MA_OD_CTL DB 3
                DB 24,7
                DB 29,7
                DB 0
                DB 44,13
                DB 46,13
                DB 1
                DB 44,15
                DB 46,15
                DB 2
MA_OD_CTL_K DB 3
                DW F1_KEY
                DB 0
                DW F2_KEY
                DB 1
                DW F3_KEY

```

DB 2

```
ANALY_MESS1 DB OFFH,50H,0FH,OFFH,0,2
  DB "VIEW_DATA VIEW_GRAPH QUIT",ODH
ANALY_HELP1 DB 0DH,"$"
ANALY_S1 DB 3
  DB 0FH,70H
  DW OFFSET ANALY_S1+5
  DB 1,2,0,8
  DW 0
  DW OFFSET VD_S1
  DW OFFSET VD_MESS1
  DW OFFSET RETRIEVE_HELP1
  DB 1,2,10,19
  DW 0
  DW OFFSET OD_S1
  DW OFFSET OD_MESS1
  DW OFFSET LINK_S_HELP1
  DB 1,2,21,24
  DW 8400H
  DW OFFSET QUIT_HANDLER
  DW ?
  DW OFFSET QUIT_HELP1

ANALY_S2 DB 3
  DB 0FH,70H
  DW OFFSET ANALY_S2+5
  DB 1,2,0,8
  DW 8103H
  DW OFFSET CI_S1
  DW 0
  DW OFFSET RETRIEVE_HELP1
  DB 1,2,10,19
  DW 0
  DW OFFSET NVD_S1
  DW OFFSET NVD_MESS1
  DW OFFSET LINK_S_HELP1
  DB 1,2,21,24
  DW 8400H
  DW OFFSET QUIT_HANDLER
  DW ?
  DW OFFSET QUIT_HELP1

NVD_MESS1 DB OFFH,50H,0FH,OFFH,0,2
  DB "PACKET_DELAY THROUGHPUT QUIT",ODH
NVD_S1 DB 3
  DB 0FH,70H
  DW OFFSET NVD_S1+5
  DB 1,2,0,11
  DW 8101H
  DW OFFSET CI_S1
  DW 0
  DW OFFSET RETRIEVE_HELP1
  DB 1,2,13,22
```

```

DW 8102H
DW OFFSET CI_S1
DW 0
DW OFFSET LINK_S_HELP1
DB 1,2,24,27
DW 8400H
DW OFFSET QUIT_HANDLER
DW ?
DW OFFSET QUIT_HELP1

VD_MESS1 DB OFFH,50H,0FH,OFFH,0,2
DB "PERFORMANCE_DATA TRANSMITTER_FILTER_DESIGN_DATA "
DB "RECEIVER_FILTER_DESIGN_DATA"
DB OFFH,0,3
DB "QUIT",0DH
VD_S1 DB 4
DB 0FH,70H
DW OFFSET VD_S1+5
DB 0,2,0,15
DW 8100H
DW OFFSET RE_S1
DW 0
DW OFFSET RETRIEVE_HELP1
DB 0,2,17,46
DW 8101H
DW OFFSET RE_S1
DW 0
DW OFFSET LINK_S_HELP1
DB 0,2,48,74
DW 8102H
DW OFFSET RE_S1
DW 0
DW OFFSET LINK_S_HELP1
DB 1,3,0,3
DW 8400H
DW OFFSET QUIT_HANDLER
DW ?
DW OFFSET QUIT_HELP1

BERH_HELP1 DB 0DH,"$"
FIR_S_HELP1 DB 0DH,"$"
FIR_R_HELP1 DB 0DH,"$"
DOS_HELP1 DB 0DH,"$"
YES_HELP1 DB 0DH,"$"
NO_HELP1 DB 0DH,"$"
SC_HELP1 DB 0DH,"$"
RUN_HELP1 DB 0DH,"$"
BPSK_HELP1 DB 0DH,"$"
QPSK_HELP1 DB 0DH,"$"
PSK8_HELP1 DB 0DH,"$"
BFSK_HELP1 DB 0DH,"$"
QFSK_HELP1 DB 0DH,"$"
FSK8_HELP1 DB 0DH,"$"
QAM16_HELP1 DB 0DH,"$"
QAM64_HELP1 DB 0DH,"$"

```

SOFT\_HELP1 DB ODH,"\$"
HARD\_HELP1 DB ODH,"\$"
SF\_HELP1 DB ODH,"\$"
CF\_HELP1 DB ODH,"\$"
PA\_HELP1 DB ODH,"\$"
SA\_HELP1 DB ODH,"\$"
CODED\_HELP1 DB ODH,"\$"
UNCODED\_HELP1 DB ODH,"\$"
NONE\_BE\_HELP1 DB ODH,"\$"
CCR2K3\_HELP1 DB ODH,"\$"
CCR2K7\_HELP1 DB ODH,"\$"
DUAL3\_HELP1 DB ODH,"\$"
RS15\_9\_HELP1 DB ODH,"\$"
RS255\_223\_HELP1 DB ODH,"\$"
NONE\_EC\_HELP1 DB ODH,"\$"
CCD\_SV\_HELP1 DB ODH,"\$"
CCD\_HV\_HELP1 DB ODH,"\$"
RSD\_HELP1 DB ODH,"\$"
R\_CONFIG\_S\_HELP1 DB ODH,"\$"
S\_CONFIG\_S\_HELP1 DB ODH,"\$"
WGN\_HELP1 DB ODH,"\$"
RBS\_HELP1 DB ODH,"\$"
SINU\_F\_HELP1 DB ODH,"\$"
SINU\_A\_HELP1 DB ODH,"\$"
ORDER2\_HELP1 DB ODH,"\$"
SFO\_HELP1 DB ODH,"\$"
CFO\_HELP1 DB ODH,"\$"
NONE\_CS\_HELP1 DB ODH,"\$"
NONE\_C\_HELP1 DB ODH,"\$"
NONED\_HELP1 DB ODH,"\$"
LP\_HELP1 DB ODH,"\$"
WGNC\_HELP1 DB ODH,"\$"
BSC\_HELP1 DB ODH,"\$"
DPSK\_HELP1 DB ODH,"\$"
QUIT\_HELP1 DB ODH,"\$"
CODEDS\_HELP1 DB ODH,"\$"
UNCODEDS\_HELP1 DB ODH,"\$"
RETRIEVE\_HELP1 DB ODH,"\$"
CI\_HELP1 DB ODH,"\$"
IS\_HELP1 DB ODH,"\$"
R\_NETCON\_S\_HELP1 DB ODH,"\$"
S\_NETCON\_S\_HELP1 DB ODH,"\$"
E\_NETCON\_S\_HELP1 DB ODH,"\$"
UP\_HELP1 DB ODH,"\$"
NOC\_HELP1 DB ODH,"\$"
NON\_HELP1 DB ODH,"\$"
ITR\_HELP1 DB ODH,"\$"
TCR\_HELP1 DB ODH,"\$"
LC\_HELP1 DB ODH,"\$"
ITM\_HELP1 DB ODH,"\$"
SP\_HELP1 DB ODH,"\$"
RR\_HELP1 DB ODH,"\$"
SAF\_PB\_HELP1 DB ODH,"\$"
SAF\_PD\_HELP1 DB ODH,"\$"
NONE\_OD\_HELP1 DB ODH,"\$"

```

MA_NONE_HELP1 DB 0DH,"$"
MA_PD_HELP1 DB 0DH,"$"
STAR_HELP1 DW OFFSET STAR_HELP
            DW OFFSET TOPOLOGY_SET
            DW OFFSET DUMMY_CALL
STAR_HELP DB 0DH,"$"
SAF_OD_HELP1 DB 0DH,"$"
MA_OD_HELP1 DB 0DH,"$"
UD_LOOP_HELP1 DW OFFSET UD_LOOP_HELP
              DW OFFSET TOPOLOGY_SET
              DW OFFSET DUMMY_CALL
UD_LOOP_HELP DB 0DH,"$"
BD_LOOP_HELP1 DW OFFSET BD_LOOP_HELP
              DW OFFSET TOPOLOGY_SET
              DW OFFSET DUMMY_CALL
BD_LOOP_HELP DB 0DH,"$"

```

```

CON_SCREEN DB OFFH,5BH,50H,OFFH,50H,0FH
            DB OFFH,59H,0,9,5,13,"  "
            DB OFFH,59H,17,9,5,13,"  "
            DB OFFH,59H,34,9,5,13,"  "
            DB OFFH,59H,51,9,5,13,"  "
            DB OFFH,59H,67,14,4,11,"  "
            DB OFFH,59H,0,18,5,13,"  "
            DB OFFH,59H,17,18,5,13,"  "
            DB OFFH,59H,34,18,5,13,"  "
            DB OFFH,59H,51,18,5,13,"  "
S1         DB OFFH,50H,7,OFFH,4,10,"SOURCE",0DH
S2         DB OFFH,50H,7,OFFH,21,10,"ENCODER",0DH
S3         DB OFFH,50H,7,OFFH,37,10,"MODULATOR",0DH
S4         DB OFFH,50H,7,OFFH,53,10,"TRANSMITTER"
            DB OFFH,55,11,"FILTER"
            DB 0DH
S5         DB OFFH,50H,7,OFFH,70,15,"CHANNEL",0DH
            DB OFFH,50H,7,OFFH,5,19,"SINK"
S6         DB OFFH,50H,7,OFFH,21,19,"DECODER",0DH
S7         DB OFFH,50H,7,OFFH,36,19,"DEMODULATOR",0DH
S8         DB OFFH,50H,7,OFFH,54,19,"RECEIVER",OFFH,55,20,"FILTER"
            DB 0DH
            DB OFFH,50H,0FH
            DB OFFH,14,12,"  "
            DB OFFH,31,12,"  "
            DB OFFH,48,12,"  "
            DB OFFH,65,12,"  "
            DB OFFH,73,13,"  "
            DB OFFH,73,14,"  "
            DB OFFH,14,21,"  "
            DB OFFH,31,21,"  "
            DB OFFH,48,21,"  "
            DB OFFH,73,19,"  "
            DB OFFH,73,20,"  "
            DB OFFH,65,21,"  "
            DB 0DH,"$"

```

```

CON_SCREEN_CLEAR DB OFFH,57H,0,9,79,24,0DH

```

T1 DB OFFH,50H,87H,OFFH,4,10,"SOURCE",0DH  
T2 DB OFFH,50H,87H,OFFH,21,10,"ENCODER",0DH  
T3 DB OFFH,50H,87H,OFFH,37,10,"MODULATOR",0DH  
T4 DB OFFH,50H,87H,OFFH,53,10,"TRANSMITTER",OFFH,55,11  
DB "FILTER",0DH  
T5 DB OFFH,50H,87H,OFFH,70,15,"CHANNEL",0DH  
T6 DB OFFH,50H,87H,OFFH,21,19,"DECODER",0DH  
T7 DB OFFH,50H,87H,OFFH,36,19,"DEMODULATOR",0DH  
T8 DB OFFH,50H,87H,OFFH,54,19,"RECEIVER",OFFH,55,20  
DB "FILTER",0DH  
MOD\_P DW OFFSET MOD\_BPSK  
DW OFFSET MOD\_QPSK  
DW OFFSET MOD\_8PSK  
DW OFFSET MOD\_16QAM  
DW OFFSET MOD\_64QAM  
DW OFFSET MOD\_BFSK  
DW OFFSET MOD\_4FSK  
DW OFFSET MOD\_8FSK  
DW OFFSET MOD\_NONE  
DW OFFSET MOD\_TBPSK  
MOD\_UDF DB OFFH,57H,35,11,47,14,OFFH,50H,7,0DH  
MOD\_BPSK DB OFFH,39,12,"BPSK",0DH  
MOD\_QPSK DB OFFH,39,12,"QPSK",0DH  
MOD\_8PSK DB OFFH,39,12,"8-PSK",0DH  
MOD\_16QAM DB OFFH,38,12,"16-QAM",0DH  
MOD\_64QAM DB OFFH,38,12,"64-QAM",0DH  
MOD\_BFSK DB OFFH,39,12,"BFSK",0DH  
MOD\_4FSK DB OFFH,39,12,"4-FSK",0DH  
MOD\_8FSK DB OFFH,39,12,"8-FSK",0DH  
MOD\_NONE DB OFFH,39,12,"None",0DH  
MOD\_TBPSK DB OFFH,36,12,"Time Domain",OFFH,39,13,"BPSK",0DH  
ECD\_P DW OFFSET ECD\_NONE  
DW OFFSET ECD\_CCK3  
DW OFFSET ECD\_CCK7  
DW OFFSET ECD\_DUAL3  
DW OFFSET ECD\_RS15  
DW OFFSET ECD\_RS255  
ECD\_UDF DB OFFH,57H,18,11,30,14,OFFH,50H,7,0DH  
ECD\_NONE DB OFFH,22,12,"None",0DH  
ECD\_CCK3 DB OFFH,19,12,"(r=1/2,K=3)",OFFH,18,13,"Convolutional"  
DB 0DH  
ECD\_CCK7 DB OFFH,19,12,"(r=1/2,K=7)",OFFH,18,13,"Convolutional"  
DB 0DH  
ECD\_DUAL3 DB OFFH,20,12,"Dual-3",OFFH,18,13,"Convolutional"  
DB 0DH  
ECD\_RS15 DB OFFH,20,12,"(15,9)",OFFH,18,13,"Reed-Solomon"  
DB 0DH  
ECD\_RS255 DB OFFH,19,12,"(255,223)",OFFH,18,13,"Reed-Solomon"  
DB 0DH  
SOU\_P DW OFFSET SOU\_RBS  
DW OFFSET SOU\_SINU  
DW OFFSET SOU\_SW  
DW OFFSET SOU\_WGN  
SOU\_UDF DB OFFH,57H,1,11,13,14,OFFH,50H,7,0DH  
SOU\_RBS DB OFFH,2,12,"Random Bit",OFFH,4,13,"Stream",0DH

SOU\_SINU DB OFFH,3,12,"Sinusoid",0DH  
 SOU\_SW DB OFFH,2,12,"Square Wave",0DH  
 SOU\_WGN DB OFFH,4,12,"White",OFFH,3,13,"Gaussian",OFFH,4,14  
     DB "Noise".0DH  
 DCD\_P DW OFFSET DCD\_NONE  
     DW OFFSET DCD\_SOFT  
     DW OFFSET DCD\_HARD  
     DW OFFSET DCD\_REED  
 DCD\_UDF DB OFFH,57H,18,20,30,23,OFFH,50H,7,0DH  
 DCD\_NONE DB OFFH,22,21,"None",0DH  
 DCD\_SOFT DB OFFH,18,21,"Soft Viterbi",0DH  
 DCD\_HARD DB OFFH,18,21,"Hard Viterbi",0DH  
 DCD\_REED DB OFFH,18,21,"Reed-Solomon",0DH  
 DEMOD\_P DW OFFSET DEMOD\_SOFT  
     DW OFFSET DEMOD\_HARD  
     DW OFFSET DEMOD\_NONE  
 DEMOD\_UDF DB OFFH,57H,35,20,47,23,OFFH,50H,7,0DH  
 DEMOD\_SOFT DB OFFH,39,21,"Soft",0DH  
 DEMOD\_HARD DB OFFH,39,21,"Hard",0DH  
 DEMOD\_NONE DB OFFH,39,21,"None",0DH  
 CF\_P DW OFFSET CF\_NONE  
     DW OFFSET CF\_LP  
     DW OFFSET CF\_BW  
     DW OFFSET CF\_CB  
     DW OFFSET CF\_EP  
     DW OFFSET CF\_FIR  
     DW OFFSET CF\_IIR  
 CF\_UDF DB OFFH,57H,52,12,64,14,OFFH,50H,7,0DH  
 CF\_NONE DB OFFH,56,13,"None",0DH  
 CF\_LP DB OFFH,54,13,"Low-Pass",0DH  
 CF\_BW DB OFFH,53,13,"Butterworth",0DH  
 CF\_CB DB OFFH,54,13,"Chebychev",0DH  
 CF\_EP DB OFFH,54,13,"Elliptic",0DH  
 CF\_FIR DB OFFH,57,13,"FIR",0DH  
 CF\_IIR DB OFFH,57,13,"IIR",0DH  
 CFT\_P1 DW OFFSET CFT\_ORDER  
     DW OFFSET CFT\_ATTEN  
 CFT\_P2 DW OFFSET CFT\_ORDER1  
     DW OFFSET CFT\_ATTEN  
 CFT\_UDF DB OFFH,57H,52,14,64,14,OFFH,50H,7,0DH  
 CFT\_ORDER DB OFFH,56,14,"Order",0DH  
 CFT\_ATTEN DB OFFH,53,14,"Attenuation",0DH  
 CFT\_ORDER1 DB OFFH,52,14,"Coefficients",0DH  
 RF\_P DW OFFSET RF\_NONE  
     DW OFFSET RF\_LP  
     DW OFFSET RF\_BW  
     DW OFFSET RF\_CB  
     DW OFFSET RF\_EP  
     DW OFFSET RF\_FIR  
     DW OFFSET RF\_IIR  
 RF\_UDF DB OFFH,57H,52,21,64,23,OFFH,50H,7,0DH  
 RF\_NONE DB OFFH,56,22,"None",0DH  
 RF\_LP DB OFFH,54,22,"Low-Pass",0DH  
 RF\_BW DB OFFH,53,22,"Butterworth",0DH  
 RF\_CB DB OFFH,54,22,"Chebychev",0DH

```

RF_EP DB OFFH,54,22,"Elliptic",0DH
RF_FIR DB OFFH,57,22,"FIR",0DH
RF_IIR DB OFFH,57,22,"IIR",0DH
RFT_P1 DW OFFSET RFT_ORDER
      DW OFFSET RFT_ATTEN
RFT_P2 DW OFFSET RFT_ORDER1
      DW OFFSET RFT_ATTEN
RFT_UDF DB OFFH,57H,52,23,64,23,OFFH,50H,7,0DH
RFT_ORDER DB OFFH,56,23,"Order",0DH
RFT_ATTEN DB OFFH,53,23,"Attenuation",0DH
RFT_ORDER1 DB OFFH,52,23,"Coefficients",0DH
CHL_P DW OFFSET CHL_NONE
      DW OFFSET CHL_WGN
      DW OFFSET CHL_BSC
      DW OFFSET CHL_PM
CHL_UDF DB OFFH,57H,68,16,78,18,OFFH,50H,7,0DH
CHL_NONE DB OFFH,71,17,"None",0DH
CHL_WGN DB OFFH,71,16,"White",OFFH,69,17,"Gaussian"
        DB OFFH,71,18,"Noise",0DH
CHL_BSC DB OFFH,72,17,"BSC",0DH
CHL_PM DB OFFH,68,16,"Propagation",OFFH,71,17,"Model",0DH

SIG_CON1 DB OFFH,50H,70H,OFFH,49,12,"—"
SIG_CON2 DB OFFH,50H,70H,OFFH,66,12,"—"
        DB OFFH,73,13,"|"
SIG_CON5 DB OFFH,50H,70H,OFFH,49,21,"—"
SIG_CON4 DB OFFH,50H,70H,OFFH,73,20,"|"
        DB OFFH,66,21,"—"
        DB OFFH,50H,70H,OFFH,50,3,"FINISH",OFFH,50H,0FH
        DB " : F1"
TX1 DB OFFH,50H,0FH,OFFH,50,4,"TRANSMITTER INPUT: F2"
TX2 DB OFFH,50H,0FH,OFFH,50,5,"TRANSMITTER OUTPUT: F3"
RX1 DB OFFH,50H,0FH,OFFH,50,6,"RECEIVER INPUT: F4"
RX2 DB OFFH,50H,0FH,OFFH,50,7,"RECEIVER OUTPUT: F5"
      DB OFFH,79,25,0DH
SIG_CON3 DB OFFH,50H,70H,OFFH,49,21,"—"
        DB OFFH,50H,70H,OFFH,45,6,"FINISH",OFFH,50H,0FH," : F1"
        DB OFFH,45,7,"PLOT: F2"
        DB OFFH,79,25,0DH
SIG_CON_M1 DB 2
          DB 49,21
          DB 50,21
          DB 8
          DB 45,6
          DB 50,6
          DB 10H
SIG_CON_KM1 DB 2
          DW F2_KEY
          DB 8
          DW F1_KEY
          DB 10H

SIG_CON_M2 DB 7
          DB 49,12
          DB 50,12

```

```

DB 2
DB 66,12
DB 73,12
DB 3
DB 73,13
DB 73,13
DB 3
DB 49,21
DB 50,21
DB 4
DB 73,20
DB 73,20
DB 5
DB 66,21
DB 73,21
DB 5
DB 50,3
DB 55,3
DB 10H
SIG_CON_KM2 DB 5
              DW F5_KEY
              DB 4
              DW F4_KEY
              DB 5
              DW F3_KEY
              DB 3
              DW F2_KEY
              DB 2
              DW F1_KEY
              DB 10H

BER_HIS1 DB OFFH,50H,70H,OFFH,15,21,"—"
          DB OFFH,50H,70H,OFFH,11,6,"FINISH",OFFH,50H,OFH,": F1"
          DB OFFH,11,7,"PLOT: F2"
          DB OFFH,79,25,0DH
BER_HIS_M1 DB 2
            DB 15,21
            DB 16,21
            DB 1
            DB 11,6
            DB 16,6
            DB 10H
BER_HIS_KM1 DB 2
             DW F2_KEY
             DB 1
             DW F1_KEY
             DB 10H

KEYBOARD_CONTROL_BUFFER DB 4
                          DB 0
                          DW 0
                          DW 0
                          DW 0
                          DW OFFSET MOUSE_HANDLER

```

STACK\_POINTER DW ?  
STACK\_POINTER1 DW ?  
STACK\_POINTER3 DW ?

PUBLIC BIT\_RATE,AMPLITUDE\_SIN,AMPLITUDE\_SW,SOU\_NO,SF\_BW\_CF  
PUBLIC CHL\_NO,CHL\_ER,SBU,MA\_UP,NET\_CON,SAF\_TP,SOU\_CON,MOD\_CON  
PUBLIC ECD\_CON,CF\_CON,RF\_CON,DCD\_CON,DEMOD\_CON,CHL\_CON,ITM\_MA  
PUBLIC ITM\_S,S\_NN,LC\_BL,LC\_UL,LC\_S,ORDER\_BW\_CF,FIR\_CF,MA\_CON  
PUBLIC PM\_CONST,PM\_CON,CF\_BW\_T,BER\_HIS,SIG\_CON,NET\_OD\_SAF,NET\_OD\_MA  
PUBLIC NUMBER\_ITERATION,SAF\_RTG,REAL\_TIME,PB\_SN,MA\_AMBF,MA\_CMBF

NUMBER\_OF\_INTEGER EQU 10

NUMBER\_OF\_DATA EQU 73

N\_B EQU 4002

LINK\_NET DB 0 ; Link : 0

; Network : 1

SOU\_CON DB 0 ; Undefined : 0

; Random bit : 1

; Sinusoidal : 2

; Square wave : 3

; White Gaussian : 4

MOD\_CON DB 0 ; Undefined : 0

; BPSK : 1

; QPSK : 2

; 8-PSK : 3

; 16-QAM : 4

; 64-QAM : 5

; BFSK : 6

; 4-FSK : 7

; 8-FSK : 8

; None : 9

; Time domain BPSK : 10

ECD\_CON DB 0 ; Undefined : 0

; None : 1

; (r=1/2,K=3) binary convolutional : 2

; (r=1/2,K=7) binary convolutional : 3

; Dual-3 : 4

; (15,9) Reed-Solomon : 5

; (255,223) Reed-Solomon : 6

CF\_CON DB 0 ; Undefined : 0

; None : 1

; Low-Pass : 2

; Butterworth : 3

; Chebychev : 4

; Elliptic : 5

; FIR : 6

; User : 7

RF\_CON DB 0 ; Undefined : 0

; None : 1

; Low-Pass : 2

; Butterworth : 3

; Chebychev : 4

; Elliptic : 5

; FIR : 6

; User : 7

```

DCD_CON DB 0 ; Undefined : 0
          ; None : 1
          ; Soft Viterbi : 2
          ; Hard Viterbi : 3
          ; Reed-Solomon : 4
DEMOD_CON DB 0 ; Undefined : 0
          ; Soft : 1
          ; Hard : 2
          ; None : 3
CHL_CON DB 0 ; Undefined : 0
          ; None : 1
          ; White Gaussian Noise : 2
          ; BSC : 3
          ; Propagation Model : 4
CF_BW_T DB 8 DUP (?) ; 0 : Undefined
          ; 1 : Order
          ; 2 : Attenuation
PM_CON DB 0 ; Bit 1 : 0 : Open
          ; 1 : Suburban
          ; Bit 2 : 0 : No Foliage
          ; 1 : Foliaged
ORDER_BW_CF DB 5 DUP (?)
            DB 5 DUP (?) ; ORDER_CB_CF
            DB 5 DUP (?) ; ORDER_EP_CF
            DB 5 DUP (?) ; ORDER_LP_CF
FI_CF_O DB 5 DUP (?) ; ORDER_FI_CF
        DB 5 DUP (?) ; ORDER_BW_RF
        DB 5 DUP (?) ; ORDER_CB_RF
        DB 5 DUP (?) ; ORDER_EP_RF
        DB 5 DUP (?) ; ORDER_LP_RF
FI_RF_O DB 5 DUP (?) ; ORDER_FI_RF
AMPLITUDE_SIN DB 15 DUP (?)
FREQUENCY_SIN DB 15 DUP (?)
AMPLITUDE_SW DB 15 DUP (?)
FREQUENCY_SW DB 15 DUP (?)
BIT_RATE DB 15 DUP (?)
SOU_NO DB 15 DUP (?)
CHL_NO DB 15 DUP (?)
CHL_ER DB 15 DUP (?)
PM_CONST DB 135 DUP (?) ; Transmitter Antenna Height,
          ; Receiver Antenna Height, Transmitter Average Power,
          ; Transmitter Antenna Gain, Receiver Antenna Gain,
          ; Carrier Frequency, Distance, Receiver Noise Temperature,
          ; Receiver Bandwidth
SF_BW_CF DB 15 DUP (?)
        DB 15 DUP (?)
        DB 15 DUP (?)
        DB 15 DUP (?)
        DB 15 DUP (?)
        DB 15 DUP (?)
        DB 15 DUP (?)
        DB 15 DUP (?) ; SF_CB_CF
        DB 15 DUP (?)
        DB 15 DUP (?)
        DB 15 DUP (?)

```

```

DB 15 DUP (?)
DB 15 DUP (?)
DB 15 DUP (?)
DB 15 DUP (?) ; SF_EP_CF
DB 15 DUP (?)
DB 15 DUP (?)
DB 15 DUP (?)
DB 15 DUP (?)
DB 15 DUP (?)
DB 15 DUP (?)
DB 15 DUP (?) ; SF_FI_CF
DB 15 DUP (?)
DB 15 DUP (?)
DB 15 DUP (?)
DB 15 DUP (?)
DB 15 DUP (?)
DB 15 DUP (?) ; SF_BW_RF
DB 15 DUP (?)
DB 15 DUP (?)
DB 15 DUP (?)
DB 15 DUP (?)
DB 15 DUP (?)
DB 15 DUP (?) ; SF_CB_RF
DB 15 DUP (?)
DB 15 DUP (?)
DB 15 DUP (?)
DB 15 DUP (?)
DB 15 DUP (?)
DB 15 DUP (?) ; SF_EP_RF
DB 15 DUP (?)
DB 15 DUP (?)
DB 15 DUP (?)
DB 15 DUP (?)
DB 15 DUP (?)
DB 15 DUP (?) ; SF_FI_RF
DB 15 DUP (?)
DB 15 DUP (?)
DB 15 DUP (?)
DB 15 DUP (?)
DB 15 DUP (?)
DB 15 DUP (?) ; Guard space

```

```

FIR_CF DW 99
DB 14
DB 1400 DUP (?)
DW 99 ; FIR_RF
DB 14
DB 1400 DUP (?)

```

```

BER_HIS DB 0 ; None : 0
          ; Coded : 1

```

```

; Uncoded : 2
SIG_CON DB 0 ; None : 0
; Modulator Output : 1
; Channel Filter Output : 2
; Channel Output : 4
; Receiver Filter Output : 8
SETUP DB 0 ; None : 0
; Coded : 1
; Uncoded : 2
NUMBER_ITERATION DB 20 DUP (?)

NETCON_C DB OFFH,57H,0,9,4FH,24,0DH
NETCON_M DB OFFH,0,9,0FFH,50H,0FH,"Type of Network: ",0DH
NETCON_SAF DB OFFH,17,9,"Store-and-Forward"
DB 0DH
NETCON_MA DB OFFH,17,9,"Multiple-Access "
DB 0DH
NETCON_UD DB OFFH,17,9,"Undefined "
DB 0DH
NETCON_MA_M DB OFFH,0,10,0FFH,50H,0FH,"Type of Protocol: ",0DH
NETCON_MA_A DB OFFH,18,10
DB "ALOHA ",0DH
NETCON_MA_TCR DB OFFH,18,10
DB "Tree ",0DH
NETCON_MA_CSMA DB OFFH,18,10
DB "CSMA ",0DH
NETCON_MA_UD DB OFFH,18,10
DB "Undefined ",0DH
NETCON_TP_M DB OFFH,0,10,0FFH,50H,0FH,"Type of Topology: ",0DH
NETCON_TP_S DB OFFH,18,10
DB "Star ",0DH
NETCON_TP_UL DB OFFH,18,10
DB "Unidirectional Loop",0DH
NETCON_TP_BL DB OFFH,18,10
DB "Bidirectional Loop ",0DH
NETCON_TP_UD DB OFFH,18,10
DB "Undefined ",0DH
NETCON_RG_M DB OFFH,0,11,0FFH,50H,0FH
DB "Type of Routing Algorithm: ",0DH
NETCON_RG_S DB OFFH,27,11
DB "Shortest-Path",0DH
NETCON_RG_R DB OFFH,27,11
DB "Random ",0DH
NETCON_RG_UD DB OFFH,27,11
DB "Undefined ",0DH
REAL_TIME_M DB OFFH,0,12,0FFH,50H,0FH
DB "Real Time Animation: ",0DH
REAL_TIME_ON DB OFFH,21,12
DB "On ",0DH
REAL_TIME_OFF DB OFFH,21,12
DB "Off",0DH

NC_N_B EQU 1344
NUMBER_OF_INTEGER2 EQU 3
NET_CON DB 0 ; 0 : Unknown

```

```

; 1 : Store-and-forward
; 2 : Multiple-access
SAF_TP DB 0 ; 0 : Unknown
; 1 : Star
; 2 : Unidirectional loop
; 3 : Bidirectional loop
SAF_RTG DB 0 ; 0 : Unknown
; 1 : Shortest path routing
; 2 : Random routing
MA_CON DB 0 ; 0 : Unknown
; 1 : ALOHA
; 2 : Tree collision resolution
; 3 : CSMA
S_NN DB 5 DUP (?)
MA_UP DB 5 DUP (?)
MA_NC DB 5 DUP (?)
MA_AMBF DB 10 DUP (?)
MA_CMBF DB 10 DUP (?)
DB 10 DUP (?) ; MA_CPR
ITM_S DW 64
DB 9
DB 630 DUP (?)
ITM_MA DW 20
DB 9
DB 200 DUP (?)
LC_S DW 16
DB 9
DB 180 DUP (?)
LC_UL DW 8
DB 9
DB 90 DUP (?)
LC_BL DW 16
DB 9
DB 180 DUP (?)
DB 10 DUP (?)
NET_OD_SAF DB 0 ; None : 0
; Packet delay : 1
; Packet throughput : 2
NET_OD_MA DB 0 ; None : 0
; Packet delay : 1
; Packet throughput : 2
REAL_TIME DB 0 ; Off : 0
; On : 1
PB_NM DW 4
DB 5
PB_SN DB 5 DUP (?)
PB_DN DB 5 DUP (?)
PD_SN DB 5 DUP (?)
PD_DN DB 5 DUP (?)
FILE_NAME DB 12
DB 0
DB 4
DW 425H

```

```

    DW 425H
    DW 431H
    DW 0
    DW 0
    DB 0
    DW 0
    DW ?
    DB 0
    DW 0
    DB OFFH,25H,4
    DB OFFH,50H,7
A10 DB 12 DUP (?)
    DB 0DH
ENTER_FILE DB OFFH,0BH,4,OFFH,50H,7
    DB "Enter File Specification: ",0DH
FILE_HANDLE DW ?
FLP_NUMBER DW OFFSET FRACTION
    DW OFFSET EXPONENT
    DW 1245H
    DW 1265H
    DB "E"
    DW 62FH
    DB 7
    DB 0
    DW ?
    DB 20H
    DW 0
    DW ?
    DW 3 DUP(?)
FRACTION DB 8
    DB 0
    DB 0
    DW 62EH
    DW 625H
    DW 62EH
    DW MINUS_KEY
    DW MINUS_KEY_NORMAL
    DB 0
    DW 0
    DW ?
    DB 1
    DW 0
    DB OFFH,25H,6
    DB OFFH,50H,7
    DB 8 DUP (?)
    DB 0DH
EXPONENT DB 3
    DB 0
    DB 0
    DW 632H
    DW 630H
    DW 632H
    DW MINUS_KEY
    DW MINUS_KEY_NORMAL
    DB 2

```

```

DW 0
DW ?
DB 1
DW 0
DB 0FFH,24H,6
DB 0FFH,50H,7
DB 3 DUP (?)
DB 0DH
FLP_NUMBER2 DW OFFSET FRACTION2
              DW OFFSET EXPONENT2
              DW 1245H
              DW 1265H
              DB "E"
              DW 52FH
              DB 7
              DB 0
              DW ?
              DB 20H
              DW 0
              DW ?
              DW 3 DUP(?)
FRACTION2 DB 5
           DB 0
           DB 0
           DW 52EH
           DW 525H
           DW 52EH
           DW MINUS_KEY
           DW MINUS_KEY_NORMAL
           DB 20H
           DW 0
           DW ?
           DB 1
           DW 0
           DB 0FFH,25H,5
           DB 0FFH,50H,7
           DB 5 DUP (?)
           DB 0DH
EXPONENT2 DB 2
          DB 0
          DB 0
          DW 533H
          DW 530H
          DW 533H
          DW MINUS_KEY
          DW MINUS_KEY_NORMAL
          DB 2
          DW 0
          DW ?
          DB 1
          DW 0
          DB 0FFH,25H,5
          DB 0FFH,50H,7
          DB 2 DUP (?)
          DB 0DH

```

```

INTEGERN DB 3
          DB 0
          DB 9
          DW 527H
          DW 51EH
          DW 527H
          DW MINUS_KEY
          DW MINUS_KEY_NORMAL
          DB 22H
          DW 0
          DW OFFSET TY_BEEP
          DB 2
          DW 0
          DB OFFH,1EH,5
          DB OFFH,50H,7
          DB 20H,31H,20H
          DB 0DH

```

```

INTEGERN1 DB 8
           DB 0
           DB 9
           DW 527H
           DW 51EH
           DW 527H
           DW MINUS_KEY
           DW MINUS_KEY_NORMAL
           DB 22H
           DW 0
           DW OFFSET TY_BEEP
           DB 2
           DW 0
           DB OFFH,1EH,5
           DB OFFH,50H,7
           DB 20H,30H,0,0,0,0,0,0
           DB 0DH

```

```

NB_ITERA DB 14
          DB 0
          DB 9
          DW 532H
          DW 529H
          DW 532H
          DW MINUS_KEY
          DW MINUS_KEY_NORMAL
          DB 22H
          DW 0
          DW OFFSET TY_BEEP
          DB 2
          DW 0
          DB OFFH,29H,5
          DB OFFH,50H,7
          DB 20H,31H
          DB 12 DUP (?)
          DB 0DH

```

```

ENTER_AMPL DB OFFH,0CH,6,OFFH,50H,7
              DB "Sampling Frequency (Hz): ",0DH
ENTER_FREQ DB OFFH,0CH,6,OFFH,50H,7
              DB "Sinusoid Frequency (Hz): ",0DH
ENTER_BITR DB OFFH,0FH,6,OFFH,50H,7
              DB "Enter Bit Rate (bps): ",0DH
ENTER_NO DB OFFH,11,5,OFFH,50H,7
           DB "Received Signal Power to Noise"
           DB OFFH,11,6,"Power Density Ratio (dB): ",0DH
S_P_NUMB DB OFFH,0,2,OFFH,50H,7,"Press the ",22H,"E",22H," key to"
           DB " toggle between the mantissa and exponent. Press the "
           DB 22H,"-",22H
           DB OFFH,0,3,"key to change the sign of the "
           DB "exponent when the exponent field is active."
S_INTEGE DB OFFH,0,4,OFFH,50H,7,"Press the Enter key to accept "
           DB "the number.",0DH
S_NUMBER DB OFFH,0,0,OFFH,50H,7,"Press the ",22H,"E",22H," key to"
           DB " toggle between the mantissa and exponent. Press the "
           DB 22H,"-",22H
           DB OFFH,0,1,"key to change the sign of the active field."
           DB OFFH,0,4,OFFH,50H,7,"Press the Enter key to accept "
           DB "the number.",0DH
A_NUMBER DB OFFH,0,0,OFFH,50H,7,"Press the ",22H,"E",22H," key to"
           DB " toggle between the mantissa and exponent. Press the "
           DB 22H,"-",22H
           DB OFFH,0,1,"key to change the sign of the active field."
           DB " Press the ",22H,"F1",22H," key "
           DB "to accept the"
           DB OFFH,0,2,"numbers.",0DH
A_P_NUMB DB OFFH,0,0,OFFH,50H,7,"Press the ",22H,"E",22H," key to"
           DB " toggle between the mantissa and exponent. Press the "
           DB 22H,"-",22H
           DB OFFH,0,1,"key to change the sign of the exponent when "
           DB "the exponent field is active."
           DB OFFH,0,2,"Press the ",22H,"F1",22H," key "
           DB "to accept the numbers.",0DH
LINK_C_UNIT DB OFFH,0,3,"The unit of the link capacities is "
             DB "packets/sec.",0DH
ITM_UNIT DB OFFH,0,3,"The unit of the input traffic is "
          DB "packets/sec.",0DH
ITM_UNIT1 DB OFFH,0,3,"The unit of the input traffic is "
           DB "packets/slot.",0DH
S_NI DB OFFH,9,5,OFFH,50H,7
      DB "Maximum Simulation Time: ",0DH
E_ORDER DB OFFH,18H,5,OFFH,50H,7
         DB "Enter Order: ",0DH
E_ORDER_ODD DB OFFH,10,5,OFFH,50H,7
             DB "Enter Order (Odd Integer): ",0DH
E_SAMPLE DB OFFH,6,6,OFFH,50H,7
          DB "Enter Sampling Frequency (Hz): ",0DH
E_CUTOFF DB OFFH,7,6,OFFH,50H,7
          DB "Enter Cutoff Frequency (Hz): ",0DH
E_PASEDG DB OFFH,0,6,OFFH,50H,7
          DB "Enter Passband Edge Frequency (Hz): ",0DH
E_STOEDG DB OFFH,0,6,OFFH,50H,7

```

```

DB "Enter Stopband Edge Frequency (Hz): ",0DH
E_SA DB OFFH,4,6,OFFH,50H,7
DB "Enter Stopband Attenuation (dB): ",0DH
E_PA DB OFFH,4,6,OFFH,50H,7
DB "Enter Passband Attenuation (dB): ",0DH
ENTER_ER DB OFFH,16,6,OFFH,50H,7
DB "Enter Bit Error Rate: ",0DH
TAH DB OFFH,5,6,OFFH,50H,7
DB "Transmitter Antenna Height (m): ",0DH
RAH DB OFFH,8,6,OFFH,50H,7
DB "Receiver Antenna Height (m): ",0DH
TAP DB OFFH,0,6,OFFH,50H,7
DB "Average Transmitter Power (in watts): ",0DH
TAG DB OFFH,6,6,OFFH,50H,7
DB "Transmitter Antenna Gain (dB): ",0DH
RAG DB OFFH,9,6,OFFH,50H,7
DB "Receiver Antenna Gain (dB): ",0DH
CFM DB OFFH,13,6,OFFH,50H,7
DB "Carrier Frequency (MHz):",0DH
DBTR DB OFFH,21,6,OFFH,50H,7
DB "Distance (km):",0DH
RNT DB OFFH,2,6,OFFH,50H,7
DB "Enter Receiver Noise Power (dBm): ",0DH
RBW DB OFFH,7,6,OFFH,50H,7
DB "Receiver Noise Bandwidth (Hz):",0DH
IL_FIR DB OFFH,56H,5,"J",OFFH,56H,5,"K",OFFH,57H,0,3,79,4
DB OFFH,27,3,"Illegal FIR data file!"
DB OFFH,27,4,"Press any key to continue.",OFFH,79,25,0DH
IL_COF DB OFFH,56H,5,"J",OFFH,56H,5,"K",OFFH,57H,0,3,79,4
DB OFFH,22,3,"Illegal configuration data file!"
DB OFFH,22,4,"Press any key to continue.",OFFH,79,25,0DH
O_ERROR DB OFFH,57H,0,5,4FH,5,OFFH,0,5
DB "The order must be a positive integer. Press any "
DB "key to continue.",OFFH,56H,5,"J",OFFH,56H,5,"M"
DB OFFH,79,25,0DH
O_ERROR_E DB OFFH,57H,0,5,4FH,5,0DH
I_ERROR DB OFFH,57H,0,5,4FH,5,OFFH,0,5
DB "The number of iterations must be a positive integer."
DB " Press any "
DB "key to continue.",OFFH,56H,5,"J",OFFH,56H,5,"M"
DB OFFH,79,25,0DH
ENTER_UP DB OFFH,11,5,OFFH,50H,7
DB "Enter Number of Stations: ",0DH
ENTER_NC DB OFFH,11,5,OFFH,50H,7
DB "Enter Number of Channels: ",0DH
ENTER_NN DB OFFH,14,5,OFFH,50H,7
DB "Enter Number of Nodes: ",0DH
ENTER_AMBF DB OFFH,2,5,OFFH,50H,7
DB "Maximum Backoff Delay (slots): ",0DH
ENTER_PR DB OFFH,14,5,OFFH,50H,7
DB "Propagation Ratio: ",0DH
NN_ERROR DB OFFH,57H,0,5,4FH,5,OFFH,0,5
DB "The number of nodes must be a positive integer "
DB "greater than 1 and less than 9."
DB OFFH,56H,5,"J",OFFH,56H,5,"M"

```

```

        DB 0FFH,79,25,0DH
UP_ERROR DB 0FFH,57H,0,5,4FH,5,0FFH,0,5
        DB "The number of stations must be a positive integer "
        DB "less than 21."
        DB 0FFH,56H,5,"J",0FFH,56H,5,"M"
        DB 0FFH,79,25,0DH
NC_ERROR DB 0FFH,57H,0,5,4FH,5,0FFH,0,5
        DB "The number of channels must be a positive integer "
        DB "less than 11."
        DB 0FFH,56H,5,"J",0FFH,56H,5,"M"
        DB 0FFH,79,25,0DH
LEFT_F  DB 0FFH,50H,0FH,0FFH,0,5
        DB "1 ",0FFH,0,6
        DB "2 ",0FFH,0,7
        DB "3 ",0FFH,0,8
        DB "4 ",0FFH,0,9
        DB "5 ",0FFH,0,10
        DB "6 ",0FFH,0,11
        DB "7 ",0FFH,0,12
        DB "8 "
        DB 0DH

A_FLP_S DB 0FFH,50H,0FH
        DB 0FFH,59H,0,3,20,77," □┘└|",0DH
        DB 13
        DB 2
        DB 1
        DW 0
        DW 5
    A_N1 DW 99
        DB ?
        DB ?
        DB 0FFH
        DW ?
        DB 0FFH,50H,0FH
        DB 7 DUP (?)
A_FLP_NUMBER DW 99
            DW OFFSET A_FLP_BUFFER
            DW 0
            DB 0
            DW ?
            DW ?
            DB 3 DUP (?)
            DB 70H
            DW F1_KEY
            DB 0FFH,0,3,0FFH,50H,70H,"FINISH",0DH
            DB 1,0,3,5,3,0
            C1 DW 99 DUP (?)
A_FLP_BUFFER DB 9400 DUP (?)
A_FLP_S2  DB 0FFH,50H,0FH
        DB 0FFH,59H,0,4,8,77," □┘└|",0DH
        DB 8
        DB 1
        DB 1
        DW 1

```

```

        DW 8
A_N2 DW 64
        DB ?
        DB ?
        DB 0FFH
        DW ?
        DB 0FFH,50H,0FH
        DB 7 DUP (?)
A_FLP_S3 DB 0FFH,50H,0FH
        DB 0FFH,59H,0,4,8,77,"□┘└|",0DH
        DB 8
        DB 1
        DB 1
        DW 1
        DW 8
        DW 64
        DB ?
        DB ?
        DB 0FFH
        DW ?
        DB 0FFH,50H,0FH
        DB 7 DUP (?)
A_FLP_NUMBER2 DW 64
                DW OFFSET A_FLP_BUFFER2
                DW 0
                DB 0
                DW ?
                DW ?
                DB 3 DUP (?)
                DB 70H
                DW F1_KEY
                DB 0FFH,0,4,0FFH,50H,70H,"FINISH",0DH
                DB 1,0,4,5,4,0
                C22 DW 64 DUP (?)
A_FLP_BUFFER2 DB 6580 DUP (?)
A_FLP_NUMBER3 DW 8
                DW OFFSET A_FLP_BUFFER3
                DW 0
                DB 0
                DW ?
                DW ?
                DB 3 DUP (?)
                DB 70H
                DW F1_KEY
                DB 0FFH,1DH,5,0FFH,50H,70H,"FINISH",0DH
                DB 1,1DH,5,22H,5,0
                C23 DW 64 DUP (?)
A_FLP_BUFFER3 DB 720 DUP (?)

A_FDP_NUMBER DW 4
                DW OFFSET A_FDP_BUFFER
                DW 0
                DB 3
                DW OFFSET FUNCTION_KEY_HANDLERS
                DW OFFSET MOUSE_HANDLERS

```

```

DB 0
DB 70H
DW F1_KEY
DB OFFH,30,5,OFFH,50H,70H,"FINISH",ODH
DB 1,35,7,40,7,0
DC1 DW 4 DUP (?)
A_FDP_BUFFER DB 150 DUP (?)

A_FLP_BOUND DB OFFH,59H,17H,7,8,15,"┐┘└┘|"
DB OFFH,22,8,"1",26,"2 "
DB OFFH,22,9,"2",26,"3 "
DB OFFH,22,10,"3",26,"4 "
DB OFFH,22,11,"4",26,"5 "
DB OFFH,22,12,"5",26,"6 "
DB OFFH,22,13,"6",26,"7 "
DB OFFH,22,14,"7",26,"8 "
DB OFFH,22,15,"8",26,"1 "
DB ODH
A_FLP_NUMBER4 DW 16
DW OFFSET A_FLP_BUFFER4
DW 0
DB 0
DW ?
DW ?
DB 3 DUP (?)
DB 70H
DW F1_KEY
DB OFFH,22H,5,OFFH,50H,70H,"FINISH",ODH
DB 1,22H,5,27H,5,0
C24 DW 64 DUP (?)
A_FLP_BUFFER4 DB 1440 DUP (?)
A_FLP_BOUND2 DB OFFH,59H,12,7,8,15,"┐┘└┘|"
DB OFFH,11,8,"1",26,"2 "
DB OFFH,11,9,"2",26,"3 "
DB OFFH,11,10,"3",26,"4 "
DB OFFH,11,11,"4",26,"5 "
DB OFFH,11,12,"5",26,"6 "
DB OFFH,11,13,"6",26,"7 "
DB OFFH,11,14,"7",26,"8 "
DB OFFH,11,15,"8",26,"1 ",ODH
A_FLP_BOUND3 DB OFFH,59H,44,7,8,15,"┐┘└┘|"
DB OFFH,43,8,"1",27,"2 "
DB OFFH,43,9,"2",27,"3 "
DB OFFH,43,10,"3",27,"4 "
DB OFFH,43,11,"4",27,"5 "
DB OFFH,43,12,"5",27,"6 "
DB OFFH,43,13,"6",27,"7 "
DB OFFH,43,14,"7",27,"8 "
DB OFFH,43,15,"8",27,"1 "
DB ODH
A_FLP_BOUND4 DB OFFH,59H,12,7,8,15,"┐┘└┘|"
DB OFFH,11,8,"2",26,"1 "
DB OFFH,11,9,"3",26,"1 "
DB OFFH,11,10,"4",26,"1 "
DB OFFH,11,11,"5",26,"1 "

```

```

        DB OFFH,11,12,"6",26,"1 "
        DB OFFH,11,13,"7",26,"1 "
        DB OFFH,11,14,"8",26,"1 "
        DB ODH
A_FLP_BOUNDS DB OFFH,59H,44,7,8,15,"┐┘┌└|"
        DB OFFH,43,8,"1",26,"2 "
        DB OFFH,43,9,"1",26,"3 "
        DB OFFH,43,10,"1",26,"4 "
        DB OFFH,43,11,"1",26,"5 "
        DB OFFH,43,12,"1",26,"6 "
        DB OFFH,43,13,"1",26,"7 "
        DB OFFH,43,14,"1",26,"8 "
        DB ODH

```

```

LSD DB 1
    DW CONST
    DW OFFSET SBU
    DW ?
    DW ?
    DB 1
    DW OFFSET LSD1

```

```

LSD1 DW OFFSET A3
    DW CONST
    DB 1
    DB ?
    DW ?
    DW OFFSET A4
    DW CONST
    DW OFFSET FCB1
    DW CONST
    DW OFFSET FCB2
    DW CONST
    DW OFFSET DISBU
    DW CONST

```

```

A3 DB "A:\COMMAND.COM",0
A4 DB 1," ",ODH
FCB1 DB 30H DUP (?)
FCB2 DB 30H DUP (?)
DISBU DB 200 DUP (?)

```

```

SBU DB 4000 DUP (?)
TY_BEEP1 DB OFFH,56H,5,"K",OFFH,56H,5,"L",ODH
CONST ENDS

```

```

EXTRN INITIALIZE_FUNCTION_CONTROL:NEAR,FUNCTION_CONTROL:NEAR
EXTRN GET_KEYBOCOD2:NEAR,MONITOR_TYPE:NEAR
EXTRN CLEAR_BUFFER GET_KEY:NEAR,CLEAR_BUFFER:NEAR
EXTRN RESET_MONITOR_TYPE:NEAR,CLS:NEAR,CURSER_POSITION:NEAR
EXTRN CURRENT_CURSER:NEAR,SQUARE_INDEX:NEAR
EXTRN DOUBLE_DISPLAY:NEAR,DOUBLE_DIS_LINE:NEAR
EXTRN SAVE_SCREEN:NEAR,RECOVER_SCREEN:NEAR,A_FLP_EDITOR:NEAR
EXTRN DOUBLE_DIS_LINE_OFF:NEAR

```

EXTRN SB\_MOUSE\_INPUT:NEAR,MOUSE\_SETUP:NEAR,MOUSE\_END:NEAR  
EXTRN MOUSE\_FUN\_I:NEAR,SET\_MOUSE\_CONFIG:NEAR

EXTRN LINE\_EDITOR:NEAR,GET\_MOUSE\_STATE:NEAR  
EXTRN FLP\_NUMBER\_EDITOR:NEAR,FDP\_NUMBER\_EDITOR:NEAR  
EXTRN LFLP\_ASC:NEAR,ULFLP\_ASC:NEAR,LFDP\_ASC:NEAR,ULFDP\_ASC:NEAR  
EXTRN LAF\_ASC:NEAR,ULAF\_ASC:NEAR,A\_FDP\_EDITOR:NEAR,LAD\_ASC:NEAR  
EXTRN ULAD\_ASC:NEAR,ERASE\_AD\_LAST:NEAR,CLEAR\_BUFFER\_GET\_KEY2:NEAR  
EXTRN KEY\_FUN\_I:NEAR

EXTRN OPEN\_FILE\_READ:NEAR,CREATE\_FILE:NEAR  
EXTRN SET\_RETURN\_ADDRESS:NEAR,SET\_CRITICAL\_ADDRESS:NEAR  
EXTRN READ\_FROM\_FILE:NEAR,CLOSE\_FILE:NEAR,WRITE\_TO\_FILE:NEAR  
EXTRN GENERATE\_PARITY:NEAR,RELEASE\_MEMORY:NEAR,LOAD\_PROGRAM:NEAR  
EXTRN GET\_PSP\_SEGMENT:NEAR

EXTRN RUN\_SIMULATE:NEAR,CI\_S1:NEAR,RE\_S1:NEAR,POSTRUN\_LINK:NEAR

CODE SEGMENT 'CODE' PUBLIC  
ASSUME CS:CODE,DS:CONST,ES:NOTHING,SS:STACK  
  
PUBLIC NUMERIC\_NUM\_LOCK,NUMERIC\_NUM\_NOT\_LOCK

LINKNET PROC FAR  
PUSH DS  
XOR AX,AX  
PUSH AX  
MOV SI,2CH  
MOV SI,DS:[SI]  
PUSH SI  
CALL RELEASE\_MEMORY  
POP SI  
MOV AX,CONST  
MOV DS,AX  
MOV ES,AX  
MOV ENVIRON,SI  
CLD  
MOV STACK\_POINTER,SP  
CALL MONITOR\_TYPE  
MOV AL,1  
CALL SET\_MOUSE\_CONFIG  
CALL MOUSE\_SETUP  
CALL COEF\_EDIT\_SETUP  
CALL CLS  
MOV SI,OFFSET LOGO  
CALL DOUBLE\_DIS\_LINE  
CALL CLEAR\_BUFFER\_GET\_KEY  
GET\_KEY1:  
CALL CLS  
MOV SI,OFFSET ROOT\_CONTROL  
CALL INITIALIZE\_FUNCTION\_CONTROL  
GET\_KEY:  
MOV SI,OFFSET KEYBOARD\_CONTROL\_BUFFER  
CALL GET\_KEYBOCOD2

```

JAE GET_KEY
MOV SI,OFFSET ROOT_CONTROL
CALL FUNCTION_CONTROL
JMP GET_KEY
RETURN_DOS:
CALL MOUSE_END
CALL RESET_MONITOR_TYPE
CALL CLS
MOV DX,0
CALL CURSER_POSITION
RET
LINKNET ENDP

COEF_EDIT_SETUP PROC NEAR
MOV BX,OFFSET C1
MOV DI,OFFSET A_FLP_BUFFER
MOV CX,99
MOV DX,40AH
C2:
MOV [BX],DI
PUSH CX
MOV CX,93
MOV SI,OFFSET FLP_NUMBER
REP MOVSB
MOV ES:[DI-91],DI
MOV ES:[DI-93],DI
SUB WORD PTR ES:[DI-91],31
SUB WORD PTR ES:[DI-93],67
MOV ES:[DI-60],DX
MOV ES:[DI-64],DX
MOV ES:[DI-62],DX
SUB WORD PTR ES:[DI-62],7
MOV ES:[DI-24],DX
MOV ES:[DI-26],DX
MOV ES:[DI-28],DX
ADD WORD PTR ES:[DI-24],4
ADD WORD PTR ES:[DI-26],2
ADD WORD PTR ES:[DI-28],4
MOV ES:[DI-84],DX
INC WORD PTR ES:[DI-84]
CMP DL,70
JC CE9
MOV DL,0AH
INC DH
JMP CE10
CE9:
ADD DL,15
CE10:
INC BX
INC BX
POP CX
LOOP C2

MOV SI,OFFSET A_FLP_NUMBER
MOV DI,OFFSET FIR_CF

```

```

CALL ULAF_ASC
MOV DI,OFFSET FIR_CF+1403
CALL ULAF_ASC

MOV SI,OFFSET FLP_NUMBER
MOV DI,OFFSET AMPLITUDE_SIN
MOV CX,NUMBER_OF_DATA
C3:
CALL ULFLP_ASC
ADD DI,15
LOOP C3

MOV SI,OFFSET INTEGERN
MOV DI,OFFSET ORDER_BW_CF
MOV CX,NUMBER_OF_INTEGER
C4:
CALL ULFDP_ASC
ADD DI,5
LOOP C4

MOV SI,OFFSET NB_ITERA
MOV DI,OFFSET NUMBER_ITERATION
CALL ULFDP_ASC

MOV SI,OFFSET INTEGERN
MOV DI,OFFSET S_NN
MOV CX,NUMBER_OF_INTEGER2
C5:
CALL ULFDP_ASC
ADD DI,5
LOOP C5
MOV DI,OFFSET S_NN
MOV BYTE PTR [DI+1],"2"

MOV SI,OFFSET INTEGERN1
MOV DI,OFFSET MA_AMBF
MOV CX,3
C765:
CALL ULFDP_ASC
ADD DI,10
LOOP C765

MOV BX,OFFSET C22
MOV DI,OFFSET A_FLP_BUFFER2
MOV CX,64
MOV DX,508H
CE21:
MOV [BX],DI
PUSH CX
MOV CX,89
MOV SI,OFFSET FLP_NUMBER2
REP MOVSB
MOV ES:[DI-87],DI
MOV ES:[DI-89],DI
SUB WORD PTR ES:[DI-87],30

```

```

SUB WORD PTR ES:[DI-89],63
MOV ES:[DI-56],DX
MOV ES:[DI-60],DX
MOV ES:[DI-58],DX
SUB WORD PTR ES:[DI-58],4
MOV ES:[DI-23],DX
MOV ES:[DI-25],DX
MOV ES:[DI-27],DX
ADD WORD PTR ES:[DI-23],3
ADD WORD PTR ES:[DI-25],2
ADD WORD PTR ES:[DI-27],3
MOV ES:[DI-80],DX
INC WORD PTR ES:[DI-80]
CMP DL,68
JC CE22
MOV DL,8
INC DH
JMP CE11
CE22:
ADD DL,9
CE11:
INC BX
INC BX
POP CX
LOOP CE21

```

```

MOV SI,OFFSET A_FLP_NUMBER2
MOV DI,OFFSET ITM_S
CALL ULAF_ASC
MOV SI,OFFSET A_FLP_NUMBER2
MOV DI,OFFSET ITM_MA
CALL ULAF_ASC

```

```

MOV BX,OFFSET C23
MOV DI,OFFSET A_FLP_BUFFER3
MOV CX,8
MOV DX,820H
CA21:
MOV [BX],DI
PUSH CX
MOV CX,89
MOV SI,OFFSET FLP_NUMBER2
REP MOVSB
MOV ES:[DI-87],DI
MOV ES:[DI-89],DI
SUB WORD PTR ES:[DI-87],30
SUB WORD PTR ES:[DI-89],63
MOV ES:[DI-56],DX
MOV ES:[DI-60],DX
MOV ES:[DI-58],DX
SUB WORD PTR ES:[DI-58],4
MOV ES:[DI-23],DX
MOV ES:[DI-25],DX
MOV ES:[DI-27],DX
ADD WORD PTR ES:[DI-23],3

```

```

ADD WORD PTR ES:[DI-25],2
ADD WORD PTR ES:[DI-27],3
MOV ES:[DI-80],DX
INC WORD PTR ES:[DI-80]
INC DH
INC BX
INC BX
POP CX
LOOP CA21

```

```

MOV SI,OFFSET A_FLP_NUMBER3
MOV DI,OFFSET LC_UL
CALL ULAF_ASC

```

```

MOV BX,OFFSET C24
MOV DI,OFFSET A_FLP_BUFFER4
MOV CX,16
MOV DX,815H

```

CB21:

```

MOV [BX],DI
PUSH CX
MOV CX,89
MOV SI,OFFSET FLP_NUMBER2
REP MOVSB
MOV ES:[DI-87],DI
MOV ES:[DI-89],DI
SUB WORD PTR ES:[DI-87],30
SUB WORD PTR ES:[DI-89],63
MOV ES:[DI-56],DX
MOV ES:[DI-60],DX
MOV ES:[DI-58],DX
SUB WORD PTR ES:[DI-58],4
MOV ES:[DI-23],DX
MOV ES:[DI-25],DX
MOV ES:[DI-27],DX
ADD WORD PTR ES:[DI-23],3
ADD WORD PTR ES:[DI-25],2
ADD WORD PTR ES:[DI-27],3
MOV ES:[DI-80],DX
INC WORD PTR ES:[DI-80]
CMP DL,30
JC CB22
MOV DL,15H
INC DH
JMP CB11

```

CB22:

```
ADD DL,20H
```

CB11:

```

INC BX
INC BX
POP CX

```

LOOP CB21

```

MOV SI,OFFSET A_FLP_NUMBER4
MOV DI,OFFSET LC_BL

```

```
CALL ULAF_ASC
MOV SI,OFFSET A_FLP_NUMBER4
MOV DI,OFFSET LC_S
CALL ULAF_ASC
```

```
MOV BX,OFFSET DC1
MOV DI,OFFSET A_FDP_BUFFER
MOV CX,4
```

CD1:

```
MOV [BX],DI
PUSH CX
MOV SI,OFFSET INTEGERN
MOV CX,31
REP MOVSB
POP CX
INC BX
INC BX
LOOP CD1
MOV DX,0C33H
MOV DI,OFFSET A_FDP_BUFFER
CALL SET_CUR
ADD DI,31
MOV DH,14
CALL SET_CUR
ADD DI,31
MOV DH,14H
CALL SET_CUR
ADD DI,31
MOV DH,16H
CALL SET_CUR
MOV SI,OFFSET A_FDP_NUMBER
MOV DI,OFFSET PB_NM
CALL ULAD_ASC
MOV BYTE PTR [DI+4],"0"
MOV BYTE PTR [DI+9],"0"
MOV BYTE PTR [DI+14],"0"
MOV BYTE PTR [DI+19],"0"
RET
```

SET\_CUR:

```
MOV [DI+3],DX
ADD BYTE PTR [DI+3],2
MOV [DI+5],DX
MOV [DI+7],DX
ADD BYTE PTR [DI+7],2
MOV [DI+22],DX
RET
```

COEF\_EDIT\_SETUP ENDP

MOUSE\_HANDLER PROC NEAR

```
TEST CH,1
JZ TY12
TEST BL,3
JZ TY12
```

```

MOV SI,OFFSET ROOT_CONTROL
CALL SB_MOUSE_INPUT
MOV SP,STACK_POINTER
JMP GET_KEY
TY12:
RET
MOUSE_HANDLER ENDP

QUIT_HANDLER PROC NEAR
MOV SI,OFFSET ROOT_STACK1
MOV BYTE PTR [SI],1
MOV WORD PTR [SI+3],OFFSET ROOT_STACK1+5
MOV SI,OFFSET ROOT_STACK_MESS1
MOV BYTE PTR [SI+15],0DH
MOV SP,STACK_POINTER
MOV AX,OFFSET GET_KEY1
JMP AX
QUIT_HANDLER ENDP

EXIT_HANDLER PROC NEAR
MOV AX,CONST
MOV DS,AX
MOV SP,STACK_POINTER
MOV AX,OFFSET RETURN_DOS
JMP AX
RET
EXIT_HANDLER ENDP

DEMODULATOR_SET PROC NEAR
MOV DEMOD_CON,AH
CALL DISPLAY_DEMOD
RET
DEMODULATOR_SET ENDP

RUN_S1 PROC NEAR
PUSH AX
MOV SI,OFFSET S_INTEGE
CALL DOUBLE_DIS_LINE
CF25:
MOV SI,OFFSET S_NI
CALL DOUBLE_DIS_LINE
MOV DI,OFFSET NUMBER_ITERATION
MOV SI,OFFSET NB_ITERA
CALL LFDP_ASC
CALL FDP_NUMBER_EDITOR
MOV AX,OFFSET I_ERROR
CALL ERROR_DETECT
JC CF25
CALL ULFDP_ASC
POP AX
CALL RUN_SIMULATE
RET
RUN_S1 ENDP

ENCODER_SET PROC NEAR

```

```

MOV ECD_CON,AH
CALL ED_CONSISTENT
CALL DISPLAY_ECD
CALL DISPLAY_DCD
CALL DISABLE_ED
RET
ENCODER_SET ENDP

DECODER_SET PROC NEAR
MOV DCD_CON,AH
CALL DE_CONSISTENT
CALL DISPLAY_ECD
CALL DISPLAY_DCD
CALL DISABLE_ED
RET
DECODER_SET ENDP

ED_CONSISTENT PROC NEAR
CMP ECD_CON,1
JZ ESDP1
CMP ECD_CON,2
JZ ESDP2
CMP ECD_CON,3
JZ ESDP2
CMP ECD_CON,5
JZ ESDP3
RET
ESDP1:
MOV DCD_CON,1
RET
ESDP2:
CMP DCD_CON,2
JNZ ESDP4
RET
ESDP4:
CMP DCD_CON,3
JNZ ESDP5
RET
ESDP5:
MOV DCD_CON,0
RET
ESDP3:
MOV DCD_CON,4
RET
ED_CONSISTENT ENDP

DE_CONSISTENT PROC NEAR
CMP DCD_CON,1
JZ DSEP1
CMP DCD_CON,2
JZ DSEP2
CMP DCD_CON,3
JZ DSEP2
CMP DCD_CON,4
JZ DSEP3

```

```

    RET
DSEP1:
    MOV ECD_CON,1
    RET
DSEP2:
    CMP ECD_CON,2
    JNZ DSEP4
    RET
DSEP4:
    CMP ECD_CON,3
    JNZ DSEP5
    RET
DSEP5:
    MOV ECD_CON,0
    RET
DSEP3:
    CMP ECD_CON,5
    JNZ DSEP5
    RET
DE_CONSISTENT ENDP

R_CONFIG_S1 PROC NEAR
    MOV STACK_POINTER1,SP
    CALL FILE_PREPARE
    MOV DX,OFFSET A10
    CALL OPEN_FILE_READ
    MOV FILE_HANDLE,AX
    MOV CX,N_B+2
    MOV BX,AX
    MOV DX,OFFSET SBU
    CALL READ_FROM_FILE
    MOV BX,FILE_HANDLE
    CALL CLOSE_FILE

    MOV SI,OFFSET SBU
    MOV CX,N_B
    CALL GENERATE_PARITY
    CMP [SI],AX
    JZ RE12
    MOV SI,OFFSET IL_COF
    CALL DOUBLE_DIS_LINE
    CALL CLEAR_BUFFER_GET_KEY
    CALL GET_MOUSE_STATE
    RET
RE12:
    MOV SI,OFFSET SBU
    MOV DI,OFFSET LINK_NET
    MOV CX,N_B
    REP MOVSB
    CALL DISPLAY_CON
    CALL DISABLE_ED
    RET
R_CONFIG_S1 ENDP

FILE_PREPARE PROC NEAR

```

```

MOV SI,OFFSET ENTER_FILE
CALL DOUBLE_DIS_LINE
MOV SI,OFFSET FILE_NAME
CALL LINE_EDITOR
PUSH DS
MOV AX,CODE
MOV DS,AX
MOV SI,OFFSET ERROR_HANDLER
CALL SET_RETURN_ADDRESS
CALL SET_CRITICAL_ADDRESS
POP DS
MOV SI,OFFSET A10-1
MOV AL,0DH

```

R12:

```

INC SI
CMP [SI],AL
JNZ R12
MOV BYTE PTR [SI],0
RET

```

FILE\_PREPARE ENDP

S\_CONFIG\_S1 PROC NEAR

```

MOV STACK_POINTER1,SP
CALL FILE_PREPARE
MOV DX,OFFSET A10
CALL CREATE_FILE
MOV FILE_HANDLE,AX

```

```

MOV SI,OFFSET LINK_NET
MOV DI,OFFSET SBU
MOV CX,N_B
REP MOVSB
MOV SI,OFFSET SBU
MOV CX,N_B
CALL GENERATE_PARITY
MOV [SI],AX

```

```

MOV CX,N_B+2
MOV BX,FILE_HANDLE
MOV DX,OFFSET SBU
CALL WRITE_TO_FILE
MOV BX,FILE_HANDLE
CALL CLOSE_FILE
RET

```

S\_CONFIG\_S1 ENDP

R\_NETCON\_S1 PROC NEAR

```

MOV STACK_POINTER1,SP
CALL FILE_PREPARE
MOV DX,OFFSET A10
CALL OPEN_FILE_READ
MOV FILE_HANDLE,AX
MOV CX,NC_N_B+2
MOV BX,AX
MOV DX,OFFSET SBU

```

```
CALL READ_FROM_FILE
MOV BX,FILE_HANDLE
CALL CLOSE_FILE
```

```
MOV SI,OFFSET SBU
MOV CX,NC_N_B
CALL GENERATE_PARITY
CMP [SI],AX
JZ RE35
MOV SI,OFFSET IL_COF
CALL DOUBLE_DIS_LINE
CALL CLEAR_BUFFER_GET_KEY
CALL GET_MOUSE_STATE
RET
```

RE35:

```
MOV SI,OFFSET SBU
MOV DI,OFFSET NET_CON
MOV CX,NC_N_B
REP MOVSB
CALL DISPLAY_NETCON
RET
```

R\_NETCON\_S1 ENDP

S\_NETCON\_S1 PROC NEAR

```
MOV STACK_POINTER1,SP
CALL FILE_PREPARE
MOV DX,OFFSET A10
CALL CREATE_FILE
MOV FILE_HANDLE,AX
```

```
MOV SI,OFFSET NET_CON
MOV DI,OFFSET SBU
MOV CX,NC_N_B
REP MOVSB
MOV SI,OFFSET SBU
MOV CX,NC_N_B
CALL GENERATE_PARITY
MOV [SI],AX
```

```
MOV CX,NC_N_B+2
MOV BX,FILE_HANDLE
MOV DX,OFFSET SBU
CALL WRITE_TO_FILE
MOV BX,FILE_HANDLE
CALL CLOSE_FILE
RET
```

S\_NETCON\_S1 ENDP

SOURCE\_SET PROC NEAR

```
PUSH AX
AND AH,OFH
MOV SOU_CON,AH
PUSH AX
CALL DISPLAY_SOU
```

```
MOV SI,OFFSET S_P_NUMB
CALL DOUBLE_DIS_LINE
POP AX
CMP AH,2
JAE S13
POP AX
MOV SI,OFFSET ENTER_BITR
MOV DI,OFFSET BIT_RATE
JMP S12
```

```
S13:
  CMP AH,4
  POP AX
  JC S14
  MOV SI,OFFSET ENTER_NO
  MOV DI,OFFSET SOU_NO
  JMP S12
```

```
S14:
  TEST AH,10H
  JZ S11
  MOV SI,OFFSET ENTER_AMPL
  AND AH,0FH
  CMP AH,2
  JNZ S33
  MOV DI,OFFSET AMPLITUDE_SIN
  JMP S12
```

```
S33:
  MOV DI,OFFSET AMPLITUDE_SW
  JMP S12
```

```
S11:
  MOV SI,OFFSET ENTER_FREQ
  AND AH,0FH
  CMP AH,2
  JNZ S34
  MOV DI,OFFSET FREQUENCY_SIN
  JMP S12
```

```
S34:
  MOV DI,OFFSET FREQUENCY_SW
```

```
S12:
  CALL DOUBLE_DIS_LINE
  CALL ENTER_NUMBER
  RET
```

ENTER\_NUMBER:

```
MOV SI,OFFSET FLP_NUMBER
PUSH DI
CALL LFLP_ASC
AND BYTE PTR [SI+12],0FEH
MOV DI,[SI]
AND BYTE PTR [DI+13],0CFH
XOR BYTE PTR [DI+13],20H
CALL FLP_NUMBER_EDITOR
POP DI
CALL ULFLP_ASC
RET
```

SOURCE\_SET ENDP

```
ENTER_NUMBER1 PROC NEAR
    MOV SI,OFFSET FLP_NUMBER
    PUSH DI
    CALL LFLP_ASC
    AND BYTE PTR [SI+12],0FEH
    MOV DI,[SI]
    AND BYTE PTR [DI+13],0CFH
    CALL FLP_NUMBER_EDITOR
    POP DI
    CALL ULFLP_ASC
    RET
```

ENTER\_NUMBER1 ENDP

```
ORDER_SET PROC NEAR
    MOV SI,OFFSET CF_BW_T
    TEST AH,80H
    JZ OSP1
    ADD SI,4
OSP1:
    MOV BL,1
    TEST AH,40H
    JZ OSP2
    MOV BL,2
OSP2:
    PUSH AX
    AND AH,0FH
    MOV AL,AH
    SUB AL,3
    XOR AH,AH
    ADD SI,AX
    MOV BYTE PTR [SI],BL
    POP AX
    TEST AH,80H
    JZ OSP3
    CALL DISPLAY_RFT
    RET
OSP3:
    CALL DISPLAY_CFT
    RET
```

ORDER\_SET ENDP

```
L_COUNT_BYTE EQU 18
L_DATA_START EQU 27
L_SECOND_DATA EQU 28
```

```
FILTER_SET PROC NEAR
    MOV BH,AH
    MOV BL,AH
    AND AH,07H
    TEST BH,80H
    JNZ FS1
    CALL CHANNEL_FILTER_SET
    RET
```

```

FS1:
    CALL RECEIVER_FILTER_SET
    RET

CHANNEL_FILTER_SET:
    PUSH BX
    MOV CF_CON, AH
    CALL DISPLAY_CF
RF3:
    POP BX
    MOV AH, BH
    AND AH, 0FH
    CMP AH, 1
    JNZ CFOA9
    RET
CFOA9:
    CMP AH, 7
    JNZ CFOA10
    RET
CFOA10:
    TEST BH, 40H
    JNZ CFOA1
    CALL CF_ORDER_SET
    RET
CFOA1:
    CALL CF_ATTENUATION_SET
    RET

CF_ORDER_SET:
    PUSH BX
    TEST BH, 30H
    JNZ CF1
    MOV SI, OFFSET S_INTEGE
    CALL DOUBLE_DIS_LINE
    POP BX
    MOV DI, OFFSET ORDER_BW_CF
    CALL CF_DIAD2
CF21:
    TEST BH, 80H
    JZ CF31
    CMP RF_CON, 6
    JNZ CF32
    JMP CF33
CF31:
    CMP CF_CON, 6
    JNZ CF32
CF33:
    MOV SI, OFFSET E_ORDER_ODD
    JMP CF34
CF32:
    MOV SI, OFFSET E_ORDER
CF34:
    CALL DOUBLE_DIS_LINE
    MOV SI, OFFSET INTEGERN
    CALL LFDP_ASC

```

```

CALL FDP_NUMBER_EDITOR
MOV AX,OFFSET O_ERROR
CALL ERROR_DETECT
JC CF21
CALL ULFDP_ASC
RET
CF1:
MOV SI,OFFSET S_P_NUMB
CALL DOUBLE_DIS_LINE
POP BX
MOV DI,OFFSET SF_BW_CF
CALL CF_DIAD
PUSH BX
AND BH,30H
CMP BH,10H
JZ CF2
CMP BH,20H
JZ CF10
POP BX
MOV AH,BH
AND AH,0FH
CMP AH,6
JNZ CF10
CALL ENTER_COEF
RET
CF10:
POP BX
MOV SI,OFFSET E_CUTOFF
ADD DI,15
JMP CF3
CF2:
POP BX
MOV SI,OFFSET E_SAMPLE
CF3:
CALL DOUBLE_DIS_LINE
CALL ENTER_NUMBER
RET

ERROR_DETECT:
CMP BYTE PTR [SI+L_COUNT_BYTE],1
JNZ CF22
CALL ORDER_ERROR
JMP CF23
CF22:
CMP BYTE PTR [SI+L_COUNT_BYTE],2
JNZ CF24
CMP BYTE PTR [SI+L_DATA_START+1],"0"
JNZ CF24
CALL ORDER_ERROR
CF23:
STC
RET
CF24:
CLC
RET

```

```
ORDER_ERROR:
    PUSH SI
    MOV SI,AX
    CALL DOUBLE_DIS_LINE
    CALL CLEAR_BUFFER_GET_KEY
    MOV SI,OFFSET O_ERROR_E
    CALL DOUBLE_DIS_LINE_OFF
    POP SI
    RET
```

```
CF_ATTENUATION_SET:
    PUSH BX
    MOV SI,OFFSET S_P_NUMB
    CALL DOUBLE_DIS_LINE
    POP BX
    MOV DI,OFFSET SF_BW_CF+30
    CALL CF_DIAD
    PUSH BX
    TEST BH,30H
    JNZ CF5
    MOV SI,OFFSET E_SAMPLE
    JMP CF6
```

```
CF5:
    TEST BH,10H
    JNZ CF7
    MOV SI,OFFSET E_PA
    ADD DI,45
    JMP CF6
```

```
CF7:
    TEST BH,20H
    JNZ CF8
    TEST BL,8
    JZ CF9
    MOV SI,OFFSET E_STOEDG
    ADD DI,30
    JMP CF6
```

```
CF9:
    MOV SI,OFFSET E_PASEDG
    ADD DI,15
    JMP CF6
```

```
CF8:
    MOV SI,OFFSET E_SA
    ADD DI,60
```

```
CF6:
    CALL DOUBLE_DIS_LINE
    CALL ENTER_NUMBER
    POP BX
    RET
```

```
CF_DIAD:
    TEST BH,80H
    JZ CF973
```

```
ADD DI,420
CF973:
MOV AH,BH
AND AH,07H
CMP AH,3
JZ CF974
CMP AH,5
JZ CF975
CMP AH,6
JZ CF955
ADD DI,105
JMP CF974
CF955:
ADD DI,315
JMP CF974
CF975:
ADD DI,210
CF974:
RET
```

```
CF_DIAD2:
TEST BH,80H
JZ CF976
ADD DI,25
CF976:
MOV AH,BH
AND AH,07H
CMP AH,2
JZ CF979
CMP AH,3
JZ CF977
CMP AH,5
JZ CF978
CMP AH,6
JZ CF980
ADD DI,5
JMP CF977
CF979:
ADD DI,15
JMP CF977
CF980:
ADD DI,20
JMP CF977
CF978:
ADD DI,10
CF977:
RET
```

```
RECEIVER_FILTER_SET:
PUSH BX
MOV RF_CON,AH
CALL DISPLAY_RF
```

```

MOV AX,OFFSET RF3
JMP AX
RET
FILTER_SET ENDP

```

```

SQ_INDEX_NR      EQU 7
SQ_INDEX_NC      EQU 8
SQ_INDEX_TNE     EQU 23
SQ_INDEX_NEPR    EQU 21
SQ_INDEX_NEPWR   EQU 25

```

```

ENTER_COEF PROC NEAR
MOV SI,OFFSET FI_CF_O
MOV DI,OFFSET FIR_CF
TEST BH,80H
JZ EC1
MOV SI,OFFSET FI_RF_O
ADD DI,1403

```

```

EC1:
CALL COMPUTE_NUMBER

```

```

PUSH DI
PUSH AX
MOV SI,OFFSET A_FLP_S
MOV DS:[SI+SQ_INDEX_TNE],AX
MOV DL,5
DIV DL
CMP AH,0
JZ G1
INC AL

```

```

G1:
MOV DS:[SI+SQ_INDEX_NR],AL

```

```

MOV SI,OFFSET SBU
CALL SAVE_SCREEN
CALL CLS
MOV SI,OFFSET A_FLP_S
CALL SQUARE_INDEX
MOV SI,OFFSET A_NUMBER
CALL DOUBLE_DIS_LINE
POP AX
POP DI
MOV SI,OFFSET A_FLP_NUMBER
CALL LAF_ASC
MOV [SI],AX
CALL A_FLP_EDITOR
CALL ULAF_ASC
MOV SI,OFFSET SBU
CALL RECOVER_SCREEN
RET

```

```

COMPUTE_NUMBER:
INC SI
MOV DL,10
XOR CH,CH

```

```

    XOR AX,AX
CN1:
    CMP BYTE PTR DS:[SI],0DH
    JZ CN2
    MUL DL
    MOV CL,DS:[SI]
    SUB CL,30H
    ADD AX,CX
    INC SI
    JMP CN1
CN2:
    RET
ENTER_COEF ENDP

CHANNEL_SET PROC NEAR
    MOV CHL_CON,AH
    CALL DISPLAY_CHL
    CMP CHL_CON,1
    JZ CS1
    CMP CHL_CON,4
    JZ CS1
    CMP CHL_CON,2
    JZ CS2
    MOV SI,OFFSET S_P_NUMB
    CALL DOUBLE_DIS_LINE
    MOV SI,OFFSET ENTER_ER
    MOV DI,OFFSET CHL_ER
    CALL DOUBLE_DIS_LINE
    CALL ENTER_NUMBER
    RET
CS2:
    MOV SI,OFFSET S_NUMBER
    CALL DOUBLE_DIS_LINE
    MOV SI,OFFSET ENTER_NO
    MOV DI,OFFSET CHL_NO
    CALL DOUBLE_DIS_LINE
    CALL ENTER_NUMBER1
CS1:
    RET
CHANNEL_SET ENDP

PM_DISPLAY PROC NEAR
    MOV CHL_CON,4
    CALL DISPLAY_CHL
    RET
PM_DISPLAY ENDP

PM_SET PROC NEAR
PM_CF:
    MOV SI,OFFSET CFM
    MOV DI,OFFSET PM_CONST+75
    JMP PM_SET1
PM_DS:
    MOV SI,OFFSET DBTR
    MOV DI,OFFSET PM_CONST+90

```

```

    JMP PM_SET1
PM_AT_H_T:
    MOV SI,OFFSET TAH
    MOV DI,OFFSET PM_CONST
    JMP PM_SET1
PM_AT_H_R:
    MOV SI,OFFSET RAH
    MOV DI,OFFSET PM_CONST+15
    JMP PM_SET1
PM_AT_G_T:
    MOV SI,OFFSET TAG
    MOV DI,OFFSET PM_CONST+45
    JMP PM_SET2
PM_AT_G_R:
    MOV SI,OFFSET RAG
    MOV DI,OFFSET PM_CONST+60
    JMP PM_SET2
PM_POWER:
    MOV SI,OFFSET TAP
    MOV DI,OFFSET PM_CONST+30
    JMP PM_SET1
PM_NOISE:
    MOV SI,OFFSET RNT
    MOV DI,OFFSET PM_CONST+105
    JMP PM_SET2
PM_AREA:
    AND AH,1
    AND PM_CON,0FEH
    XOR PM_CON,AH
    RET
PM_FOLIAGE:
    AND AH,2
    AND PM_CON,0FDH
    XOR PM_CON,AH
    RET
PM_BANDWIDTH:
    MOV SI,OFFSET RBW
    MOV DI,OFFSET PM_CONST+120
PM_SET1:
    PUSH DI
    PUSH SI
    MOV SI,OFFSET S_P_NUMB
    CALL DOUBLE_DIS_LINE
    POP SI
    CALL DOUBLE_DIS_LINE
    POP DI
    CALL ENTER_NUMBER
    RET

PM_SET2:
    PUSH DI
    PUSH SI
    MOV SI,OFFSET S_NUMBER
    CALL DOUBLE_DIS_LINE

```

```

    POP SI
    CALL DOUBLE_DIS_LINE
    POP DI
    CALL ENTER_NUMBER1
    RET
PM_SET ENDP

MODULATOR_SET PROC NEAR
    MOV MOD_CON, AH
    CALL DISPLAY_MOD
    RET
MODULATOR_SET ENDP

DISPLAY_CON PROC NEAR
    MOV SI, OFFSET CON_SCREEN
    CALL DOUBLE_DISPLAY
    CALL DISPLAY_SOU
    CALL DISPLAY_MOD
    CALL DISPLAY_ECD
    CALL DISPLAY_CF
    CALL DISPLAY_RF
    CALL DISPLAY_DCD
    CALL DISPLAY_DEMOD
    CALL DISPLAY_CHL
    RET
DISPLAY_CON ENDP

DISPLAY_MOD PROC NEAR
    MOV SI, OFFSET MOD_UDF
    MOV DI, OFFSET MOD_P
    MOV AL, MOD_CON
    CALL DISPLAY_ITEM
    CALL BLOCK_TEST
    RET
DISPLAY_MOD ENDP

DISPLAY_ECD PROC NEAR
    MOV SI, OFFSET ECD_UDF
    MOV DI, OFFSET ECD_P
    MOV AL, ECD_CON
    CALL DISPLAY_ITEM
    RET
DISPLAY_ECD ENDP

DISPLAY_SOU PROC NEAR
    MOV SI, OFFSET SOU_UDF
    MOV DI, OFFSET SOU_P
    MOV AL, SOU_CON
    CALL DISPLAY_ITEM
    CALL BLOCK_TEST
    RET
DISPLAY_SOU ENDP

DISPLAY_DCD PROC NEAR
    MOV SI, OFFSET DCD_UDF

```

```

MOV DI,OFFSET DCD_P
MOV AL,DCD_CON
CALL DISPLAY_ITEM
RET
DISPLAY_DCD ENDP

```

```

DISPLAY_CF PROC NEAR
MOV SI,OFFSET CF_UDF
MOV DI,OFFSET CF_P
MOV AL,CF_CON
CALL DISPLAY_ITEM
CALL DISPLAY_CFT
CALL BLOCK_TEST
RET
DISPLAY_CF ENDP

```

```

BLOCK_TEST PROC NEAR
MOV SI,OFFSET OD_S1+17
AND BYTE PTR [SI],0FEH
CMP MOD_CON,0
JZ BLOCK_PLOT4
CMP MOD_CON,1
JZ BLOCK_PLOT4
CMP MOD_CON,9
JZ BLOCK_PLOT4
CMP MOD_CON,10
JZ BLOCK_PLOT4
CMP CHL_CON,3
JZ BLOCK_PLOT4
CMP SOU_CON,2
JZ BLOCK_PLOT4
XOR BYTE PTR [SI],1

```

BLOCK\_PLOT4:

```

MOV SI,OFFSET VD_S1+5
MOV DI,OFFSET OD_S1+5
AND BYTE PTR [SI],0FEH
AND BYTE PTR [DI],0FEH
CMP SOU_CON,1
JNZ BLOCK_PLOT
XOR BYTE PTR [SI],1
XOR BYTE PTR [DI],1

```

BLOCK\_PLOT:

```

MOV DI,OFFSET OD_S1+29
AND BYTE PTR [DI],0FEH
CMP CHL_CON,3
JZ BLOCK_PLOT5
CMP CF_CON,0
JZ TEST21
CMP CF_CON,1
JZ TEST21

```

UNBLOCK1:

```

XOR BYTE PTR [DI],1
JMP BLOCK_PLOT5

```

```

TEST21:
  CMP RF_CON,0
  JZ BLOCK_PLOT5
  CMP RF_CON,1
  JNZ UNBLOCK1
BLOCK_PLOT5:

  MOV SI,OFFSET VD_S1+17
  AND BYTE PTR [SI],0FEH
  CMP CF_CON,0
  JZ BLOCK_PLOT1
  CMP CF_CON,1
  JZ BLOCK_PLOT1
  CMP CHL_CON,3
  JZ BLOCK_PLOT1
  XOR BYTE PTR [SI],1
BLOCK_PLOT1:

  MOV SI,OFFSET VD_S1+29
  AND BYTE PTR [SI],0FEH
  CMP RF_CON,0
  JZ BLOCK_PLOT2
  CMP RF_CON,1
  JZ BLOCK_PLOT2
  CMP CHL_CON,3
  JZ BLOCK_PLOT2
  XOR BYTE PTR [SI],1
BLOCK_PLOT2:
  RET
BLOCK_TEST ENDP

DISPLAY_CFT PROC NEAR
  CMP CF_CON,0
  JNZ DCFT1
  RET
DCFT1:
  CMP CF_CON,1
  JNZ DCFT2
  RET
DCFT2:
  MOV SI,OFFSET CF_BW_T
  MOV BL,CF_CON
  SUB BL,3
  XOR BH,BH
  MOV AL,[SI+BX]
  MOV SI,OFFSET CFT_UDF
  MOV DI,OFFSET CFT_P1
  CMP BL,3
  JNZ DCT1
  MOV DI,OFFSET CFT_P2
DCT1:
  CALL DISPLAY_ITEM
  RET
DISPLAY_CFT ENDP

```

```

DISPLAY_RF PROC NEAR
    MOV SI,OFFSET RF_UDF
    MOV DI,OFFSET RF_P
    MOV AL,RF_CON
    CALL DISPLAY_ITEM
    CALL DISPLAY_RFT
    CALL BLOCK_TEST
    RET
DISPLAY_RF ENDP

DISPLAY_RFT PROC NEAR
    CMP RF_CON,0
    JNZ DRFT1
    RET
DRFT1:
    CMP RF_CON,1
    JNZ DRFT2
    RET
DRFT2:
    MOV SI,OFFSET CF_BW_T+4
    MOV BL,RF_CON
    SUB BL,3
    XOR BH,BH
    MOV AL,[SI+BX]
    MOV SI,OFFSET RFT_UDF
    MOV DI,OFFSET RFT_P1
    CMP BL,3
    JNZ DRT1
    MOV DI,OFFSET RFT_P2
DRT1:
    CALL DISPLAY_ITEM
    RET
DISPLAY_RFT ENDP

DISPLAY_DEMOD PROC NEAR
    MOV SI,OFFSET DEMOD_UDF
    MOV DI,OFFSET DEMOD_P
    MOV AL,DEMOD_CON
    CALL DISPLAY_ITEM
    RET
DISPLAY_DEMOD ENDP

DISPLAY_CHL PROC NEAR
    MOV SI,OFFSET CHL_UDF
    MOV DI,OFFSET CHL_P
    MOV AL,CHL_CON
    CALL DISPLAY_ITEM
    RET
DISPLAY_CHL ENDP

DISPLAY_ITEM PROC NEAR
    PUSH DI
    PUSH AX
    CALL DOUBLE_DIS_LINE
    POP AX

```

```

    POP SI
    CMP AL,0
    JNZ CON1
    RET
CON1:
    DEC AL
    ADD AL,AL
    XOR AH,AH
    ADD SI,AX
    MOV SI,[SI]
    CALL DOUBLE_DIS_LINE
    RET
DISPLAY_ITEM ENDP

RECOVER_CON PROC NEAR
    MOV SI,OFFSET CON_SCREEN_CLEAR
    CALL DOUBLE_DIS_LINE
    RET
RECOVER_CON ENDP

SOURCE_POINTER PROC NEAR
    MOV SI,OFFSET T1
    JMP D1
RECOVER_SOURCE:
    MOV SI,OFFSET S1
    JMP D1
ENCODER_POINTER:
    MOV SI,OFFSET T2
    JMP D1
RECOVER_ENCODER:
    MOV SI,OFFSET S2
    JMP D1
MODULATOR_POINTER:
    MOV SI,OFFSET T3
    JMP D1
RECOVER_MODULATOR:
    MOV SI,OFFSET S3
    JMP D1
CHANNEL_S_POINTER:
    MOV SI,OFFSET T4
    JMP D1
RECOVER_CHANNEL_S:
    MOV SI,OFFSET S4
D1:
    CALL DOUBLE_DIS_LINE
    RET
CHANNEL_POINTER:
    MOV SI,OFFSET T5
    JMP D1
RECOVER_CHANNEL:
    MOV SI,OFFSET S5
    JMP D1
DECODER_POINTER:
    MOV SI,OFFSET T6
    JMP D1

```

```

RECOVER_DECODER:
    MOV SI,OFFSET S6
    JMP D1
DEMODULATOR_POINTER:
    MOV SI,OFFSET T7
    JMP D1
RECOVER_DEMODULATOR:
    MOV SI,OFFSET S7
    JMP D1
RECEIVER_POINTER:
    MOV SI,OFFSET T8
    JMP D1
RECOVER_RECEIVER:
    MOV SI,OFFSET S8
    JMP D1
SOURCE_POINTER ENDP

NUMERIC_NUM_LOCK PROC NEAR
    RET
NUMERIC_NUM_LOCK ENDP

NUMERIC_NUM_NOT_LOCK PROC NEAR
    RET
NUMERIC_NUM_NOT_LOCK ENDP

ERROR_HANDLER PROC NEAR
    MOV AX,CONST
    MOV DS,AX
    MOV ES,AX
    MOV SP,STACK_POINTER1
    RET
ERROR_HANDLER ENDP

TY_BEEP PROC NEAR
    TEST BL,3
    JNZ TY93
    RET
TY93:
    PUSH DX
    PUSH SI
    CALL CURRENT_CURSER
    PUSH DX
    MOV SI,OFFSET TY_BEEP1
    CALL DOUBLE_DIS_LINE
    POP DX
    CALL CURSER_POSITION
    POP SI
    POP DX
    RET
TY_BEEP ENDP

DUMMY_CALL PROC NEAR
    RET
DUMMY_CALL ENDP

```

```

FIR_SAVE1 PROC NEAR
    MOV STACK_POINTER1, SP
    PUSH AX
    CALL FILE_PREPARE
    MOV DX, OFFSET A10
    CALL CREATE_FILE
    MOV FILE_HANDLE, AX
    POP AX

    MOV DI, OFFSET SBU
    MOV SI, OFFSET ORDER_BW_CF+20
    TEST AH, 80H
    JZ CSS1
    ADD SI, 25
CSS1:
    MOV CX, 5
    REP MOVSB
    MOV SI, OFFSET SF_BW_CF+315
    TEST AH, 80H
    JZ CSS2
    ADD SI, 420
CSS2:
    MOV CX, 105
    REP MOVSB
    MOV SI, OFFSET FIR_CF
    TEST AH, 80H
    JZ CSS3
    ADD SI, 1403
CSS3:
    MOV CX, 1403
    REP MOVSB
    MOV SI, OFFSET SBU
    MOV CX, 1513
    CALL GENERATE_PARITY
    MOV [SI], AX

    MOV CX, 1515
    MOV BX, FILE_HANDLE
    MOV DX, OFFSET SBU
    CALL WRITE_TO_FILE
    MOV BX, FILE_HANDLE
    CALL CLOSE_FILE
    RET
FIR_SAVE1 ENDP

FIR_RETRIEVE1 PROC NEAR
    MOV STACK_POINTER1, SP
    PUSH AX
    CALL FILE_PREPARE
    MOV DX, OFFSET A10
    CALL OPEN_FILE_READ
    MOV FILE_HANDLE, AX
    MOV CX, 1515
    MOV BX, FILE_HANDLE
    MOV DX, OFFSET SBU

```

```

CALL READ_FROM_FILE
MOV BX,FILE_HANDLE
CALL CLOSE_FILE

MOV SI,OFFSET SBU
MOV CX,1513
CALL GENERATE_PARITY
CMP [SI],AX
POP AX
JZ CSS8
MOV SI,OFFSET IL_FIR
CALL DOUBLE_DIS_LINE
CALL CLEAR_BUFFER_GET_KEY
CALL GET_MOUSE_STATE
RET
CSS8:
MOV SI,OFFSET SBU
MOV DI,OFFSET ORDER_BW_CF+20
TEST AH,80H
JZ CSS4
ADD DI,25
CSS4:
MOV CX,5
REP MOVSB
MOV DI,OFFSET SF_BW_CF+315
TEST AH,80H
JZ CSS5
ADD DI,420
CSS5:
MOV CX,105
REP MOVSB
MOV DI,OFFSET FIR_CF
TEST AH,80H
JZ CSS6
ADD DI,1403
CSS6:
MOV CX,1403
REP MOVSB
RET
FIR_RETRIEVE1 ENDP

BW_SET PROC NEAR
TEST AH,80H
JNZ BWS1
AND AH,07H
MOV CF_CON,AH
CALL DISPLAY_CF
RET
BWS1:
AND AH,07H
MOV RF_CON,AH
CALL DISPLAY_RF
RET
BW_SET ENDP

```

```

BERH_S1 PROC NEAR
    CALL DISPLAY_CON
    MOV SI,OFFSET BER_HIS1
    CALL DOUBLE_DIS_LINE
    CALL GET_MOUSE_STATE
GET_M1:
    CALL CLEAR_BUFFER_GET_KEY2
    CMP AX,0
    JZ GET_M522
    MOV SI,OFFSET BER_HIS_KM1
    CALL KEY_FUN_I
    JC GET_M1
    JMP GET_M523
GET_M522:
    CALL GET_MOUSE_STATE
    TEST CH,1
    JZ GET_M1
    TEST BL,3
    JZ GET_M1
    MOV SI,OFFSET BER_HIS_M1
    CALL MOUSE_FUN_I
    JC GET_M1
GET_M523:
    TEST AL,10H
    JZ TY924
TY922:
    CALL CLS
    RET
TY924:
    MOV AH,AL
    CALL POSTRUN_LINK
    JMP TY922
BERH_S1 ENDP

```

```

SC_S1 PROC NEAR
    CALL DISPLAY_CON
    MOV SI,OFFSET SIG_CON3
    CALL DOUBLE_DIS_LINE
    CALL GET_MOUSE_STATE
GET_M2:
    CALL CLEAR_BUFFER_GET_KEY2
    CMP AX,0
    JZ GET_M518
    MOV SI,OFFSET SIG_CON_KM1
    CALL KEY_FUN_I
    JC GET_M2
    JMP GET_M519
GET_M518:
    CALL GET_MOUSE_STATE
    TEST CH,1
    JZ GET_M2
    TEST BL,3
    JZ GET_M2
    MOV SI,OFFSET SIG_CON_M1
    CALL MOUSE_FUN_I

```

```

JC GET_M2
GET_M519:
TEST AL,10H
JZ TY921
TY923:
CALL CLS
RET
TY921:
MOV AH,AL
CALL POSTRUN_LINK
JMP TY923
SC_S1 ENDP

```

```

FD_S1 PROC NEAR
CALL DISPLAY_CON

```

```

CMP CF_CON,0
JZ TY671
CMP CF_CON,1
JNZ TY672
TY671:
MOV SI,OFFSET SIG_CON1+2
MOV BYTE PTR [SI],0FH
MOV SI,OFFSET SIG_CON2+2
MOV BYTE PTR [SI],0FH
MOV SI,OFFSET TX1+2
MOV BYTE PTR [SI],7
MOV SI,OFFSET TX2+2
MOV BYTE PTR [SI],7
JMP TY673

```

```

TY672:
MOV SI,OFFSET SIG_CON1+2
MOV BYTE PTR [SI],70H
MOV SI,OFFSET SIG_CON2+2
MOV BYTE PTR [SI],70H
MOV SI,OFFSET TX1+2
MOV BYTE PTR [SI],0FH
MOV SI,OFFSET TX2+2
MOV BYTE PTR [SI],0FH

```

```

TY673:
CMP RF_CON,0
JZ TY674
CMP RF_CON,1
JNZ TY675

```

```

TY674:
MOV SI,OFFSET SIG_CON5+2
MOV BYTE PTR [SI],0FH
MOV SI,OFFSET SIG_CON4+2
MOV BYTE PTR [SI],0FH
MOV SI,OFFSET RX1+2
MOV BYTE PTR [SI],7
MOV SI,OFFSET RX2+2
MOV BYTE PTR [SI],7
JMP TY676

```

```

TY675:
  MOV SI,OFFSET SIG_CON5+2
  MOV BYTE PTR [SI],70H
  MOV SI,OFFSET SIG_CON4+2
  MOV BYTE PTR [SI],70H
  MOV SI,OFFSET RX1+2
  MOV BYTE PTR [SI],0FH
  MOV SI,OFFSET RX2+2
  MOV BYTE PTR [SI],0FH
TY676:

  MOV SI,OFFSET SIG_CON1
  CALL DOUBLE_DIS_LINE
  CALL GET_MOUSE_STATE
GET_M3:
  CALL CLEAR_BUFFER_GET_KEY2
  CMP AX,0
  JZ GET_M520
  MOV SI,OFFSET SIG_CON_KM2
  CALL KEY_FUN_I
  JC GET_M3
  JMP GET_M521
GET_M520:
  CALL GET_MOUSE_STATE
  TEST CH,1
  JZ GET_M3
  TEST BL,3
  JZ GET_M3
  MOV SI,OFFSET SIG_CON_M2
  CALL MOUSE_FUN_I
  JC GET_M3
GET_M521:
  TEST AL,10H
  JZ TY971
TY973:
  CALL CLS
  RET
TY971:
  MOV AH,AL

  CMP AH,2
  JZ TY677
  CMP AH,3
  JNZ TY678
TY677:
  CMP CF_CON,0
  JZ GET_M3
  CMP CF_CON,1
  JZ GET_M3
  JMP TY679
TY678:
  CMP RF_CON,0
  JZ GET_M3
  CMP RF_CON,1
  JZ GET_M3

```

```
TY679:
  CALL POSTRUN_LINK
  JMP TY973
FD_S1 ENDP
```

```
DOS_HANDLER PROC NEAR
  MOV STACK_POINTER1,SP
  MOV AX, CODE
  MOV DS, AX
  MOV SI, OFFSET ERROR_RETURN
  CALL SET_RETURN_ADDRESS
  CALL SET_CRITICAL_ADDRESS
  CALL GET_PSP_SEGMENT
  MOV ES, AX
  MOV SI, 2CH
  MOV AX, ES:[SI]
  MOV BX, CONST
  MOV DS, BX
  MOV ES, BX
  MOV SI, OFFSET LSD1
  MOV DS:[SI+6], AX
  MOV SI, OFFSET LSD
  CALL LOAD_PROGRAM
```

R11:

RET

DOS\_HANDLER ENDP

```
ERROR_RETURN PROC NEAR
  MOV AX, CONST
  MOV DS, AX
  MOV ES, AX
  MOV SP, STACK_POINTER1
  JMP R11
```

ERROR\_RETURN ENDP

```
UP_SET PROC NEAR
  MOV SI, OFFSET S_INTEGE
  CALL DOUBLE_DIS_LINE
```

UP1:

```
  MOV SI, OFFSET ENTER_UP
  CALL DOUBLE_DIS_LINE
  MOV DI, OFFSET MA_UP
  MOV SI, OFFSET INTEGERN
  CALL LFDP_ASC
  CALL FDP_NUMBER_EDITOR
  MOV BX, OFFSET UP_ERROR
  MOV DX, 21
  CALL ERROR_DETECT1
  JC UP1
  CALL ULFDP_ASC
  RET
```

UP\_SET ENDP

```
NOC_SET PROC NEAR
  MOV SI, OFFSET S_INTEGE
```

```

CALL DOUBLE_DIS_LINE
NC1:
MOV SI,OFFSET ENTER_NC
CALL DOUBLE_DIS_LINE
MOV DI,OFFSET MA_NC
MOV SI,OFFSET INTEGERN
CALL LFDP_ASC
CALL FDP_NUMBER_EDITOR
MOV BX,OFFSET NC_ERROR
MOV DX,11
CALL ERROR_DETECT1
JC NC1
CALL ULFDP_ASC
RET
NOC_SET ENDP

MA_PRO_SET PROC NEAR
AND AH,0FH
MOV MA_CON,AH
CALL DISPLAY_MA_PT
RET
MA_PRO_SET ENDP

AMBD_S1 PROC NEAR
PUSH AX
MOV SI,OFFSET S_INTEGE
CALL DOUBLE_DIS_LINE
POP AX

MOV DI,OFFSET MA_AMBF
MOV SI,OFFSET ENTER_AMBF
AND AH,0FH
CMP AH,1
JZ ASPN1
CMP AH,3
JZ ASPN2
ADD DI,20
MOV SI,OFFSET ENTER_PR
JMP ASPN1
ASPN2:
ADD DI,10
ASPN1:
CALL DOUBLE_DIS_LINE
MOV SI,OFFSET INTEGERN1
CALL LFDP_ASC
CALL FDP_NUMBER_EDITOR
CALL ULFDP_ASC
RET
AMBD_S1 ENDP

LC_SET PROC NEAR
PUSH AX
MOV SI,OFFSET S_NN
CALL COMPUTE_NUMBER
MOV BX,AX

```

```

POP AX
PUSH BX
PUSH AX
MOV SI,OFFSET SBU
CALL SAVE_SCREEN
CALL CLS
MOV SI,OFFSET A_P_NUMB
CALL DOUBLE_DIS_LINE
MOV SI,OFFSET LINK_C_UNIT
CALL DOUBLE_DIS_LINE
POP AX
AND AH,0FH
CMP AH,1
JNZ LC279
MOV AX,OFFSET LC1
JMP AX
LC279:
CMP AH,2
JNZ LC278
MOV AX,OFFSET LC2
JMP AX
LC278:
POP AX
CMP AL,2
JNZ LC893
DEC AL
LC893:
PUSH AX
MOV SI,OFFSET A_FLP_BOUND2
MOV DI,OFFSET A_FLP_BOUND3
MOV BYTE PTR [DI+4],AL
CALL FILL_RETURN
MOV WORD PTR [DI+BX+10],0D31H
POP AX
PUSH AX
CMP AL,1
JNZ LC894
MOV BYTE PTR [SI+BX+10],32H
MOV BYTE PTR [DI+BX+10],32H
LC894:
PUSH SI
PUSH DI
PUSH BX
CALL DOUBLE_DIS_LINE
MOV SI,OFFSET A_FLP_BOUND3
CALL DOUBLE_DIS_LINE
POP BX
POP DI
POP SI
POP AX
PUSH AX
MOV BYTE PTR [SI+BX+11]," "
MOV BYTE PTR [DI+BX+11]," "
ADD AL,31H
MOV [SI+BX+10],AL

```

```

MOV [DI+BX+10],AL
POP AX
MOV BX,AX
ADD AX,BX
PUSH AX
MOV DI,OFFSET LC_BL
MOV SI,OFFSET A_FLP_NUMBER4
JMP LC3
LC2:
POP AX
PUSH AX
MOV SI,OFFSET A_FLP_BOUND
CALL FILL_RETURN
PUSH SI
PUSH BX
CALL DOUBLE_DIS_LINE
POP BX
POP SI
POP AX
PUSH AX
MOV BYTE PTR [SI+BX+11]," "
ADD AL,31H
MOV [SI+BX+10],AL
MOV DI,OFFSET LC_UL
MOV SI,OFFSET A_FLP_NUMBER3
JMP LC3
LC1:
POP AX
DEC AL
PUSH AX
MOV SI,OFFSET A_FLP_BOUND4
MOV DI,OFFSET A_FLP_BOUNDS5
MOV BYTE PTR [DI+4],AL
CALL FILL_RETURN
MOV BYTE PTR [DI+BX+11],0DH
PUSH DI
PUSH SI
PUSH BX
CALL DOUBLE_DIS_LINE
MOV SI,OFFSET A_FLP_BOUNDS5
CALL DOUBLE_DIS_LINE
POP BX
POP SI
POP DI
MOV BYTE PTR [SI+BX+11]," "
MOV BYTE PTR [DI+BX+11]," "
MOV DI,OFFSET LC_S
MOV SI,OFFSET A_FLP_NUMBER4
POP AX
SHL AX,1
PUSH AX
LC3:
CALL LAF_ASC
POP AX
MOV [SI],AX

```

```
CALL A_FLP_EDITOR
CALL ULAF_ASC
MOV SI,OFFSET SBU
CALL RECOVER_SCREEN
RET
```

FILL RETURN:

```
MOV BYTE PTR [SI+4],AL
MOV DL,7
MUL DL
XOR BH,BH
MOV BL,AL
MOV WORD PTR [SI+BX+10],0D31H
RET
```

LC\_SET ENDP

ITM\_SET PROC NEAR

```
MOV SI,OFFSET S_NN
CALL COMPUTE_NUMBER
MOV SI,OFFSET A_FLP_S3
MOV DS:[SI+SQ_INDEX_NR],AL
MOV DS:[SI+SQ_INDEX_NEPWR],AL
MOV DS:[SI+SQ_INDEX_NEPR],AL
PUSH AX
MOV DL,9
MUL DL
ADD AX,5
MOV DS:[SI+SQ_INDEX_NC],AL
POP AX
PUSH AX
MOV DL,AL
MUL DL
MOV DS:[SI+SQ_INDEX_TNE],AL
MOV SI,OFFSET SBU
CALL SAVE_SCREEN
CALL CLS
MOV SI,OFFSET A_P_NUMB
CALL DOUBLE_DIS_LINE
MOV SI,OFFSET ITM_UNIT
CALL DOUBLE_DIS_LINE
MOV SI,OFFSET A_FLP_S3
CALL SQUARE_INDEX
MOV SI,OFFSET LEFT_F
POP AX
PUSH AX
MOV DL,6
MUL DL
XOR BH,BH
MOV BL,AL
MOV BYTE PTR [SI+BX+2],0DH
PUSH BX
PUSH SI
CALL DOUBLE_DIS_LINE
POP SI
```

```

POP BX
MOV BYTE PTR [SI+BX+2], " "
MOV DI, OFFSET ITM_S
MOV SI, OFFSET A_FLP_NUMBER2
POP AX
CALL SET_DISABLE
CALL LAF_ASC
CALL A_FLP_EDITOR
CALL ULAF_ASC
MOV SI, OFFSET SBU
CALL RECOVER_SCREEN
RET
ITM_SET ENDP

FLP_CONTROL EQU 15
AF_BUFFER_BEGIN EQU 36

SET_DISABLE PROC NEAR
    PUSH DI
    PUSH SI
    MOV CX, 64
    MOV [SI], CX
    ADD SI, AF_BUFFER_BEGIN
    XOR BX, BX
SD1:
    MOV DI, [SI]
    AND BYTE PTR [DI+FLP_CONTROL], 0BFH
    CMP BH, AL
    JAE SD7
    CMP BL, AL
    JC SD2
SD7:
    XOR BYTE PTR [DI+FLP_CONTROL], 40H
SD2:
    INC SI
    INC SI
    CMP BH, AL
    JAE SD4
    CMP BL, 7
    JC SD3
    INC BH
    XOR BL, BL
    JMP SD4
SD3:
    INC BL
SD4:
    LOOP SD1
    POP SI
    POP DI
    RET
SET_DISABLE ENDP

ITM_SET1 PROC NEAR
    MOV SI, OFFSET MA_UP
    CALL COMPUTE_NUMBER

```

```

MOV SI,OFFSET A_FLP_S2
MOV DS:[SI+SQ_INDEX_TNE],AL
PUSH AX
CMP AL,9
JC IT1
MOV AL,8
IT1:
MOV DS:[SI+SQ_INDEX_NEPWR],AL
MOV DS:[SI+SQ_INDEX_NEPR],AL
MOV DL,9
MUL DL
ADD AX,5
MOV DS:[SI+SQ_INDEX_NC],AL
POP AX
PUSH AX
MOV DL,8
DIV DL
CMP AH,0
JZ ITM9
INC AL
ITM9:
MOV DS:[SI+SQ_INDEX_NR],AL
MOV SI,OFFSET SBU
CALL SAVE_SCREEN
CALL CLS
MOV SI,OFFSET A_P_NUMB
CALL DOUBLE_DIS_LINE
MOV SI,OFFSET ITM_UNIT1
CALL DOUBLE_DIS_LINE
MOV SI,OFFSET A_FLP_S2
CALL SQUARE_INDEX
MOV DI,OFFSET ITM_MA
MOV SI,OFFSET A_FLP_NUMBER2
POP AX
PUSH AX
MOV DS:[SI],AX
CALL SET_ABLE
CALL LAF_ASC
POP AX
MOV DS:[SI],AX
CALL A_FLP_EDITOR
CALL ULAF_ASC
MOV SI,OFFSET SBU
CALL RECOVER_SCREEN
RET
ITM_SET1 ENDP

SET_ABLE PROC NEAR
PUSH SI
PUSH DI
MOV CX,DS:[SI]
ADD SI,AF_BUFFER_BEGIN
SA1:
MOV DI,[SI]
AND BYTE PTR [DI+FLP_CONTROL],0BFH

```

```
    INC SI
    INC SI
    LOOP SA1
    POP DI
    POP SI
    RET
SET_ABLE ENDP
```

```
REAL_TIME_SET PROC NEAR
    AND AH,0FH
    MOV REAL_TIME,AH
    CALL DISPLAY_REAL_TIME
    RET
REAL_TIME_SET ENDP
```

```
DISPLAY_REAL_TIME PROC NEAR
    MOV SI,OFFSET REAL_TIME_M
    CALL DOUBLE_DIS_LINE
    MOV SI,OFFSET REAL_TIME_OFF
    CMP REAL_TIME,0
    JZ REL1
    MOV SI,OFFSET REAL_TIME_ON
REL1:
    CALL DOUBLE_DIS_LINE
    RET
DISPLAY_REAL_TIME ENDP
```

```
TOPOLOGY_SET PROC NEAR
    AND AH,0FH
    MOV SAF_TP,AH
    CALL DISPLAY_SAF_TP
    MOV SAF_RTG,1
    CALL DISPLAY_SAF_RG
    RET
TOPOLOGY_SET ENDP
```

```
NON_SET PROC NEAR
    MOV SI,OFFSET S_INTEGE
    CALL DOUBLE_DIS_LINE
NNE10:
    MOV SI,OFFSET ENTER_NN
    CALL DOUBLE_DIS_LINE
    MOV DI,OFFSET S_NN
    MOV SI,OFFSET INTEGERN
    CALL LFDP_ASC
    CALL FDP_NUMBER_EDITOR
    MOV BX,OFFSET NN_ERROR
    MOV DL,9
    MOV DH,1
    CALL ERROR_DETECT1
    JC NNE10
    CALL ULFDP_ASC
    RET
```

```
ERROR_DETECT1:
```

```

PUSH SI
MOV CL,DS:[SI+L_COUNT_BYTE]
CMP CL,1
JNZ NNE1
MOV AX,BX
CALL ORDER_ERROR
JMP NNE2
NNE1:
PUSH DX
DEC CL
XOR CH,CH
XOR AX,AX
MOV DL,10
ADD SI,L_SECOND_DATA
NN3:
MUL DL
PUSH CX
MOV CL,DS:[SI]
SUB CL,30H
XOR CH,CH
ADD AX,CX
INC SI
POP CX
LOOP NN3
POP DX
CMP DH,AL
JAE NNE98
CMP AL,DL
JC NNE4
NNE98:
MOV AX,BX
CALL ORDER_ERROR
JMP NNE2
NNE4:
CLC
NNE5:
POP SI
RET
NNE2:
STC
JMP NNE5
NON_SET ENDP

SAF_OD_S1 PROC NEAR
MOV STACK_POINTER3,SP
CALL CLEAR_FDP_MARK
MOV DI,OFFSET PB_NM
MOV SI,OFFSET A_FDP_NUMBER
CALL LAD_ASC
SAF9:
CALL DISPLAY_SAF_OU
TEST NET_OD_SAF,3
JZ GET_M67
MOV SI,OFFSET A_FDP_NUMBER
CALL A_FDP_EDITOR

```

SAF7:  
CALL CLEAR\_FDP\_MARK  
MOV DI,OFFSET PB\_NM  
MOV SI,OFFSET A\_FDP\_NUMBER  
CALL ULAD\_ASC  
CALL CLS  
RET

GET\_M67:  
CALL GET\_MOUSE\_STATE

GET\_M65:  
CALL CLEAR\_BUFFER\_GET\_KEY2  
CMP AX,0  
JZ GET\_M70  
MOV BL,1  
MOV SI,OFFSET MA\_OD\_CTL\_K  
CALL KEY\_FUN\_I  
JC GET\_M65  
JMP GET\_M69

GET\_M70:  
CALL GET\_MOUSE\_STATE  
TEST CH,1  
JZ GET\_M65  
TEST BL,3  
JZ GET\_M65  
MOV SI,OFFSET SAF\_OD\_CTL  
CALL MOUSE\_FUN\_I  
JC GET\_M65

GET\_M69:  
CMP AL,0  
JZ SAF7  
MOV NET\_OD\_SAF,AL  
JMP SAF9  
RET

SAF\_OD\_S1 ENDP

MOUSE\_HANDLERS PROC NEAR

MOV BL,1  
MOV SI,OFFSET SAF\_OD\_CTL  
CALL MOUSE\_FUN\_I  
JC GET\_M68

GET\_M9742:  
MOV SP,STACK\_POINTER3  
PUSH AX  
MOV DI,OFFSET A\_FDP\_NUMBER  
MOV SI,[DI+2]  
CALL ERASE\_AD\_LAST  
POP AX  
CMP AL,0  
JZ SAF7  
CMP NET\_OD\_SAF,AL  
JNZ GET\_M99  
XOR NET\_OD\_SAF,AL  
JMP SAF9

GET\_M99:  
MOV NET\_OD\_SAF,AL

```

    JMP SAF9
GET_M68:
    RET
MOUSE_HANDLER8 ENDP

FUNCTION_KEY_HANDLER8 PROC NEAR
    MOV BL,1
    MOV SI,OFFSET MA_OD_CTL_K
    CALL KEY_FUN_I
    JC GET_M68
    JMP GET_M9742
    RET
FUNCTION_KEY_HANDLER8 ENDP

CLEAR_FDP_MARK PROC NEAR
    MOV SI,OFFSET A_FDP_NUMBER
    MOV CX,[SI]
    ADD SI,34
SAF89:
    MOV DI,[SI]
    AND BYTE PTR [DI+1],0FDH
    INC SI
    INC SI
    LOOP SAF89
    RET
CLEAR_FDP_MARK ENDP

DISPLAY_SAF_OU PROC NEAR
    MOV SI,OFFSET SAF_OD_MESS1
    CALL DOUBLE_DIS_LINE
    TEST NET_OD_SAF,1
    JNZ SAF1
        MOV SI,OFFSET SAF_OD_MESS3
        CALL DOUBLE_DIS_LINE
        MOV SI,OFFSET SAF_OD_PD_OFF
        CALL DOUBLE_DIS_LINE
SAF1:
    TEST NET_OD_SAF,2
    JNZ SAF2
        MOV SI,OFFSET SAF_OD_MESS5
        CALL DOUBLE_DIS_LINE
        MOV SI,OFFSET SAF_OD_PB_OFF
        CALL DOUBLE_DIS_LINE
SAF2:
    CALL CLEAR_FDP_MARK
    MOV SI,OFFSET A_FDP_NUMBER
    ADD SI,34
    TEST NET_OD_SAF,1
    JNZ SAF96
        MOV DI,[SI]
        XOR BYTE PTR [DI+1],2
        INC SI
        INC SI
        MOV DI,[SI]
        XOR BYTE PTR [DI+1],2

```

```

        INC SI
        INC SI
        JMP SAF961
SAF96:
        ADD SI,4
SAF961:
        TEST NET_OD_SAF,2
        JNZ SAF97
            MOV DI,[SI]
            XOR BYTE PTR [DI+1],2
            INC SI
            INC SI
            MOV DI,[SI]
            XOR BYTE PTR [DI+1],2
SAF97:
        RET
DISPLAY_SAF_OU ENDP

MA_OD_S1 PROC NEAR
        MOV STACK_POINTER3,SP
SAF924:
        CALL DISPLAY_MA_OU
        CALL MA_MOUSE_A
        JZ SAF923
        CMP NET_OD_MA,AL
        JNZ SAF883
        XOR NET_OD_MA,AL
        JMP SAF924
SAF883:
        MOV NET_OD_MA,AL
        JMP SAF924
SAF923:
        CALL CLS
        RET
MA_OD_S1 ENDP

DISPLAY_MA_OU PROC NEAR
        PUSH SI
        MOV SI,OFFSET MA_OD_MESS1
        CALL DOUBLE_DIS_LINE
        TEST NET_OD_MA,1
        JNZ SAF3
            MOV SI,OFFSET MA_OD_MESS3
            CALL DOUBLE_DIS_LINE
SAF3:
        TEST NET_OD_MA,2
        JNZ SAF91
            MOV SI,OFFSET MA_OD_MESS5
            CALL DOUBLE_DIS_LINE
SAF91:
        POP SI
        RET
DISPLAY_MA_OU ENDP

MA_MOUSE_A PROC NEAR

```

```

GET_M66:
  CALL CLEAR_BUFFER_GET_KEY2
  CMP AX,0
  JZ GET_M665
  MOV SI,OFFSET MA_OD_CTL_K
  CALL KEY_FUN_I
  JC GET_M66
  CMP AL,0
  RET

```

```

GET_M665:
  CALL GET_MOUSE_STATE
  TEST CH,1
  JZ GET_M66
  TEST BL,3
  JZ GET_M66
  MOV SI,OFFSET MA_OD_CTL
  CALL MOUSE_FUN_I
  JC GET_M66
  CMP AL,0
  RET

```

```

MA_MOUSE_A ENDP

```

```

DISPLAY_NETCON PROC NEAR
  CALL RECOVER_NETCON
  MOV SI,OFFSET NETCON_M
  CALL DOUBLE_DIS_LINE
  CMP NET_CON,0
  JZ DNC4
  CMP NET_CON,1
  JZ DNC2
  MOV SI,OFFSET NETCON_MA
  JMP DNC3

```

```

DNC4:
  MOV SI,OFFSET NETCON_UD
  JMP DNC3

```

```

DNC2:
  MOV SI,OFFSET NETCON_SAF

```

```

DNC3:
  CALL DOUBLE_DIS_LINE

  CALL DISPLAY_MA_PT
  CALL DISPLAY_SAF_TP
  CALL DISPLAY_SAF_RG
  CALL DISPLAY_REAL_TIME
  RET

```

```

DISPLAY_NETCON ENDP

```

```

RECOVER_NETCON PROC NEAR
  MOV SI,OFFSET NETCON_C
  CALL DOUBLE_DIS_LINE
  RET

```

```

RECOVER_NETCON ENDP

```

```

DISPLAY_MA_PT PROC NEAR
  CMP NET_CON,2

```

```

JZ DMP1
RET
DMP1:
MOV SI,OFFSET NETCON_MA_M
CALL DOUBLE_DIS_LINE
CMP MA_CON,0
JZ DMP5
CMP MA_CON,3
JZ DMP6
CMP MA_CON,2
JZ DMP3
MOV SI,OFFSET NETCON_MA_A
JMP DMP4
DMP5:
MOV SI,OFFSET NETCON_MA_UD
JMP DMP4
DMP3:
MOV SI,OFFSET NETCON_MA_TCR
DMP4:
CALL DOUBLE_DIS_LINE
RET
DMP6:
MOV SI,OFFSET NETCON_MA_CSMA
JMP DMP4
DISPLAY_MA_PT ENDP

```

```

DISPLAY_SAF_TP PROC NEAR
CMP NET_CON,1
JZ DMP11
RET
DMP11:
MOV SI,OFFSET NETCON_TP_M
CALL DOUBLE_DIS_LINE
CMP SAF_TP,0
JZ DMP10
CMP SAF_TP,2
JZ DMP7
CMP SAF_TP,3
JZ DMP9
MOV SI,OFFSET NETCON_TP_S
JMP DMP8
DMP9:
MOV SI,OFFSET NETCON_TP_BL
JMP DMP8
DMP10:
MOV SI,OFFSET NETCON_TP_UD
JMP DMP8
DMP7:
MOV SI,OFFSET NETCON_TP_UL
DMP8:
CALL DOUBLE_DIS_LINE
RET
DISPLAY_SAF_TP ENDP

```

```

DISPLAY_SAF_RG PROC NEAR

```

```

    CMP NET_CON, 1
    JZ DMR1
    RET
DMR1:
    MOV SI, OFFSET NETCON_RG_M
    CALL DOUBLE_DIS_LINE
    CMP SAF_RTG, 0
    JZ DMR5
    CMP SAF_RTG, 2
    JZ DMR3
    MOV SI, OFFSET NETCON_RG_S
    JMP DMR4
DMR5:
    MOV SI, OFFSET NETCON_RG_UD
    JMP DMR4
DMR3:
    MOV SI, OFFSET NETCON_RG_R
DMR4:
    CALL DOUBLE_DIS_LINE
    RET
DISPLAY_SAF_RG ENDP

SET_NET_CON PROC NEAR
    MOV SI, OFFSET NET_OD_S1+LIST_OVERHEAD
    AND BYTE PTR [SI+STATUS], 0EH
    XOR BYTE PTR [SI+STATUS], 1
    ADD SI, ELEMENT_SIZE
    AND BYTE PTR [SI+STATUS], 0EH
    XOR BYTE PTR [SI+STATUS], 1
    TEST AH, 10H
    JZ DNC45
    MOV NET_CON, 1
    MOV SI, OFFSET NET_OD_S1+LIST_OVERHEAD+ELEMENT_SIZE
    JMP DNC5
DNC45:
    MOV NET_CON, 2
    MOV SI, OFFSET NET_OD_S1+LIST_OVERHEAD
DNC5:
    AND BYTE PTR [SI+STATUS], 0EH
    CALL DISPLAY_NETCON
    RET
SET_NET_CON ENDP

LIST_OVERHEAD EQU 5
ELEMENT_SIZE EQU 12
STATUS EQU 0

DISABLE_ED PROC NEAR
    MOV SI, OFFSET ENCODER_S1+LIST_OVERHEAD
    CALL RESET_ED
    MOV SI, OFFSET DECODER_S1+LIST_OVERHEAD
    CALL RESET_ED
    CMP ECD_CON, 0
    JNZ DED4
    CMP DCD_CON, 0

```

```

    JNZ DED5
    RET
DED4:
    CMP ECD_CON,1
    JZ DED1
    CMP ECD_CON,2
    JZ DED2
    CMP ECD_CON,3
    JZ DED2
    CMP ECD_CON,5
    JZ DED3
    RET
DED1:
    MOV SI,OFFSET DECODER_S1+LIST_OVERHEAD
DED10:
    AND BYTE PTR [SI+STATUS],0EH
    ADD SI,ELEMENT_SIZE
    AND BYTE PTR [SI+STATUS],0EH
    RET
DED2:
    MOV SI,OFFSET DECODER_S1+LIST_OVERHEAD+ELEMENT_SIZE
    JMP DED10
DED3:
    MOV SI,OFFSET DECODER_S1+LIST_OVERHEAD
DED11:
    AND BYTE PTR [SI+STATUS],0EH
    ADD SI,ELEMENT_SIZE
    ADD SI,ELEMENT_SIZE
    AND BYTE PTR [SI+STATUS],0EH
    RET
DED5:
    CMP DCD_CON,1
    JZ DED6
    CMP DCD_CON,2
    JZ DED7
    CMP DCD_CON,3
    JZ DED7
    CMP DCD_CON,4
    JZ DED8
    RET
DED6:
    MOV SI,OFFSET ENCODER_S1+LIST_OVERHEAD
    JMP DED10
DED7:
    MOV SI,OFFSET ENCODER_S1+LIST_OVERHEAD+ELEMENT_SIZE
    JMP DED10
DED8:
    MOV SI,OFFSET ENCODER_S1+LIST_OVERHEAD
    JMP DED11
DISABLE_ED ENDP

RESET_ED PROC NEAR
    AND BYTE PTR [SI+STATUS],0EH
    XOR BYTE PTR [SI+STATUS],1
    ADD SI,ELEMENT_SIZE

```

```
AND BYTE PTR [SI+STATUS],0EH
XOR BYTE PTR [SI+STATUS],1
ADD SI,ELEMENT_SIZE
AND BYTE PTR [SI+STATUS],0EH
XOR BYTE PTR [SI+STATUS],1
RET
RESET_ED ENDP
```

```
CODE ENDS
END
```

```
; ***** LN1.ASM *****
```

```
; ALOHA.TDT:  
; TREE.TDT  
; CSMA.TDT
```

```
; User population  
; Number of channels  
; Input traffic matrix: User 1  
; : User 2  
; .  
; .  
; .  
; : User 12  
; Maximum simulation time  
; Delay-throughput switch  
; Real time on-off switch
```

```
; STARNET.TDT:  
; Number of nodes  
; Traffic matrix: 64 numbers  
; (0,1)  
; (0,2)  
; .  
; .  
; (0,7)  
; (1,0)  
; .  
; .  
; (1,7)  
; .  
; .  
; (7,7)  
; Link capacities: 7 numbers  
; (1,0)  
; (2,0)  
; (3,0)  
; .  
; .  
; (7,0)
```

```
; UNILoop.TDT:  
; Number of nodes  
; Traffic matrix: 64 numbers  
; (0,1)  
; (0,2)  
; .  
; .  
; .  
; (0,7)  
; (1,0)
```



```

; Sampling frequency
; Cutoff frequency
; Coefficient #0
; Coefficient #1
;
;
;
; Coefficient #98
;
; CHLFLTR2.TDT: One of the following four cases:
; Butterworth case:
; Attenuation sampling frequency
; Passband edge frequency
; Stopband edge frequency
; Passband attenuation
; Stopband attenuation
; Chebychev case:
; Same as Butterworth case
; Elliptic case:
; Same as Butterworth case
; FIR case:
; Same as Butterworth case
;
; RCVRFLTR1.TDT: Same as CHLFLTR1.TDT
; RCVRFLTR2.TDT: Same as CHLFLTR2.TDT
;
; CHANNEL.TDT: One of the following three cases:
; White Gaussian noise:
; Signal-to-noise ratio
; BSC case:
; bit error rate
; Propagation Model:
; Type : 0 : Open Area + No Foliage
;         1 : Suburban Area + No Foliage
;         2 : Open Area + Foliaged
;         3 : Suburban Area + Foliaged
; Transmitter Antenna Height
; Receiver Antenna Height
; Transmitter Average Power
; Transmitter Antenna Gain
; Receiver Antenna Gain
; Carrier Frequency
; Distance between Transmitter and Receiver
; Receiver Noise Power
; Receiver Bandwidth
;
STACK SEGMENT 'STACK' STACK
STACK ENDS

EXTRN BIT_RATE:BYTE,AMPLITUDE SIN:BYTE,AMPLITUDE_SW:BYTE
EXTRN SOU_NO:BYTE,SF_BW_CF:BYTE,CHL_NO:BYTE,CHL_ER:BYTE
EXTRN SBU:BYTE,MA_UP:BYTE,NET_CON:BYTE,SAF_TP:BYTE
EXTRN SOU_CON:BYTE,MOD_CON:BYTE,ECD_CON:BYTE,CF_CON:BYTE
EXTRN RF_CON:BYTE,DCD_CON:BYTE,DEMOD_CON:BYTE,CHL_CON:BYTE
EXTRN ITM_MA:BYTE,ITM_S:BYTE,S_NN:BYTE,LC_BL:BYTE,LC_UL:BYTE

```

EXTRN LC\_S:BYTE,ORDER\_BW\_CF:BYTE,FIR\_CF:BYTE,PM\_CON:BYTE,PM\_CONST:BYTE  
 EXTRN CF\_BW\_T:BYTE,NET\_OD\_SAF:BYTE,PB\_SN:BYTE  
 EXTRN NET\_OD\_MA:BYTE,MA\_CON:BYTE,SAF\_RTG:BYTE,REAL\_TIME:BYTE  
 EXTRN NUMBER\_ITERATION:BYTE,MA\_AMBF:BYTE,MA\_CMBF:BYTE

CONST SEGMENT 'DATA' PARA PUBLIC

O\_LINK NET DB 8 DUP (?)  
 LOAD\_STATUS DB 0  
 DATA\_FILE\_NAME1 DB "PBRESULT.DAT",0  
 DATA\_FILE\_NAME2 DB "TXFILT.DAT",0  
 DATA\_FILE\_NAME3 DB "RCFILT.DAT",0  
 SOU\_CON\_TDT DB "SOU\_CON.TDT",0  
 STOP\_TDT DB "STOP.TDT",0  
 STOP\_TDT1 DB 1  
           DB 0  
           DW OFFSET NUMBER\_ITERATION  
           DW 1  
           DW 13  
 SOU\_TDT DB "SOURCE.TDT",0  
 SOU\_TDT1 DB 4  
           DB 1  
           DW OFFSET BIT\_RATE  
           DW 1  
           DW 15  
           DB 2  
           DW OFFSET AMPLITUDE\_SIN  
           DW 2  
           DW 15  
           DB 3  
           DW OFFSET AMPLITUDE\_SW  
           DW 2  
           DW 15  
           DB 4  
           DW OFFSET SOU\_NO  
           DW 1  
           DW 15  
 CF\_TDTA DB "CHLFLTR1.TDT",0  
 CF\_TDTB DB "CHLFLTR2.TDT",0  
 CF\_TDT1 DB 4  
           DB 3  
           DW OFFSET SF\_BW\_CF  
           DW 2  
           DW 15  
           DB 4  
           DW OFFSET SF\_BW\_CF+105  
           DW 2  
           DW 15  
           DB 5  
           DW OFFSET SF\_BW\_CF+210  
           DW 2  
           DW 15  
           DB 6  
           DW OFFSET SF\_BW\_CF+315  
           DW 2  
           DW 15

```

CF_TDT2 DB 4
        DB 3
        DW OFFSET ORDER_BW_CF
        DW 1
        DW 5
        DB 4
        DW OFFSET ORDER_BW_CF+5
        DW 1
        DW 5
        DB 5
        DW OFFSET ORDER_BW_CF+10
        DW 1
        DW 5
        DB 6
        DW OFFSET ORDER_BW_CF+20
        DW 1
        DW 5
CF_TDT3 DB 1
        DB 6
        DW OFFSET FIR_CF+3
        DW 98
        DW 14
CF_TDT4 DB 4
        DB 3
        DW OFFSET SF_BW_CF+30
        DW 5
        DW 15
        DB 4
        DW OFFSET SF_BW_CF+135
        DW 5
        DW 15
        DB 5
        DW OFFSET SF_BW_CF+240
        DW 5
        DW 15
        DB 6
        DW OFFSET SF_BW_CF+345
        DW 5
        DW 15
RF_TDTA DB "RCVFLTR1.TDT",0
RF_TDTB DB "RCVFLTR2.TDT",0
RF_TDT1 DB 4
        DB 3
        DW OFFSET SF_BW_CF+420
        DW 2
        DW 15
        DB 4
        DW OFFSET SF_BW_CF+525
        DW 2
        DW 15
        DB 5
        DW OFFSET SF_BW_CF+630
        DW 2
        DW 15
        DB 6

```

```

        DW OFFSET SF_BW_CF+735
        DW 2
        DW 15
RF_TDT2 DB 4
        DB 3
        DW OFFSET ORDER_BW_CF+25
        DW 1
        DW 5
        DB 4
        DW OFFSET ORDER_BW_CF+30
        DW 1
        DW 5
        DB 5
        DW OFFSET ORDER_BW_CF+35
        DW 1
        DW 5
        DB 6
        DW OFFSET ORDER_BW_CF+45
        DW 1
        DW 5
RF_TDT3 DB 1
        DB 6
        DW OFFSET FIR_CF+1406
        DW 98
        DW 14
RF_TDT4 DB 4
        DB 3
        DW OFFSET SF_BW_CF+450
        DW 5
        DW 15
        DB 4
        DW OFFSET SF_BW_CF+555
        DW 5
        DW 15
        DB 5
        DW OFFSET SF_BW_CF+660
        DW 5
        DW 15
        DB 6
        DW OFFSET SF_BW_CF+765
        DW 5
        DW 15
CHL_TDT DB "CHANNEL.TDT",0
CHL_TDT1 DB 3
        DB 2
        DW OFFSET CHL_NO
        DW 1
        DW 15
        DB 3
        DW OFFSET CHL_ER
        DW 1
        DW 15
        DB 4
        DW OFFSET PM_CONST
        DW 9

```

```

        DW 15
COD_TDT DB "CODERATE.TDT",0
MOD_TDT DB "MODSIZE.TDT",0
M_ARY DB 10
        DB 1,1
        DB 2,2
        DB 3,3
        DB 4,4
        DB 5,6
        DB 6,1
        DB 7,2
        DB 8,3
        DB 10,1
        DB 20,1

ALOHA_TDT DB "ALOHA.TDT",0
TREE_TDT DB "TREE.TDT",0
CSMA_TDT DB "CSMA.TDT",0
MA_TDT1 DB 1
        DB 0
        DW OFFSET MA_UP
        DW 2
        DW 5
MA_TDT2 DB 1
        DB 0
        DW OFFSET ITM_MA+3
        DW 20
        DW 9
MA_TDT3 DB 1
        DB 0
        DW OFFSET NUMBER_ITERATION
        DW 1
        DW 13
MA_TDT4 DB 1
        DB 0
        DW OFFSET MA_AMBF
        DW 1
        DW 10
MA_TDT5 DB 1
        DB 0
        DW OFFSET MA_CMBF
        DW 2
        DW 10
STAR_TDT DB "STARNET.TDT",0
S_TDT1 DB 1
        DB 0
        DW OFFSET LC_S+3
        DW 8
        DW 18
S_TDT3 DB 1
        DB 0
        DW OFFSET LC_S+12
        DW 8
        DW 18
S_TDT2 DB 1

```

```

DB 0
DW OFFSET PB_SN
DW 4
DW 5
ULOOP_TDT DB "UNILOOP.TDT",0
U_TDT1 DB 1
DB 0
DW OFFSET LC_UL+3
DW 8
DW 9
BLOOP_TDT DB "BIDLOOP.TDT",0
B_TDT1 DB 1
DB 0
DW OFFSET LC_BL+3
DW 8
DW 18
B_TDT3 DB 1
DB 0
DW OFFSET LC_BL+12
DW 8
DW 18
AFS_TDT1 DB 1
DB 0
DW OFFSET S_NN
DW 1
DW 5
AFS_TDT2 DB 1
DB 0
DW OFFSET ITM_S+3
DW 64
DW 9
FILE_HANDLE DW ?
STACK_POINTER1 DW ?
GDF DB OFFH,59H,28,1,3,23,"□┘└|"
DB OFFH,30,3,"Generating Data Files",OFFH,79,25,0DH
LDB DB OFFH,59H,28,1,3,24,"□┘└|"
DB OFFH,30,3,"Loading Star Data Base",OFFH,79,25,0DH
GUS DB OFFH,59H,25,1,3,29,"□┘└|"
DB OFFH,27,3,"Generating Universe Program",OFFH,79,25,0DH
INCOM_DEF DB OFFH,59H,24,1,5,31,"□┘└|"
DB OFFH,26,3,"Incomplete Link Configuration"
DB OFFH,27,5,"Strike any key to continue"
DB OFFH,56H,5,"K",OFFH,56H,5,"L",OFFH,79,25,0DH
INVAL_DEF DB OFFH,59H,24,1,5,30,"□┘└|"
DB OFFH,27,3,"Invalid Link Configuration"
DB OFFH,27,5,"Strike any key to continue"
DB OFFH,56H,5,"K",OFFH,56H,5,"L",OFFH,79,25,0DH
INCOM_N_DEF DB OFFH,59H,23,1,5,34,"□┘└|"
DB OFFH,25,3,"Incomplete Network Configuration"
DB OFFH,27,5,"Strike any key to continue"
DB OFFH,56H,5,"K",OFFH,56H,5,"L",OFFH,79,25,0DH

LSDS DB 2
DW CONST
DW OFFSET DISBU

```

DW ?  
DW ?  
DB 1  
DW OFFSET LSDS1  
LSDS1 DW OFFSET A3  
DW CONST  
DB 0  
DB ?  
DW ?  
DW OFFSET AS4  
DW CONST  
DW OFFSET FCB1  
DW CONST  
DW OFFSET FCB2  
DW CONST  
DW OFFSET DISBU  
DW CONST

LSD DB 2  
DW CONST  
DW OFFSET DISBU  
DW ?  
DW ?  
DB 3  
DW OFFSET LSD1  
DW OFFSET LSD3  
DW OFFSET LSD4

LSD1 DW OFFSET A3  
DW CONST  
DB 0  
DB ?  
DW ?  
DW OFFSET A4  
DW CONST  
DW OFFSET FCB1  
DW CONST  
DW OFFSET FCB2  
DW CONST  
DW OFFSET DISBU  
DW CONST

LSD3 DW OFFSET A3  
DW CONST  
DB 0  
DB ?  
DW ?  
DW OFFSET A6  
DW CONST  
DW OFFSET FCB1  
DW CONST  
DW OFFSET FCB2  
DW CONST  
DW OFFSET DISBU  
DW CONST

LSD4 DW OFFSET A3

```
DW CONST
DB 0
DB ?
DW ?
DW OFFSET A8
DW CONST
DW OFFSET FCB1
DW CONST
DW OFFSET FCB2
DW CONST
DW OFFSET DISBU
DW CONST
```

```
A3 DB "LOAD.COM",0
A4 DB 7," CC UNI",ODH
A6_LENGTH EQU 39
A6 DB 10H," LINK20 BS BB BF UNI "
LIB_STRING DB 200 DUP (?)
A7_LENGTH EQU 18
A7 DB "CLIB PASLIB -oUNI",ODH
A8 DB 4," UNI",ODH
A9 DB 8," UNILoop",ODH
AS4 DB 8," STARNET",ODH
AS5 DB 7," PLOTpb",ODH
AS6 DB 9," PLTTHRPT",ODH
AS7 DB 9," PLTDELAY",ODH
AS9 DB 9," PLTQUEUE",ODH
AS8 DB 6," ALOHA",ODH
AS10 DB 5," TREE",ODH
AS11 DB 5," CSMA",ODH
AS12 DB 9," -Z UNI.C",ODH
AS13 DB 9," PLOTCONS",ODH
AS14 DB 9," PLTTXINP",ODH
AS15 DB 9," PLTRCOUT",ODH
AS16 DB "POSTTEXT.DAT",0
AS18 DB 8," BIDLOOP",ODH
AS19 DB 9," PLTTXOUT",ODH
AS20 DB 9," PLTRCINP",ODH
FCB1 DB 30H DUP (?)
FCB2 DB 30H DUP (?)
DISBU DB 200 DUP (?)
SBU2 DB 4000 DUP (?)
```

```
D1 DB 2
DB 7
DW OFFSET B1
DB 9
DW OFFSET B2
B1 DB "negarap"
B2 DB "ssuagarap"
```

```
C1 DB 225
DB "#include <stdio.h>",ODH,0AH
DB "#include ",22H,"type.h",22H,ODH,0AH
DB "#define FIFO 100",ODH,0AH
```

```

DB "struct(long int sample_stp;",0DH,0AH,"          int seed"
DB ");paragen;",0DH,0AH,"struct(long int samp_stp;",0DH,0AH
DB "          int seed;",0DH,0AH,"          double stddev;)paragauss"
DB ";;",0DH,0AH
DB "float tmp;",0DH,0AH
DB "FILE *fp, *fopen();",0DH,0AH

C2 DW 188
DB "fp=fopen(",22H,"stop.tdt",22H,"",22H,"r",22H,");",0DH,0AH
DB "fscanf(fp,",22H,"%f",22H,"",&tmp);",0DH,0AH
DB "fclose(fp);",0DH,0AH
DB "paragen.seed=1099;",0DH,0AH
DB "paragen.sample_stp=(int)tmp;",0DH,0AH
DB "paragauss.seed=19985;",0DH,0AH
DB "paragauss.samp_stp=200000000;",0DH,0AH
DB "paragauss.stddev=0;",0DH,0AH

ERROR_COUNT1 DW ?
ERROR_PROGRAM1 DB 80 DUP (?)
ERROR_PC1 EQU 15
ERROR_P1 DB ",1,",22H,"Display",22H,"",1,"
ERROR_PC2 EQU 9
ERROR_P2 DB "*FIFO);",0DH,0AH
ERROR_PC3 EQU 8
ERROR_P3 DB "connect("
ADDER_LENGTH EQU 38
ADDER_LINE DB "connect(",22H,"Noise generator",22H,"",0,"",22H
DB "Adder",22H,"",1,"
ADDER_LENGTH1 EQU 9
ADDER_LINE1 DB "*FIFO);",0DH,0AH
STAR_NOISE_LENGTH EQU 61
STAR_NOISE DB "star(",22H,"Noise generator",22H,"",gauss,"
DB "&paragauss,sizeof(paragauss);",0DH,0AH
NUMBER_OF_BYTE DW 0

STAR_COUNT DW ?
CONNECT_COUNT DW ?
INT_COUNT DW ?
INT_COUNT1 DW ?
STAR_BASE_DAT DB "STAR_BAS.DAT",0
UNIVERSE_Q DB "UNI.C",0
STAR_BASE DB 4000 DUP (?)
STAR_POINTER DW 9 DUP (?)
DECLARE_INT DB 1500 DUP (?)
C_OVERHEAD DB "universe()",0DH,0AH,"{",0DH,0AH
C_END DB "return(0);",0DH,0AH,"}",0DH,0AH
DECLARE_INT1 DB 1500 DUP (?)
STAR_PROGRAM DB 2500 DUP (?)
CONNECT_PROGRAM DB 2500 DUP (?)

SOU_FL DB 4
DB 1
DW OFFSET SOURCE
DB 2
DW OFFSET SOURCE_SIN

```

DB 3  
 DW OFFSET SOURCE  
 DB 4  
 DW OFFSET SOURCE\_WGN  
 MOD\_FL DB 9  
 DB 1  
 DW OFFSET BPSK  
 DB 2  
 DW OFFSET QPSK  
 DB 3  
 DW OFFSET PSK8  
 DB 4  
 DW OFFSET QAM16  
 DB 5  
 DW OFFSET QAM64  
 DB 6  
 DW OFFSET BFSK  
 DB 7  
 DW OFFSET FSK4  
 DB 8  
 DW OFFSET FSK4  
 DB 10  
 DW OFFSET TBPSKMOD  
 ECD\_FL DB 6  
 DB 1  
 DW OFFSET NOCOD  
 DB 2  
 DW OFFSET CON3  
 DB 3  
 DW OFFSET CON7  
 DB 4  
 DW OFFSET CON7  
 DB 5  
 DW OFFSET RS15  
 DB 6  
 DW OFFSET RS15  
 DIS\_FL DB 5  
 DB 1  
 DW OFFSET DIS\_UNC  
 DB 2  
 DW OFFSET DIS\_CO  
 DB 3  
 DW OFFSET DIS\_CO  
 DB 5  
 DW OFFSET DIS\_UNC  
 DB 9  
 DW OFFSET SINK  
 CF\_FL DB 5  
 DB 3  
 DW OFFSET BUTTERWORTH\_CF  
 DB 4  
 DW OFFSET CHEBYCHEV\_CF  
 DB 5  
 DW OFFSET ELLIPTIC\_CF  
 DB 6

```

        DW OFFSET FIR_CF_FILE
        DB 7
RF_FL  DW OFFSET FIR_CF_FILE
        DB 5
        DB 3
        DW OFFSET BUTTERWORTH_RF
        DB 4
        DW OFFSET CHEBYCHEV_RF
        DB 5
        DW OFFSET ELLIPTIC_RF
        DB 6
        DW OFFSET FIR_RF
        DB 7
DCD_FL DW OFFSET FIR_RF
        DB 6
        DB 1
        DW OFFSET NODEC
        DB 4
        DW OFFSET RS_DEC
        DB 2
        DW OFFSET VIT3
        DB 3
        DW OFFSET VIT7
        DB 5
        DW OFFSET VIT3Q
        DB 6
        DW OFFSET VIT7Q
DEMOD_FL DB 9
        DB 1
        DW OFFSET SOFT
        DB 4
        DW OFFSET BPSK_DEM
        DB 5
        DW OFFSET QPSK_DEM
        DB 6
        DW OFFSET PSK8_DEM
        DB 7
        DW OFFSET QAM16_DEM
        DB 8
        DW OFFSET QAM64_DEM
        DB 9
        DW OFFSET BFSK_DEM
        DB 10
        DW OFFSET FSK4_DEM
        DB 13
        DW OFFSET TBPSKDEM
CHL_FL DB 5
        DB 2
        DW OFFSET FGAUSS
        DB 4
        DW OFFSET PGAUSS
        DB 7
        DW OFFSET BSC
        DB 8
        DW OFFSET CBSC

```

```

DB 9
DW OFFSET CBSC
SOURCE DB 4,"gen "
SOURCE_SIN DB 4,"sin "
SOURCE_WGN DB 4,"wgn "
BPSK DB 6,"bpmmod "
QPSK DB 6,"qpmod "
PSK8 DB 6,"8pmod "
QAM16 DB 7,"16qmod "
QAM64 DB 7,"64qmod "
BFSK DB 6,"bfmod "
FSK4 DB 6,"qfmod "
NOCOD DB 6,"nocod "
CON3 DB 6,"conv3 "
CON7 DB 6,"conv7 "
RS15 DB 6,"rscod "
BUTTERWORTH_CF DB 6,"txbut "
CHEBYCHEV_CF DB 7,"txcheb "
ELLIPTIC_CF DB 6,"txell "
FIR_CF_FILE DB 6,"txfir "
BUTTERWORTH_RF DB 6,"rcbut "
CHEBYCHEV_RF DB 7,"rccheb "
ELLIPTIC_RF DB 6,"rcell "
FIR_RF DB 6,"rcfir "
NODEC DB 6,"nodec "
SOFTV DB 6,"softv "
HARDV DB 6,"hardv "
RS_DEC DB 6,"rsdec "
SOFT DB 7,"no_dem "
NONE DB 5,"none "
BPSK_DEM DB 6,"bpdem "
QPSK_DEM DB 6,"qpdem "
PSK8_DEM DB 6,"8pdem "
QAM16_DEM DB 7,"16qdem "
QAM64_DEM DB 7,"64qdem "
BFSK_DEM DB 6,"bfdem "
FSK4_DEM DB 6,"qfdem "
FGAUSS DB 10,"gauss add "
PGAUSS DB 9,"prop add "
BSC DB 4,"bsc "
CBSC DB 5,"cbsc "
DIS_UNC DB 5,"dunc "
DIS_CO DB 6,"dconv "
SINK DB 5,"sink "
VIT3 DB 5,"vit3 "
VIT7 DB 5,"vit7 "
VIT3Q DB 6,"vit3q "
VIT7Q DB 6,"vit7q "
TBPSKMOD DB 7,"tbpmod "
TBPSKDEM DB 7,"tbpdem "

FIFO_POINTER DW 0
VALID_TABLE DB 27
DW OFFSET AC1
DW OFFSET BC1

```

DW OFFSET AC2  
DW OFFSET BC2  
DW OFFSET AC3  
DW OFFSET BC3  
DW OFFSET AC4  
DW OFFSET BC4  
DW OFFSET AC5  
DW OFFSET BC5  
DW OFFSET AC6  
DW OFFSET BC6  
DW OFFSET AC7  
DW OFFSET BC7  
DW OFFSET AC8  
DW OFFSET BC8  
DW OFFSET AC9  
DW OFFSET BC9  
DW OFFSET AC10  
DW OFFSET BC10  
DW OFFSET AC11  
DW OFFSET BC11  
DW OFFSET AC12  
DW OFFSET BC12  
DW OFFSET AC13  
DW OFFSET BC13  
DW OFFSET AC14  
DW OFFSET BC14  
DW OFFSET AC15  
DW OFFSET BC15  
DW OFFSET AC16  
DW OFFSET BC16  
DW OFFSET AC17  
DW OFFSET BC17  
DW OFFSET AC18  
DW OFFSET BC18  
DW OFFSET AC19  
DW OFFSET BC19  
DW OFFSET AC20  
DW OFFSET BC20  
DW OFFSET AC21  
DW OFFSET BC21  
DW OFFSET AC22  
DW OFFSET BC22  
DW OFFSET AC23  
DW OFFSET BC23  
DW OFFSET AC24  
DW OFFSET BC24  
DW OFFSET AC25  
DW OFFSET BC25  
DW OFFSET AC26  
DW OFFSET BC26  
DW OFFSET AC27  
DW OFFSET BC27

AC1 DB 2

DB 1,1,1,1,2,1,2,1

DB 1,1,1,1,4,1,2,1  
BC1 DB 1,1,1,1,1,1,1,1

AC2 DB 2  
DB 1,1,2,1,2,1,2,1  
DB 1,1,2,1,4,1,2,1  
BC2 DB 2,2,2,2,2,2,2,2

AC3 DB 2  
DB 1,1,3,1,2,1,2,1  
DB 1,1,3,1,4,1,2,1  
BC3 DB 3,3,3,2,2,3,3,2

AC4 DB 2  
DB 1,1,4,1,2,1,2,1  
DB 1,1,4,1,4,1,2,1  
BC4 DB 4,4,4,2,2,4,4,2

AC5 DB 2  
DB 1,1,5,1,2,1,2,1  
DB 1,1,5,1,4,1,2,1  
BC5 DB 6,6,6,2,2,6,6,2

AC6 DB 2  
DB 1,1,6,1,2,1,2,1  
DB 1,1,6,1,4,1,2,1  
BC6 DB 1,1,1,4,4,1,1,4

AC7 DB 2  
DB 1,1,7,1,2,1,2,1  
DB 1,1,7,1,4,1,2,1  
BC7 DB 2,2,2,8,8,2,2,8

AC8 DB 8  
DB 1,2,1,1,2,1,1,2  
DB 1,2,1,1,4,1,1,2  
DB 1,3,1,1,2,1,1,2  
DB 1,3,1,1,4,1,1,2  
DB 1,2,1,1,2,1,2,3  
DB 1,2,1,1,4,1,2,3  
DB 1,3,1,1,2,1,2,3  
DB 1,3,1,1,4,1,2,3  
BC8 DB 1,1,2,2,2,2,1,2

AC9 DB 8  
DB 1,2,2,1,2,1,1,2  
DB 1,2,2,1,4,1,1,2  
DB 1,3,2,1,2,1,1,2  
DB 1,3,2,1,4,1,1,2  
DB 1,2,2,1,2,1,2,3  
DB 1,2,2,1,4,1,2,3  
DB 1,3,2,1,2,1,2,3  
DB 1,3,2,1,4,1,2,3  
BC9 DB 1,1,2,2,2,2,1,2

AC10 DB 4  
DB 1,2,6,1,2,1,2,3  
DB 1,2,6,1,4,1,2,3  
DB 1,3,6,1,2,1,2,3  
DB 1,3,6,1,4,1,2,3  
BC10 DB 1,1,2,8,8,2,1,8

AC11 DB 2  
DB 1,5,4,1,2,1,2,4  
DB 1,5,4,1,4,1,2,4  
BC11 DB 18,18,30,15,15,30,18,15

AC12 DB 1  
DB 1,1,9,1,3,1,3,1  
BC12 DB 1,1,1,1,1

AC13 DB 2  
DB 1,2,9,1,3,1,3,3  
DB 1,3,9,1,3,1,3,3  
BC13 DB 1,1,2,2,1

AC14 DB 32  
DB 1,1,1,3,2,3,2,1  
DB 1,1,1,3,4,3,2,1  
DB 1,1,1,4,2,3,2,1  
DB 1,1,1,4,4,3,2,1  
DB 1,1,1,5,2,3,2,1  
DB 1,1,1,5,4,3,2,1  
DB 1,1,1,6,2,3,2,1  
DB 1,1,1,6,4,3,2,1  
DB 1,1,1,3,2,4,2,1  
DB 1,1,1,3,4,4,2,1  
DB 1,1,1,4,2,4,2,1  
DB 1,1,1,4,4,4,2,1  
DB 1,1,1,5,2,4,2,1  
DB 1,1,1,5,4,4,2,1  
DB 1,1,1,6,2,4,2,1  
DB 1,1,1,6,4,4,2,1  
DB 1,1,1,3,2,5,2,1  
DB 1,1,1,3,4,5,2,1  
DB 1,1,1,4,2,5,2,1  
DB 1,1,1,4,4,5,2,1  
DB 1,1,1,5,2,5,2,1  
DB 1,1,1,5,4,5,2,1  
DB 1,1,1,6,2,5,2,1  
DB 1,1,1,6,4,5,2,1  
DB 1,1,1,3,2,6,2,1  
DB 1,1,1,3,4,6,2,1  
DB 1,1,1,4,2,6,2,1  
DB 1,1,1,4,4,6,2,1

DB 1,1,1,5,2,6,2,1  
DB 1,1,1,5,4,6,2,1  
DB 1,1,1,6,2,6,2,1  
DB 1,1,1,6,4,6,2,1  
BC14 DB 1,1,1,1,1,1,1,1,1,1

AC15 DB 32

DB 1,1,10,3,2,3,2,1  
DB 1,1,10,3,4,3,2,1  
DB 1,1,10,4,2,3,2,1  
DB 1,1,10,4,4,3,2,1  
DB 1,1,10,5,2,3,2,1  
DB 1,1,10,5,4,3,2,1  
DB 1,1,10,6,2,3,2,1  
DB 1,1,10,6,4,3,2,1  
DB 1,1,10,3,2,4,2,1  
DB 1,1,10,3,4,4,2,1  
DB 1,1,10,4,2,4,2,1  
DB 1,1,10,4,4,4,2,1  
DB 1,1,10,5,2,4,2,1  
DB 1,1,10,5,4,4,2,1  
DB 1,1,10,6,2,4,2,1  
DB 1,1,10,6,4,4,2,1  
DB 1,1,10,3,2,5,2,1  
DB 1,1,10,3,4,5,2,1  
DB 1,1,10,4,2,5,2,1  
DB 1,1,10,4,4,5,2,1  
DB 1,1,10,5,2,5,2,1  
DB 1,1,10,5,4,5,2,1  
DB 1,1,10,6,2,5,2,1  
DB 1,1,10,6,4,5,2,1  
DB 1,1,10,3,2,6,2,1  
DB 1,1,10,3,4,6,2,1  
DB 1,1,10,4,2,6,2,1  
DB 1,1,10,4,4,6,2,1  
DB 1,1,10,5,2,6,2,1  
DB 1,1,10,5,4,6,2,1  
DB 1,1,10,6,2,6,2,1  
DB 1,1,10,6,4,6,2,1  
BC15 DB 1,1,1,40,40,40,40,1,1,40

AC16 DB 32

DB 2,1,9,3,2,3,3,1  
DB 2,1,9,3,4,3,3,1  
DB 2,1,9,4,2,3,3,1  
DB 2,1,9,4,4,3,3,1  
DB 2,1,9,5,2,3,3,1  
DB 2,1,9,5,4,3,3,1  
DB 2,1,9,6,2,3,3,1  
DB 2,1,9,6,4,3,3,1  
DB 2,1,9,3,2,4,3,1  
DB 2,1,9,3,4,4,3,1  
DB 2,1,9,4,2,4,3,1  
DB 2,1,9,4,4,4,3,1  
DB 2,1,9,5,2,4,3,1

DB 2,1,9,5,4,4,3,1  
DB 2,1,9,6,2,4,3,1  
DB 2,1,9,6,4,4,3,1  
DB 2,1,9,3,2,5,3,1  
DB 2,1,9,3,4,5,3,1  
DB 2,1,9,4,2,5,3,1  
DB 2,1,9,4,4,5,3,1  
DB 2,1,9,5,2,5,3,1  
DB 2,1,9,5,4,5,3,1  
DB 2,1,9,6,2,5,3,1  
DB 2,1,9,6,4,5,3,1  
DB 2,1,9,3,2,6,3,1  
DB 2,1,9,3,4,6,3,1  
DB 2,1,9,4,2,6,3,1  
DB 2,1,9,4,4,6,3,1  
DB 2,1,9,5,2,6,3,1  
DB 2,1,9,5,4,6,3,1  
DB 2,1,9,6,2,6,3,1  
DB 2,1,9,6,4,6,3,1  
BC16 DB 1,1,1,1,1

AC17 DB 32  
DB 4,1,9,3,2,3,3,1  
DB 4,1,9,3,4,3,3,1  
DB 4,1,9,4,2,3,3,1  
DB 4,1,9,4,4,3,3,1  
DB 4,1,9,5,2,3,3,1  
DB 4,1,9,5,4,3,3,1  
DB 4,1,9,6,2,3,3,1  
DB 4,1,9,6,4,3,3,1  
DB 4,1,9,3,2,4,3,1  
DB 4,1,9,3,4,4,3,1  
DB 4,1,9,4,2,4,3,1  
DB 4,1,9,4,4,4,3,1  
DB 4,1,9,5,2,4,3,1  
DB 4,1,9,5,4,4,3,1  
DB 4,1,9,6,2,4,3,1  
DB 4,1,9,6,4,4,3,1  
DB 4,1,9,3,2,5,3,1  
DB 4,1,9,3,4,5,3,1  
DB 4,1,9,4,2,5,3,1  
DB 4,1,9,4,4,5,3,1  
DB 4,1,9,5,2,5,3,1  
DB 4,1,9,5,4,5,3,1  
DB 4,1,9,6,2,5,3,1  
DB 4,1,9,6,4,5,3,1  
DB 4,1,9,3,2,6,3,1  
DB 4,1,9,3,4,6,3,1  
DB 4,1,9,4,2,6,3,1  
DB 4,1,9,4,4,6,3,1  
DB 4,1,9,5,2,6,3,1  
DB 4,1,9,5,4,6,3,1  
DB 4,1,9,6,2,6,3,1  
DB 4,1,9,6,4,6,3,1  
BC17 DB 1,1,1,1,1

AC18 DB 8

DB 2,1,9,3,2,1,3,1  
DB 2,1,9,3,4,1,3,1  
DB 2,1,9,4,2,1,3,1  
DB 2,1,9,4,4,1,3,1  
DB 2,1,9,5,2,1,3,1  
DB 2,1,9,5,4,1,3,1  
DB 2,1,9,6,2,1,3,1  
DB 2,1,9,6,4,1,3,1

BC18 DB 1,1,1,1

AC19 DB 8

DB 4,1,9,3,2,1,3,1  
DB 4,1,9,3,4,1,3,1  
DB 4,1,9,4,2,1,3,1  
DB 4,1,9,4,4,1,3,1  
DB 4,1,9,5,2,1,3,1  
DB 4,1,9,5,4,1,3,1  
DB 4,1,9,6,2,1,3,1  
DB 4,1,9,6,4,1,3,1

BC19 DB 1,1,1,1

AC20 DB 8

DB 2,1,9,1,2,3,3,1  
DB 2,1,9,1,4,3,3,1  
DB 2,1,9,1,2,4,3,1  
DB 2,1,9,1,4,4,3,1  
DB 2,1,9,1,2,5,3,1  
DB 2,1,9,1,4,5,3,1  
DB 2,1,9,1,2,6,3,1  
DB 2,1,9,1,4,6,3,1

BC20 DB 1,1,1,1

AC21 DB 8

DB 4,1,9,1,2,3,3,1  
DB 4,1,9,1,4,3,3,1  
DB 4,1,9,1,2,4,3,1  
DB 4,1,9,1,4,4,3,1  
DB 4,1,9,1,2,5,3,1  
DB 4,1,9,1,4,5,3,1  
DB 4,1,9,1,2,6,3,1  
DB 4,1,9,1,4,6,3,1

BC21 DB 1,1,1,1

AC22 DB 4

DB 2,1,9,3,1,1,3,1  
DB 2,1,9,4,1,1,3,1  
DB 2,1,9,5,1,1,3,1  
DB 2,1,9,6,1,1,3,1

BC22 DB 1,1

AC23 DB 4  
DB 4,1,9,3,1,1,3,1  
DB 4,1,9,4,1,1,3,1  
DB 4,1,9,5,1,1,3,1  
DB 4,1,9,6,1,1,3,1  
BC23 DB 1,1

AC24 DB 4  
DB 2,1,9,1,1,3,3,1  
DB 2,1,9,1,1,4,3,1  
DB 2,1,9,1,1,5,3,1  
DB 2,1,9,1,1,6,3,1  
BC24 DB 1,1

AC25 DB 4  
DB 4,1,9,1,1,3,3,1  
DB 4,1,9,1,1,4,3,1  
DB 4,1,9,1,1,5,3,1  
DB 4,1,9,1,1,6,3,1  
BC25 DB 1,1

AC26 DB 16  
DB 2,1,9,3,1,3,3,1  
DB 2,1,9,4,1,3,3,1  
DB 2,1,9,5,1,3,3,1  
DB 2,1,9,6,1,3,3,1  
DB 2,1,9,3,1,4,3,1  
DB 2,1,9,4,1,4,3,1  
DB 2,1,9,5,1,4,3,1  
DB 2,1,9,6,1,4,3,1  
DB 2,1,9,3,1,5,3,1  
DB 2,1,9,4,1,5,3,1  
DB 2,1,9,5,1,5,3,1  
DB 2,1,9,6,1,5,3,1  
DB 2,1,9,3,1,6,3,1  
DB 2,1,9,4,1,6,3,1  
DB 2,1,9,5,1,6,3,1  
DB 2,1,9,6,1,6,3,1  
BC26 DB 1,1,1

AC27 DB 16  
DB 4,1,9,3,1,3,3,1  
DB 4,1,9,4,1,3,3,1  
DB 4,1,9,5,1,3,3,1  
DB 4,1,9,6,1,3,3,1  
DB 4,1,9,3,1,4,3,1  
DB 4,1,9,4,1,4,3,1  
DB 4,1,9,5,1,4,3,1  
DB 4,1,9,6,1,4,3,1  
DB 4,1,9,3,1,5,3,1  
DB 4,1,9,4,1,5,3,1  
DB 4,1,9,5,1,5,3,1  
DB 4,1,9,6,1,5,3,1

```
DB 4,1,9,3,1,6,3,1
DB 4,1,9,4,1,6,3,1
DB 4,1,9,5,1,6,3,1
DB 4,1,9,6,1,6,3,1
BC27 DB 1,1,1
CONST ENDS
```

```
EXTRN CREATE_FILE:NEAR,WRITE_TO_FILE:NEAR,CLOSE_FILE:NEAR
EXTRN READ_FROM_FILE:NEAR,OPEN_FILE_READ:NEAR
EXTRN SET_RETURN_ADDRESS:NEAR,SET_CRITICAL_ADDRESS:NEAR
EXTRN LOAD_PROGRAM:NEAR
```

```
EXTRN SAVE_SCREEN:NEAR,RECOVER_SCREEN:NEAR,DOUBLE_DIS_LINE:NEAR
EXTRN CLS:NEAR,CURSER_POSITION:NEAR
```

```
EXTRN CLEAR_BUFFER_GET_KEY:NEAR,CLEAR_BUFFER:NEAR
```

```
CODE SEGMENT 'CODE' PUBLIC
ASSUME CS:CODE,DS:CONST,ES:NOTHING,SS:STACK

PUBLIC RUN_SIMULATE,CI_S1,RE_S1,POSTRUN_LINK
```

```
RUN_SIMULATE PROC NEAR
    MOV STACK_POINTER1,SP
    PUSH AX
    CALL FILE_PREPARE2
    MOV SI,OFFSET SBU2
    CALL SAVE_SCREEN
    CALL CLS
    MOV SI,OFFSET GDF
    CALL DOUBLE_DIS_LINE
    POP AX
    TEST AH,1
    JNZ RS1
    CALL CHECK_DEFINITION
    JC RS7
    CALL VALID_CONFIGURATION
    CALL LINK_PARA_FILE
    CALL CHECK_CONFIGURATION
    JC RS39
    MOV AX,OFFSET RS5
    JMP AX
```

```
RS39:
    XOR AL,AL
    MOV LOAD_STATUS,AL
    CALL CLS
    MOV SI,OFFSET LDB
    CALL DOUBLE_DIS_LINE
    CALL READ_STAR_BASE
    CALL CLS
    MOV SI,OFFSET GUS
    CALL DOUBLE_DIS_LINE
    CALL CODE_CONVERSION_ALL
    CALL STAR_GENERATOR
```

```

CALL INTEGER_DECLARE
CALL INTEGER_DECLARE1
CALL CONNECT_GENERATOR
CALL GENERATE_UNIVERSE
CALL CREATE_LIB
MOV LOAD_STATUS,1
MOV SI,OFFSET LSD
RS3:
  PUSH SI
  CALL CLS
  POP SI
  CALL LOAD_PROGRAM
RS7:
  MOV SI,OFFSET SBU2
  CALL RECOVER_SCREEN
  CALL CLEAR_BUFFER
  RET
RS1:
  CALL CHECK_N_DEFINITION
  JC RS7
  CALL NET_PARA_FILE
  MOV SI,OFFSET LSDS1
  CMP NET_CON,2
  JNZ RS24
  CMP MA_CON,1
  JNZ RS21
  MOV WORD PTR [SI+8],OFFSET AS8
  JMP RS22
RS21:
  CMP MA_CON,2
  JNZ RS23
  MOV WORD PTR [SI+8],OFFSET AS10
  JMP RS22
RS23:
  MOV WORD PTR [SI+8],OFFSET AS11
RS22:
  MOV SI,OFFSET LSDS
  JMP RS3
  RET
RS5:
  MOV SI,OFFSET LSDS1
  MOV WORD PTR [SI+8],OFFSET A8
  MOV SI,OFFSET LSDS
  JMP RS3
RS24:
  CMP SAF_TP,1
  JZ RS96
  CMP SAF_TP,2
  JZ RS97
  MOV WORD PTR [SI+8],OFFSET AS18
  JMP RS22
RS96:
  MOV WORD PTR [SI+8],OFFSET AS4
  JMP RS22
RS97:

```

```
MOV WORD PTR [SI+8],OFFSET A9
JMP RS22
RUN_SIMULATE ENDP
```

```
CI_S1 PROC NEAR
MOV STACK_POINTER1,SP
PUSH AX
CALL FILE_PREPARE2
MOV SI,OFFSET SBU2
CALL SAVE_SCREEN
MOV SI,OFFSET LSDS1
POP AX
CMP AH,2
JZ CIS5
CMP AH,3
JZ CIS3
MOV DX,OFFSET AS16
MOV AX,OFFSET A99
JMP AX
CIS5:
MOV WORD PTR [SI+8],OFFSET AS7
JMP CIS2
CIS3:
MOV WORD PTR [SI+8],OFFSET AS6
CIS2:
MOV SI,OFFSET LSDS
JMP RS3
RET
CI_S1 ENDP
```

```
POSTRUN_LINK PROC NEAR
MOV STACK_POINTER1,SP
PUSH AX
CALL FILE_PREPARE2
MOV SI,OFFSET SBU2
CALL SAVE_SCREEN
MOV SI,OFFSET LSDS1
POP AX
CMP AH,8
JZ PL1
CMP AH,2
JZ PL2
CMP AH,4
JZ PL3
CMP AH,3
JZ PL7
CMP AH,5
JZ PL8
MOV WORD PTR [SI+8],OFFSET AS5
JMP PL4
PL3:
MOV WORD PTR [SI+8],OFFSET AS15
JMP PL4
PL2:
MOV WORD PTR [SI+8],OFFSET AS14
```

```

    JMP PL4
PL1:
    MOV WORD PTR [SI+8],OFFSET AS13
PL4:
    MOV AX,OFFSET CIS2
    JMP AX
PL7:
    MOV WORD PTR [SI+8],OFFSET AS19
    JMP PL4
PL8:
    MOV WORD PTR [SI+8],OFFSET AS20
    JMP PL4
POSTRUN_LINK ENDP

```

```

RE_S1 PROC NEAR
    MOV STACK_POINTER1,SP
    AND AH,0FH
    PUSH AX
    CALL FILE_PREPARE2
    MOV SI,OFFSET SBU2
    CALL SAVE_SCREEN
    POP AX
    CMP AH,1
    JZ A98
    CMP AH,2
    JZ A59
    MOV DX,OFFSET DATA_FILE_NAME1
    JMP A99
A98:
    MOV DX,OFFSET DATA_FILE_NAME2
    JMP A99
A59:
    MOV DX,OFFSET DATA_FILE_NAME3
A99:
    CALL OPEN_FILE_READ
    MOV FILE_HANDLE,AX
    MOV DI,OFFSET CONNECT_PROGRAM

    XOR CX,CX
    MOV NUMBER_OF_BYTE,CX
A12:
    CMP NUMBER_OF_BYTE,0
    JZ A11
    DEC NUMBER_OF_BYTE
    MOVSF
    CMP WORD PTR [SI-1],0A0DH
    JZ A122
    CMP WORD PTR [SI-1],0DOAH
    JZ A121
    CMP BYTE PTR [SI-1],0DH
    JZ A56
    CMP BYTE PTR [SI-1],0AH
    JNZ A12
    MOV BYTE PTR ES:[DI-1],0DH
    JMP A56

```

```

A122:
  MOV BYTE PTR ES:[DI-1],0DH
  JMP A561
A121:
  MOV BYTE PTR ES:[DI],0DH
A561:
  CMP NUMBER_OF_BYTE,0
  JZ A56
  INC SI
  DEC NUMBER_OF_BYTE
A56:
  INC CX
  CMP CX,24
  JNZ A12
  CALL DISPLAY_A_PAGE
  MOV DI,OFFSET CONNECT_PROGRAM
  XOR CX,CX
  JMP A12
A11:
  PUSH CX
  MOV CX,4000
  MOV BX,FILE_HANDLE
  MOV DX,OFFSET SBU
  CALL READ_FROM_FILE
  MOV NUMBER_OF_BYTE,AX
  CMP AX,0
  POP CX
  JZ A14
  MOV SI,OFFSET SBU
  JMP A12
A14:
  CMP CX,0
  JZ A15
  CALL DISPLAY_A_PAGE
A15:

  MOV BX,FILE_HANDLE
  CALL CLOSE_FILE
  MOV AX,OFFSET RS7
  JMP AX

```

```

DISPLAY_A_PAGE:
  PUSH SI
  PUSH CX
  CALL CLS
  POP CX
  XOR DX,DX
  MOV SI,OFFSET CONNECT_PROGRAM
A13:
  PUSH CX
  PUSH DX
  CALL CURSER_POSITION
  CALL DOUBLE_DIS_LINE
  CMP BYTE PTR DS:[SI],0DH

```

```

JNZ A57
INC SI
A57:
CMP BYTE PTR DS:[SI],0AH
JNZ A58
INC SI
A58:
POP DX
INC DH
POP CX
LOOP A13
PUSH CX
CALL CLEAR_BUFFER_GET_KEY
POP CX
POP SI
RET
RE_S1 ENDP

```

```

CHECK_DEFINITION PROC NEAR
MOV SI,OFFSET SOU_CON
MOV CX,8
CDP1:
CMP BYTE PTR [SI],0
JZ CDP2
INC SI
LOOP CDP1
CLC
RET
CDP2:
CALL CLS
MOV SI,OFFSET INCOM_DEF
CALL DOUBLE_DIS_LINE
CALL CLEAR_BUFFER_GET_KEY
STC
RET
CHECK_DEFINITION ENDP

```

```

CHECK_N_DEFINITION PROC NEAR
CMP NET_CON,0
JZ CCFN1
CMP NET_CON,1
JZ CCFN2
CMP MA_CON,0
JZ CCFN1
CLC
RET
CCFN2:
CMP SAF_TP,0
JZ CCFN1
CMP SAF_RTG,0
JZ CCFN1
CLC
RET
CCFN1:
CALL CLS

```

```
MOV SI,OFFSET INCOM_N_DEF
CALL DOUBLE_DIS_LINE
CALL CLEAR_BUFFER_GET_KEY
STC
RET
CHECK_N_DEFINITION ENDP
```

```
CHECK_CONFIGURATION PROC NEAR
MOV SI,OFFSET SOU_CON
MOV DI,OFFSET O_LINK_NET
MOV CX,4
CC2:
MOV AX,[SI]
CMP AX,[DI]
JNZ CC1
INC SI
INC SI
INC DI
INC DI
LOOP CC2
TEST LOAD_STATUS,1
JZ CC3
CLC
RET
CC1:
MOV SI,OFFSET SOU_CON
MOV DI,OFFSET O_LINK_NET
MOV CX,4
REP MOVSW
CC3:
STC
RET
CHECK_CONFIGURATION ENDP
```

```
VALID_CONFIGURATION PROC NEAR
MOV SI,OFFSET VALID_TABLE
MOV CL,[SI]
XOR CH,CH
INC SI
VCP1:
PUSH CX
MOV DI,[SI]
MOV CL,[DI]
XOR CH,CH
INC DI
VCP2:
CALL CHECK_AN_ITEM
JAE VCP3
ADD DI,8
LOOP VCP2
ADD SI,4
POP CX
```

LOOP VCP1

CALL CLS  
MOV SI,OFFSET INVAL\_DEF  
CALL DOUBLE\_DIS\_LINE  
CALL CLEAR\_BUFFER\_GET\_KEY  
MOV SP,STACK\_POINTER1  
MOV AX,OFFSET RS7  
JMP AX

VCP3:

POP CX  
MOV SI,[SI+2]  
MOV FIFO\_POINTER,SI  
RET

CHECK\_AN\_ITEM:

MOV AL,SOU\_CON  
CMP AL,[DI]  
JNZ VCP4  
MOV AL,ECD\_CON  
CMP AL,[DI+1]  
JNZ VCP4  
MOV AL,MOD\_CON  
CMP AL,[DI+2]  
JNZ VCP4  
MOV AL,CF\_CON  
CMP AL,[DI+3]  
JNZ VCP4  
MOV AL,CHL\_CON  
CMP AL,[DI+4]  
JNZ VCP4  
MOV AL,RF\_CON  
CMP AL,[DI+5]  
JNZ VCP4  
MOV AL,DEMOD\_CON  
CMP AL,[DI+6]  
JNZ VCP4  
MOV AL,DCD\_CON  
CMP AL,[DI+7]  
JNZ VCP4  
CLC  
RET

VCP4:

STC  
RET

VALID\_CONFIGURATION ENDP

CREATE\_LIB PROC NEAR

MOV DI,OFFSET LIB\_STRING  
MOV SI,OFFSET SOU\_FL  
MOV AL,SOU\_CON  
CALL ADD\_LIB  
MOV SI,OFFSET MOD\_FL

```

MOV AL,MOD_CON
CALL ADD_LIB

CMP SOU_CON,1
JNZ CLP102
MOV SI,OFFSET ECD_FL
MOV AL,ECD_CON
CALL ADD_LIB
CLP102:

CMP MOD_CON,9
JNZ CLP91
CMP SOU_CON,1
JZ CLP91
MOV AL,MOD_CON
CLP91:
MOV SI,OFFSET DIS_FL
CALL ADD_LIB

MOV SI,OFFSET CF_FL
MOV AL,CF_CON
CALL ADD_LIB
MOV SI,OFFSET RF_FL
MOV AL,RF_CON
CALL ADD_LIB

CMP SOU_CON,1
JNZ CLP103
MOV SI,OFFSET DCD_FL
MOV AL,DCD_CON
CMP DCD_CON,4
JZ CLP2
CMP DCD_CON,1
JZ CLP2
CMP MOD_CON,2
JNZ CLP3
CMP DCD_CON,3
JZ CLP3
ADD AL,ECD_CON
INC AL
JMP CLP2
CLP3:
MOV AL,ECD_CON
CLP2:
CALL ADD_LIB
CLP103:

MOV SI,OFFSET DEMOD_FL
CALL ADJUST_DEMOD
CALL ADD_LIB

MOV SI,OFFSET CHL_FL
MOV AL,CHL_CON
CMP AL,3
JNZ CLP4

```

```

    ADD AL,ECD_CON
    ADD AL,3
CLP4:
    CALL ADD_LIB

    ADD DX,A6_LENGTH
    MOV SI,OFFSET A7
    MOV CX,A7_LENGTH
    REP MOVSB
    MOV SI,OFFSET A6
    MOV [SI],DL
    RET

ADJUST_DEMOD:
    MOV AL,DEMOD_CON
    CMP AL,2
    JNZ CLP9
    ADD AL,MOD_CON
    INC AL
CLP9:
    RET
CREATE_LIB ENDP

ADD_LIB PROC NEAR
    MOV CL,[SI]
    XOR CH,CH
    INC SI
ALP1:
    CMP AL,[SI]
    JZ ALP2
    ADD SI,3
    LOOP ALP1
    RET
ALP2:
    MOV SI,[SI+1]
    MOV CL,[SI]
    INC SI
    ADD DX,CX
    REP MOVSB
    RET
ADD_LIB ENDP

GENERATE_UNIVERSE PROC NEAR
    MOV DX,OFFSET UNIVERSE_Q
    CALL CREATE_FILE
    MOV FILE_HANDLE,AX
    MOV CX,INT_COUNT
    MOV BX,FILE_HANDLE
    MOV DX,OFFSET DECLARE_INT
    CALL WRITE_TO_FILE
    MOV CX,15
    MOV BX,FILE_HANDLE
    MOV DX,OFFSET C_OVERHEAD
    CALL WRITE_TO_FILE

```

```

MOV CX,INT_COUNT1
MOV BX,FILE_HANDLE
MOV DX,OFFSET DECLARE_INT1
CALL WRITE_TO_FILE
MOV CX,STAR_COUNT
MOV BX,FILE_HANDLE
MOV DX,OFFSET STAR_PROGRAM
CALL WRITE_TO_FILE
MOV CX,CONNECT_COUNT
MOV BX,FILE_HANDLE
MOV DX,OFFSET CONNECT_PROGRAM
CALL WRITE_TO_FILE
MOV CX,15
MOV BX,FILE_HANDLE
MOV DX,OFFSET C_END
CALL WRITE_TO_FILE
MOV BX,FILE_HANDLE
CALL CLOSE_FILE
RET

```

GENERATE\_UNIVERSE ENDP

STAR\_GENERATOR PROC NEAR

```

XOR AL,AL
XOR AH,AH
MOV ERROR_COUNT1,AX
MOV AH,SOU_CON
XOR DX,DX
MOV SI,OFFSET SBU
MOV DI,OFFSET STAR_PROGRAM
MOV BX,OFFSET STAR_POINTER
CALL ONE_LINE_STAR

```

```

CMP MOD_CON,9
JNZ SGP97
CMP CHL_CON,3
JNZ SGP98

```

SGP97:

```

MOV AL,1
MOV AH,ECD_CON
CALL ONE_LINE_STAR
CALL ERROR_PATH1

```

SGP98:

```

CMP MOD_CON,9
JZ SGP95
MOV AL,2
MOV AH,MOD_CON
CALL ONE_LINE_STAR

```

SGP95:

```

MOV AL,3
MOV AH,CF_CON
CALL ONE_LINE_STAR
MOV AL,4
MOV AH,CHL_CON

```

```
CALL ONE_LINE_STAR
MOV AL,5
MOV AH,RF_CON
CALL ONE_LINE_STAR
```

```
CMP MOD_CON,9
JZ SGP96
CALL ADJUST_DEMOD
MOV AH,AL
MOV AL,6
CALL ONE_LINE_STAR
SGP96:
```

```
CMP MOD_CON,9
JNZ SGP997
CMP CHL_CON,3
JNZ SGP998
SGP997:
MOV AL,7
MOV AH,DCD_CON
CALL ONE_LINE_STAR
SGP998:
```

```
MOV AL,8
MOV AH,1
CMP MOD_CON,9
JNZ SGP94
CMP SOU_CON,1
JZ SGP94
MOV AH,MOD_CON
```

```
SGP94:
CALL ONE_LINE_STAR
```

```
CMP CHL_CON,1
JZ SGP1
CMP CHL_CON,3
JZ SGP1
MOV CX,STAR_NOISE_LENGTH
ADD DX,STAR_NOISE_LENGTH
MOV SI,OFFSET STAR_NOISE
REP MOVSB
```

```
SGP1:
MOV BYTE PTR ES:[DI],0
MOV ES:STAR_COUNT,DX
RET
```

```
STAR_GENERATOR ENDP
```

```
ERROR_PATH1 PROC NEAR
```

```
PUSH SI
PUSH DI
MOV SI,DI
DEC SI
DEC SI
```

```
EP12:
DEC SI
```

```

CMP SI,OFFSET STAR_PROGRAM
JZ EP19
CMP BYTE PTR ES:[SI],0AH
JNZ EP12
ADD SI,6
JMP EP18
EP19:
ADD SI,5
EP18:
MOV DI,OFFSET ERROR_PROGRAM1
PUSH SI
MOV CX,ERROR_PC3
MOV SI,OFFSET ERROR_P3
MOV ERROR_COUNT1,CX
REP MOVSB
POP SI
INC ERROR_COUNT1
MOVSB
EP13:
MOVSB
INC ERROR_COUNT1
CMP BYTE PTR DS:[SI],22H
JNZ EP13
MOVSB
INC ERROR_COUNT1
MOV CX,ERROR_PC1
ADD ERROR_COUNT1,CX
MOV SI,OFFSET ERROR_P1
REP MOVSB
PUSH DX
XOR DX,DX
CALL ADD_FIFO
ADD ERROR_COUNT1,DX
POP DX
MOV CX,ERROR_PC2
ADD ERROR_COUNT1,CX
MOV SI,OFFSET ERROR_P2
REP MOVSB
POP DI
POP SI
RET
ERROR_PATH1 ENDP

ONE_LINE_STAR PROC NEAR
OLS4:
CMP [SI],AL
JZ OLS2
JAE OLS15
OLS7:
CMP WORD PTR [SI],0A0DH
JZ OLS16
INC SI
JMP OLS7
OLS16:
INC SI

```

```

    INC SI
    CMP WORD PTR [SI],"$$"
    JNZ OLS4
    JMP OLS14
OLS15:
    MOV WORD PTR [BX],OFFFH
    INC BX
    INC BX
    CMP WORD PTR [SI],"$$"
    JNZ OLS98
OLS14:
    STC
    RET
OLS98:
    CLC
    RET
OLS2:
    INC SI
    CMP BYTE PTR [SI]," "
    JZ OLS2
    CMP [SI],AH
    JNZ OLS3
OLS5:
    INC SI
    CMP BYTE PTR [SI]," "
    JZ OLS5
    MOV [BX],DI
    INC BX
    INC BX
OLS6:
    INC DX
    MOVSB
    CMP WORD PTR [SI],0A0DH
    JNZ OLS6
    INC DX
    INC DX
    MOVSW
OLS12:
    CMP [SI],AL
    JNZ OLS9
OLS11:
    INC SI
    CMP WORD PTR [SI],0A0DH
    JNZ OLS11
    INC SI
    INC SI
    JMP OLS12
OLS9:
    CMP WORD PTR [SI],"$$"
    JNZ OLS10
    STC
    RET
OLS10:
    CLC
    RET

```

```

OLS3:
  INC SI
  CMP WORD PTR [SI],0A0DH
  JNZ OLS3
  INC SI
  INC SI
  JMP OLS4
ONE_LINE_STAR ENDP

READ_STAR_BASE PROC NEAR
  CALL FILE_PREPARE2
  MOV DX,OFFSET STAR_BASE_DAT
  CALL OPEN_FILE_READ
  MOV FILE_HANDLE,AX
  MOV CX,4000
  MOV BX,FILE_HANDLE
  MOV DX,OFFSET SBU
  CALL READ_FROM_FILE
  MOV BX,FILE_HANDLE
  CALL CLOSE_FILE
  RET
READ_STAR_BASE ENDP

INTEGER_DECLARE PROC NEAR
  MOV SI,OFFSET C1+1
  MOV DL,[SI-1]
  XOR DH,DH
  MOV DI,OFFSET DECLARE_INT
  MOV CX,DX
  REP MOVSB
  MOV SI,OFFSET STAR_PROGRAM
ID2:
  CMP WORD PTR [SI],"ts"
  JNZ ID1
  CMP WORD PTR [SI+2],"ra"
  JNZ ID1

  ID3:
    INC SI
    CMP BYTE PTR [SI],0DH
    JZ ID1
    CMP BYTE PTR [SI],"&"
    JNZ ID3
    MOV WORD PTR ES:[DI],"ni"
    MOV WORD PTR ES:[DI+2]," t"
    ADD DI,4
    ADD DX,4
  ID4:
    INC SI
    CMP BYTE PTR [SI]," ,"
    JZ ID5
    MOV AL,[SI]
    MOV ES:[DI],AL
    INC DI
    INC DX

```

```

        JMP ID4
ID5:
        CALL REPLACE_STRING
        MOV BYTE PTR ES:[DI],";"
        MOV WORD PTR ES:[DI+1],0A0DH
        ADD DI,3
        ADD DX,3
ID6:
        INC SI
        CMP BYTE PTR [SI],0DH
        JNZ ID6

ID1:
        INC SI
        CMP BYTE PTR [SI],0
        JZ ID10
        JMP ID2
ID10:
        MOV ES:INT_COUNT,DX
        RET
INTEGER_DECLARE ENDP

REPLACE_STRING PROC NEAR
        PUSH SI
        MOV SI,OFFSET D1
        XOR CH,CH
        MOV CL,[SI]
        INC SI
SP1:
        PUSH SI
        PUSH DI
        PUSH CX
        XOR CH,CH
        MOV CL,[SI]
        MOV SI,[SI+1]
SP3:
        DEC DI
        MOV AL,[SI]
        CMP AL,ES:[DI]
        JNZ SP2
        INC SI
        LOOP SP3
        POP CX
        POP DI
        POP SI
        JMP SP4
SP2:
        POP CX
        POP DI
        POP SI
        ADD SI,3
        LOOP SP1
        POP SI
        RET
SP4:

```

```

MOV CL,[SI]
XOR CH,CH
ADD CX,7
SUB DI,CX
SUB DX,CX
POP SI
RET
REPLACE_STRING ENDP

```

```

INTEGER_DECLARE1 PROC NEAR

```

```

XOR DX,DX
MOV SI,OFFSET STAR_PROGRAM
MOV DI,OFFSET DECLARE_INT1

```

```

ID22:
CMP WORD PTR [SI],"ts"
JNZ ID21
CMP WORD PTR [SI+2],"ra"
JNZ ID21
XOR AH,AH

```

```

ID23:
INC SI
CMP BYTE PTR [SI],0DH
JZ ID21
CMP WORD PTR [SI],2C22H
JNZ ID23
CMP AH,1
JZ ID23
MOV AH,1
MOV WORD PTR ES:[DI],"ni"
MOV WORD PTR ES:[DI+2]," t"
ADD DI,4
ADD DX,4
INC SI

```

```

ID24:
INC SI
CMP BYTE PTR [SI]," "
JZ ID25
MOV AL,[SI]
MOV ES:[DI],AL
INC DI
INC DX
JMP ID24

```

```

ID25:
MOV WORD PTR ES:[DI]," ("
MOV BYTE PTR ES:[DI+2],";"
MOV WORD PTR ES:[DI+3],0A0DH
ADD DI,5
ADD DX,5

```

```

ID26:
INC SI
CMP BYTE PTR [SI],0DH
JNZ ID26

```

```

ID21:

```

```

    INC SI
    CMP BYTE PTR [SI],0
    JZ ID30
    JMP ID22
ID30:
    MOV SI,OFFSET C2+2
    MOV CX,[SI-2]
    ADD DX,CX
    REP MOVSB
    MOV ES:INT_COUNT1,DX
    RET
INTEGER_DECLARE1 ENDP

CONNECT_GENERATOR PROC NEAR
    XOR DX,DX
    XOR AH,AH
    MOV SI,OFFSET STAR_PROGRAM
    MOV DI,OFFSET CONNECT_PROGRAM
CG1:
    CMP WORD PTR [SI],"ts"
    JNZ CG2
    CMP WORD PTR [SI+2],"ra"
    JNZ CG2

CG3:
    INC SI
    CMP BYTE PTR [SI],0DH
    JZ CG2
    CMP BYTE PTR [SI],22H
    JNZ CG3
    CMP AH,0
    JZ CG9
    MOV WORD PTR ES:[DI],"0"
    MOV BYTE PTR ES:[DI+2],22H
    ADD DI,3
    ADD DX,3
    PUSH SI
CG8:
    INC SI
    CMP BYTE PTR [SI]," "
    JZ CG7
    MOV AL,[SI]
    MOV ES:[DI],AL
    INC DI
    INC DX
    JMP CG8
CG7:
    CALL ADD_SECOND_ENTRY
    POP SI
CG9:
    CALL ADD_FIRST_ENTRY
CG4:
    INC SI
    CMP BYTE PTR [SI]," "
    JZ CG5

```

```

    MOV AL,[SI]
    MOV ES:[DI],AL
    INC DI
    INC DX
    JMP CG4
CG5:
    MOV BYTE PTR ES:[DI]," "
    INC DI
    INC DX
CG6:
    INC SI
    CMP BYTE PTR [SI],0DH
    JNZ CG6
CG2:
    INC SI
    CMP BYTE PTR [SI],0
    JZ CG10
    CMP WORD PTR ES:[DI-4],"ya"
    JNZ CG191
    CMP WORD PTR ES:[DI-6],"lp"
    JNZ CG191
    CMP WORD PTR ES:[DI-8],"si"
    JNZ CG191
    CMP BYTE PTR ES:[DI-9],"D"
    JZ CG10
CG191:
    CMP WORD PTR ES:[DI-4],"kn"
    JNZ CG192
    CMP WORD PTR ES:[DI-6],"is"
    JZ CG10
CG192:
    MOV CX,OFFSET CG1
    JMP CX
CG10:
    DEC DI
    DEC DX
    CMP BYTE PTR ES:[DI],0AH
    JNZ CG10
    INC DI
    INC DX

    CMP CHL_CON,2
    JZ CG9098
    CMP CHL_CON,4
    JNZ CG901
CG9098:
    MOV CX,ADDER_LENGTH
    ADD DX,ADDER_LENGTH
    MOV SI,OFFSET ADDER_LINE
    REP MOVSB
    CALL ADD_FIFO
    MOV CX,ADDER_LENGTH1
    ADD DX,ADDER_LENGTH1
    MOV SI,OFFSET ADDER_LINE1
    REP MOVSB

```

CG901:

```
CMP ERROR_COUNT1,0
JZ CG98
ADD DX,ERROR_COUNT1
MOV CX,ERROR_COUNT1
MOV SI,OFFSET ERROR_PROGRAM1
REP MOVSB
```

CG98:

```
MOV ES:CONNECT_COUNT,DX
RET
```

ADD\_FIRST\_ENTRY:

```
MOV AH,1
MOV WORD PTR ES:[DI],"oc"
MOV WORD PTR ES:[DI+2],"nn"
MOV WORD PTR ES:[DI+4],"ce"
MOV WORD PTR ES:[DI+6],"(t"
MOV BYTE PTR ES:[DI+8],22H
ADD DI,9
ADD DX,9
RET
```

ADD\_SECOND\_ENTRY:

```
MOV WORD PTR ES:[DI],"0,"
MOV BYTE PTR ES:[DI+2]," ,"
ADD DI,3
ADD DX,3
```

CALL ADD\_FIFO

```
MOV WORD PTR ES:[DI],"F*"
MOV WORD PTR ES:[DI+2],"FI"
MOV BYTE PTR ES:[DI+4],"O"
MOV WORD PTR ES:[DI+5],";"
MOV WORD PTR ES:[DI+7],0A0DH
ADD DI,9
ADD DX,9
RET
```

ADD\_FIFO:

```
MOV SI,FIFO_POINTER
XOR AH,AH
MOV AL,[SI]
INC FIFO_POINTER
CMP AL,10
JC FIFO1
MOV BL,10
DIV BL
CALL ADD_ONE_DIGIT
MOV AL,AH
```

FIFO1:

```
CALL ADD_ONE_DIGIT
RET
```

```
ADD_ONE_DIGIT:
ADD AL,30H
MOV ES:[DI],AL
INC DI
INC DX
RET
```

```
CONNECT_GENERATOR ENDP
```

```
CODE_CONVERSION_ALL PROC NEAR
```

```
MOV SI,OFFSET SBU
MOV DL,10
```

```
PAS5:
```

```
CALL CODE_CONVERSION
CALL CODE_CONVERSION
```

```
PAS3:
```

```
CMP WORD PTR [SI],0A0DH
JZ PAS2
INC SI
JMP PAS3
```

```
PAS2:
```

```
CMP WORD PTR [SI+2],"$$"
JNZ PAS4
```

```
RET
```

```
PAS4:
```

```
INC SI
INC SI
JMP PAS5
```

```
CODE_CONVERSION_ALL ENDP
```

```
CODE_CONVERSION PROC NEAR
```

```
XOR AH,AH
MOV AL,[SI]
SUB AL,30H
CMP BYTE PTR [SI+1]," "
JZ PAS1
MUL DL
SUB BYTE PTR [SI+1],30H
ADD AL,[SI+1]
MOV BYTE PTR [SI+1]," "
MOV [SI],AL
INC SI
INC SI
INC SI
RET
```

```
PAS1:
```

```
MOV [SI],AL
INC SI
INC SI
RET
```

```
CODE_CONVERSION ENDP
```

LINK\_PARA\_FILE PROC NEAR

CALL CREATE\_STOP  
CALL CREATE\_SOU  
CALL CREATE\_CF  
CALL CREATE\_RF  
CALL CREATE\_CHL  
CALL CREATE\_COD  
CALL CREATE\_MOD  
RET

LINK\_PARA\_FILE ENDP

CREATE\_COD PROC NEAR

MOV DI,OFFSET SBU  
CMP ECD\_CON,1  
JZ CCOD4  
CMP ECD\_CON,2  
JZ CCOD1  
CMP ECD\_CON,3  
JZ CCOD1  
CMP ECD\_CON,5  
JZ CCOD2  
RET

CCOD4:

MOV WORD PTR ES:[DI], "1+"  
MOV WORD PTR ES:[DI+2], "0."  
MOV WORD PTR ES:[DI+4], "+E"  
MOV BYTE PTR ES:[DI+6], "0"  
MOV WORD PTR ES:[DI+7], 0A0DH  
MOV AX,9  
JMP CCOD3

CCOD2:

MOV WORD PTR ES:[DI], "0+"  
MOV WORD PTR ES:[DI+2], "6."  
MOV WORD PTR ES:[DI+4], "+E"  
MOV BYTE PTR ES:[DI+6], "0"  
MOV WORD PTR ES:[DI+7], 0A0DH  
MOV AX,9  
JMP CCOD3

CCOD1:

MOV WORD PTR ES:[DI], "0+"  
MOV WORD PTR ES:[DI+2], "5."  
MOV WORD PTR ES:[DI+4], "+E"  
MOV BYTE PTR ES:[DI+6], "0"  
MOV WORD PTR ES:[DI+7], 0A0DH  
MOV AX,9

CCOD3:

PUSH AX  
MOV DX,OFFSET COD\_TDT  
CALL CREATE\_FILE  
MOV FILE\_HANDLE,AX  
POP AX  
CALL FILE\_PREPARE3  
RET

CREATE\_COD ENDP

```

CREATE_MOD PROC NEAR
    MOV SI,OFFSET M_ARY
    MOV CL,[SI]
    XOR CH,CH
    MOV AL,MOD_CON
    CMP AL,9
    JNZ CMOD3
    CMP CHL_CON,3
    JNZ CMOD3
    MOV AL,20
CMOD3:
    INC SI
CMOD1:
    CMP AL,[SI]
    JZ CMOD2
    INC SI
    INC SI
LOOP CMOD1
    RET
CMOD2:
    XOR AX,AX
    MOV DI,OFFSET SBU
    MOV BL,[SI+1]
    CALL TRANSFER_DATA1
    XOR BL,BL
    CMP MOD_CON,10
    JNZ CMOD32
    MOV BL,1
CMOD32:
    CALL TRANSFER_DATA1

    PUSH AX
    MOV DX,OFFSET MOD_TDT
    CALL CREATE_FILE
    MOV FILE_HANDLE,AX
    POP AX
    CALL FILE_PREPARE3
    RET
CREATE_MOD ENDP

CREATE_STOP PROC NEAR
    XOR AX,AX
    MOV SI,OFFSET STOP_TDT1
    XOR BL,BL
    MOV DI,OFFSET SBU
    CALL TRANSFER_DATA
    PUSH AX
    MOV DX,OFFSET STOP_TDT
    CALL CREATE_FILE
    MOV FILE_HANDLE,AX
    POP AX
    CALL FILE_PREPARE3
    RET
CREATE_STOP ENDP

```

```
CREATE_SOU PROC NEAR
    MOV DX,OFFSET SOU_CON_TDT
    CALL CREATE_FILE
    MOV FILE_HANDLE,AX
```

```
    XOR AX,AX
    MOV DI,OFFSET SBU
    MOV BL,SOU_CON
    DEC BL
    CALL TRANSFER_DATA1
    CALL FILE_PREPARE3
```

```
    XOR AX,AX
    MOV SI,OFFSET SOU_TDT1
    MOV BL,SOU_CON
    MOV DI,OFFSET SBU
    CALL TRANSFER_DATA
    JAE CSU1
    RET
```

```
CSU1:
```

```
    PUSH AX
```

```
    MOV DX,OFFSET SOU_TDT
    CALL CREATE_FILE
    MOV FILE_HANDLE,AX
```

```
    POP AX
    CALL FILE_PREPARE3
    RET
```

```
CREATE_SOU ENDP
```

```
FILTER_TYPE DB 0
```

```
CREATE_CF PROC NEAR
```

```
    XOR AX,AX
    MOV DI,OFFSET SBU
    CMP CF_CON,0
    JZ CCF2
    MOV BL,CF_CON
    SUB BL,3
    XOR BH,BH
    MOV SI,OFFSET CF_BW_T
    MOV BL,[SI+BX]
    MOV CS:FILTER_TYPE,BL
    CALL TRANSFER_DATA1
```

```
CCF2:
```

```
    CMP FILTER_TYPE,2
    JZ CCF3
    MOV SI,OFFSET CF_TDT2
    MOV BL,CF_CON
    CALL TRANSFER_DATA
    JAE CCF1
    RET
```

```
CCF1:
```

```
    MOV BL,CF_CON
```

```
MOV SI,OFFSET CF_TDT1
CALL TRANSFER_DATA
MOV BL,CF_CON
MOV SI,OFFSET CF_TDT3
CALL TRANSFER_DATA
PUSH AX
```

```
MOV DX,OFFSET CF_TDTA
CALL CREATE_FILE
MOV FILE_HANDLE,AX
```

```
POP AX
CALL FILE_PREPARE3
RET
```

CCF3:

```
PUSH AX
MOV DX,OFFSET CF_TDTA
CALL CREATE_FILE
MOV FILE_HANDLE,AX
POP AX
CALL FILE_PREPARE3
```

```
XOR AX,AX
MOV DI,OFFSET SBU
MOV BL,CF_CON
MOV SI,OFFSET CF_TDT4
CALL TRANSFER_DATA
JAE CCF5
RET
```

CCF5:

```
PUSH AX
MOV DX,OFFSET CF_TDTB
CALL CREATE_FILE
MOV FILE_HANDLE,AX
POP AX
CALL FILE_PREPARE3
RET
```

CREATE\_CF ENDP

CREATE\_RF PROC NEAR

```
XOR AX,AX
MOV DI,OFFSET SBU
CMP RF_CON,0
JZ CRF2
MOV BL,RF_CON
SUB BL,3
XOR BH,BH
MOV SI,OFFSET CF_BW_T+4
MOV AL,[SI+BX]
MOV CS:FILTER_TYPE,AL
ADD AL,30H
MOV BYTE PTR ES:[DI], "+"
MOV ES:[DI+1],AL
MOV WORD PTR ES:[DI+2],0A0DH
```

```

    ADD DI,4
    MOV AX,4
CRF2:
    CMP FILTER_TYPE,2
    JZ CRF3
    MOV SI,OFFSET RF_TDT2
    MOV BL,RF_CON
    CALL TRANSFER_DATA
    JAE CRF1
    RET
CRF1:
    MOV BL,RF_CON
    MOV SI,OFFSET RF_TDT1
    CALL TRANSFER_DATA
    MOV BL,RF_CON
    MOV SI,OFFSET RF_TDT3
    CALL TRANSFER_DATA
    PUSH AX

    MOV DX,OFFSET RF_TDTA
    CALL CREATE_FILE
    MOV FILE_HANDLE,AX

    POP AX
    CALL FILE_PREPARE3
    RET

CRF3:
    PUSH AX
    MOV DX,OFFSET RF_TDTA
    CALL CREATE_FILE
    MOV FILE_HANDLE,AX
    POP AX
    CALL FILE_PREPARE3

    XOR AX,AX
    MOV DI,OFFSET SBU
    MOV BL,RF_CON
    MOV SI,OFFSET RF_TDT4
    CALL TRANSFER_DATA
    JAE CRF5
    RET
CRF5:
    PUSH AX
    MOV DX,OFFSET RF_TDTB
    CALL CREATE_FILE
    MOV FILE_HANDLE,AX
    POP AX
    CALL FILE_PREPARE3
    RET
CREATE_RF ENDP

CREATE_CHL PROC NEAR
    XOR AX,AX
    MOV DI,OFFSET SBU

```

```

    CMP CHL_CON,4
    JNZ CCHL2
    MOV AL,PM_CON
    ADD AL,30H
    MOV BYTE PTR ES:[DI], "+"
    MOV ES:[DI+1],AL
    MOV WORD PTR ES:[DI+2],0A0DH
    ADD DI,4
    MOV AX,4
CCHL2:
    MOV SI,OFFSET CHL_TDT1
    MOV BL,CHL_CON
    CALL TRANSFER_DATA
    JAE CCHL1
    RET
CCHL1:
    PUSH AX

    MOV DX,OFFSET CHL_TDT
    CALL CREATE_FILE
    MOV FILE_HANDLE,AX

    POP AX
    CALL FILE_PREPARE3
    RET
CREATE_CHL ENDP

TRANSFER_DATA PROC NEAR
    XOR CH,CH
    MOV CL,[SI]
    INC SI
TD2:
    CMP BL,[SI]
    JZ TD1
    ADD SI,7
LOOP TD2
    STC
    RET
TD1:
    PUSH SI
    MOV CX,[SI+3]
    MOV BX,[SI+5]
    MOV SI,[SI+1]
TD15:
    PUSH BX
    CMP BYTE PTR [SI],0DH
    JNZ TD3
    CALL NUMBER_END
    ADD SI,BX
    MOV BX,OFFSET TD16
    JMP BX
TD3:
    CMP BYTE PTR [SI], "-"
    JNZ TD4
    MOV BYTE PTR ES:[DI], "-"

```

```

JMP TD5
TD4:
MOV BYTE PTR ES:[DI], "+"
TD5:
INC SI
INC DI
DEC BX
INC AX
CMP BYTE PTR [SI], "E"
JNZ TD6
MOV BYTE PTR ES:[DI], "0"
INC DI
INC AX
TD9:
MOVSB
INC AX
DEC BX
JMP TD7
TD6:
CMP BYTE PTR [SI], 0DH
JNZ TD19
CALL NUMBER_END
ADD SI, BX
MOV BX, OFFSET TD16
JMP BX
TD19:
MOVSB
INC AX
DEC BX
TD8:
CMP BYTE PTR [SI], "E"
JZ TD9
CMP BYTE PTR [SI], 0DH
JNZ TD20
CALL NUMBER_END1
ADD SI, BX
MOV BX, OFFSET TD16
JMP BX
TD20:
MOVSB
INC AX
DEC BX
JMP TD8
TD7:
CMP BYTE PTR [SI], "-"
JNZ TD10
MOV BYTE PTR ES:[DI], "-"
JMP TD11
TD10:
MOV BYTE PTR ES:[DI], "+"
TD11:
INC SI
INC DI
DEC BX
INC AX

```

```

    CMP BYTE PTR [SI],0DH
    JNZ TD17
    CALL NUMBER_END
    ADD SI,BX
    JMP TD16
TD17:
    MOVSB
    INC AX
    DEC BX
    CMP BYTE PTR [SI],0DH
    JNZ TD17
    CALL NUMBER_END1
    ADD SI,BX
TD16:
    POP BX
    DEC CX
    CMP CX,0
    JZ TD22
    MOV DX,OFFSET TD15
    JMP DX
TD22:
    POP SI
    CLC
    RET
TRANSFER_DATA ENDP

TRANSFER_DATA1 PROC NEAR
    ADD BL,30H
    MOV BYTE PTR ES:[DI], "+"
    MOV ES:[DI+1],BL
    MOV WORD PTR ES:[DI+2],0A0DH
    ADD DI,4
    ADD AX,4
    RET
TRANSFER_DATA1 ENDP

NUMBER_END PROC NEAR
    MOV BYTE PTR ES:[DI], "0"
    INC AX
    INC DI
NUMBER_END1:
    MOVSB
    INC AX
    DEC BX
    MOV BYTE PTR ES:[DI],0AH
    INC AX
    INC DI
    RET
NUMBER_END ENDP

NET_PARA_FILE PROC NEAR
    CALL CREATE_MA
    CALL CREATE_STAR
    CALL CREATE_ULOOP
    CALL CREATE_BLOOP

```

```

    RET
NET_PARA_FILE ENDP

CREATE_MA PROC NEAR
    CMP NET_CON, 2
    JZ CMA1
    RET
CMA1:
    CMP MA_CON, 1
    JZ CMA31
    CMP MA_CON, 2
    JZ CMA32
    MOV DX, OFFSET CSMA_TDT
    JMP CMA33
CMA32:
    MOV DX, OFFSET TREE_TDT
    JMP CMA33
CMA31:
    MOV DX, OFFSET ALOHA_TDT
CMA33:
    CALL CREATE_FILE
    MOV FILE_HANDLE, AX

    XOR AX, AX
    XOR BL, BL
    MOV SI, OFFSET MA_TDT1
    MOV DI, OFFSET SBU
    CALL TRANSFER_DATA
    XOR BL, BL
    MOV SI, OFFSET MA_TDT2
    CALL TRANSFER_DATA
    XOR BL, BL
    MOV SI, OFFSET MA_TDT3
    CALL TRANSFER_DATA
    MOV BL, NET_OD_MA
    CALL TRANSFER_DATA1
    MOV BL, REAL_TIME
    CALL TRANSFER_DATA1

    CMP MA_CON, 2
    JZ CMP76
    CMP MA_CON, 1
    JZ CMP77
    MOV SI, OFFSET MA_TDT5
    JMP CMP78
CMP77:
    MOV SI, OFFSET MA_TDT4
CMP78:
    XOR BL, BL
    CALL TRANSFER_DATA
CMP76:

    CALL FILE_PREPARE3
    RET
CREATE_MA ENDP

```

```

CREATE_STAR PROC NEAR
    CMP NET_CON, 1
    JZ CSR1
    RET
CSR1:
    CMP SAF_TP, 1
    JZ CSR2
    RET
CSR2:
    MOV DX, OFFSET STAR_TDT
    CALL CREATE_FILE
    MOV FILE_HANDLE, AX

    CALL SAVE_SF
    XOR BL, BL
    MOV SI, OFFSET S_TDT1
    CALL TRANSFER_DATA
    XOR BL, BL
    MOV SI, OFFSET S_TDT3
    CALL TRANSFER_DATA
    CALL SAVE_SF2

    CALL FILE_PREPARE3
    RET
CREATE_STAR ENDP

```

```

CREATE_ULOOP PROC NEAR
    CMP NET_CON, 1
    JZ CUL1
    RET
CUL1:
    CMP SAF_TP, 2
    JZ CUL2
    RET
CUL2:
    MOV DX, OFFSET ULOOP_TDT
    CALL CREATE_FILE
    MOV FILE_HANDLE, AX

    CALL SAVE_SF
    XOR BL, BL
    MOV SI, OFFSET U_TDT1
    CALL TRANSFER_DATA
    CALL SAVE_SF2

    CALL FILE_PREPARE3
    RET
CREATE_ULOOP ENDP

```

```

CREATE_BLOOP PROC NEAR
    CMP NET_CON, 1
    JZ CBL1
    RET
CBL1:

```

```

    CMP SAF_TP, 3
    JZ CBL2
    RET
CBL2:
    MOV DX, OFFSET BLOOP_TDT
    CALL CREATE_FILE
    MOV FILE_HANDLE, AX

    CALL SAVE_SF
    XOR BL, BL
    MOV SI, OFFSET B_TDT1
    CALL TRANSFER_DATA
    XOR BL, BL
    MOV SI, OFFSET B_TDT3
    CALL TRANSFER_DATA
    CALL SAVE_SF2

    CALL FILE_PREPARE3
    RET
CREATE_BLOOP ENDP

SAVE_SF PROC NEAR
    XOR AX, AX
    XOR BL, BL
    MOV SI, OFFSET AFS_TDT1
    MOV DI, OFFSET SBU
    CALL TRANSFER_DATA
    XOR BL, BL
    MOV SI, OFFSET S_TDT2
    CALL TRANSFER_DATA
    MOV SI, OFFSET AFS_TDT2
    XOR BL, BL
    CALL TRANSFER_DATA
    RET
SAVE_SF ENDP

SAVE_SF2 PROC NEAR
    MOV SI, OFFSET MA_TDT3
    XOR BL, BL
    CALL TRANSFER_DATA
    MOV BL, NET_OD_SAF
    CALL TRANSFER_DATA1
    MOV BL, REAL_TIME
    CALL TRANSFER_DATA1
    RET
SAVE_SF2 ENDP

FILE_PREPARE2 PROC NEAR
    PUSH DS
    MOV AX, CODE
    MOV DS, AX
    MOV SI, OFFSET ERROR_HANDLER2
    CALL SET_RETURN_ADDRESS
    CALL SET_CRITICAL_ADDRESS
    POP DS

```

```
RET  
FILE_PREPARE2 ENDP
```

```
FILE_PREPARE3 PROC NEAR  
MOV CX,AX  
MOV BX,FILE_HANDLE  
MOV DX,OFFSET SBU  
CALL WRITE_TO_FILE  
MOV BX,FILE_HANDLE  
CALL CLOSE_FILE  
RET  
FILE_PREPARE3 ENDP
```

```
ERROR_HANDLER2 PROC NEAR  
MOV AX,CONST  
MOV DS,AX  
MOV ES,AX  
MOV SP,STACK_POINTER1  
MOV AX,OFFSET RS7  
JMP AX  
RET
```

```
ERROR_HANDLER2 ENDP
```

```
CODE ENDS  
END
```