



TNO-rapport

rapport no.
FEL-91-A171

exemplaar no.

titel

9 Een meetsysteem voor het testen van
radiocommunicatie-apparatuur

AD-A245 621



Niets uit deze uitgave mag worden
vermenigvuldigd en/of openbaar gemaakt
door middel van druk, fotokopie, microfilm
of op welke andere wijze dan ook, zonder
voorafgaande toestemming van TNO.
Het ter inzage geven van het TNO-rapport
aan direct belanghebbenden is toegestaan.

Indien dit rapport in opdracht werd
uitgebracht, wordt voor de rechten en
verplichtingen van opdrachtgever en
opdrachtnemer verwezen naar de
'Algemene Voorwaarden voor Onderzoeks-
opdrachten TNO', dan wel de betreffende
terzake tussen partijen gesloten
overeenkomst.

TNO

auteur(s):

P. Fecstra

datum:

augustus 1991

DTIC
ELECTE
S FEB 04 1992 D

rubricering

titel

: ongerubriceerd

samenvatting

: ongerubriceerd

rapporttekst

: ongerubriceerd

bijlagen A + B

: ongerubriceerd

This document has been approved
for public release and sale; its
distribution is unlimited.

oplage

: 21

aantal bladzijden

: 52 (incl. bijlagen;
excl. RDP + distributielijst)

aantal bijlagen

: 2

92-02808
■■■■■■■■■■

9 2 2 03 1 68





rapport no. : FEL-91-A171
titel : Een meetsysteem voor het testen van radiocommunicatie-apparatuur

auteur(s) : P. Feenstra
instituut : Fysisch en Elektronisch Laboratorium TNO

datum : augustus 1991
hdo-opdr.no. : A89KL641
no. in lwp '91 : 711.2

Onderzoek uitgevoerd o.l.v. : ir. G.J.M. Janssen
Onderzoek uitgevoerd door : Vdg. ir. W.T.E. Vaessen, P. Feenstra

Accession For	
NTIS CRA&I	☒
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution	
Availability Codes	
Dist	Availability Codes
A-1	Special

SAMENVATTING (ONGERUBRICEERD)

Dit rapport beschrijft een testfaciliteit voor het meten van het gedrag en de kwaliteit van radiocommunicatie-apparatuur in een gesimuleerde operationele omgeving. De werkzaamheden beschreven in dit rapport zijn uitgevoerd in het kader van de opdracht "ECCM storingsonderdrukker" (A89KL641). De testopstelling maakt het mogelijk de gesimuleerde propagatieomstandigheden zoals demping, fading en het optreden van meerdere transmissiepaden van de gesimuleerde verbinding te kiezen. Daarnaast is het mogelijk om verschillende soorten stoorsignalen toe te voegen. Het criterium voor het classificeren van de te testen radio-apparatuur is de kwaliteit van de overgedragen informatie, d.w.z. de SINAD-waarde in een analoog systeem en de BER (Bit Error Rate) in een digitaal systeem.

Het testsysteem wordt bestuurd door een PC, door toepassing van windows is de bediening van de testopstelling eenvoudig.

De testfaciliteit kan ingezet worden bij de ontwikkeling van nieuwe apparatuur of voor de controle en bepaling van specificaties van bestaande radiocommunicatie-apparatuur.

Met de opstelling beschikt het FEL over een faciliteit om de kwaliteit van radio-apparatuur in een gesimuleerde operationele situatie te testen.

report no. : FEL-91-A171
title : A measuring facility for testing of radio communication equipment

author(s) : P. Feenstra
Institute : TNO Physics and Electronics Laboratory

date : August 1991
NDRO no. : A89KL641
no. in pow '91 : 711.2

Research supervised by: ir. G.J.M. Janssen
Research carried out by: Vdg. ir. W.T.E. Vaessen, P. Feenstra

ABSTRACT (UNCLASSIFIED)



This report describes a test facility for measuring the behaviour and quality of radio communication equipment in a simulated operational environment. The work as described in this report was performed in the scope of the assignment "ECCM jammer suppression" (A89KL641).

The testbed allows to select the simulated propagation characteristics, such as attenuation, fading and the occurrence of multipath effects. Besides it is possible to insert various jamming signals. The criterion for classifying radio equipment under test is the quality of transferred information, that is the SINAD value in an analog system and the BER for a digital system.

The test set-up is controled by a PC, the operation is easy by use of windows.

This test facility can be employed for developing new equipment and for examining specifications of radio communication equipment in existence.

With this test formation FEL has the disposal of a facility to test the quality of radio equipment in a simulated operational situation. *Geen aanpak*

(1)

INHOUD

SAMENVATTING		2
ABSTRACT		3
INHOUD		4
1	INLEIDING	6
2	DE MEETOPSTELLING	8
2.1	Simulatie van het radiokanaal	8
2.1.1	Propagatie-effecten	9
2.1.2	Interferentie en storingen	10
2.2	De signaalanalyse	10
2.2.1	SINAD	11
2.3	De zend- en ontvangstzijde	12
2.4	Besturing en totaal overzicht van het testsysteem	12
2.5	De gerealiseerde testopstelling	13
3	HET BESTURINGSPROGRAMMA	16
3.1	Algemeen	16
3.2	LabWindows	16
3.3	Overzicht van het programma	18
4	BEVINDINGEN	22
5	CONCLUSIE	25
BIJLAGE A: DE BEDIENING VAN DE MEETOPSTELLING		
A1	Woord vooraf	A.1
A2	De bediening	A.1

A3	Grafiek	A.6
----	---------	-----

BIJLAGE B: PRINTOUTS

B1	Printout van het besturingsprogramma "TESTER.C"	B.1
----	---	-----

B2	Printout van de include file "TESTER.H"	B.19
----	---	------

1 INLEIDING

De groep Telecommunicatie doet regelmatig onderzoek en metingen aan radiocommunicatie-apparatuur. Dit onderzoek houdt in, dat radioapparatuur of subdelen hiervan op hun effectiviteit worden getest. Hierbij wordt voornamelijk gekeken naar de effecten van verschillende propagatiefenomenen of opzettelijk in de verbinding geïntroduceerde storingen. Bij dit onderzoek is het nodig dat de testen worden uitgevoerd onder reële omstandigheden.

Helaas treden de propagatiefenomenen en storingen in een radioverbinding slechts tijdelijk op en zijn bovendien niet goed gedefinieerd. Alleen als men zeer langdurig metingen uitvoert en dus alle propagatiefenomenen gemiddeld even vaak aan bod laat komen, kan men uit de meetresultaten verantwoorde conclusies trekken. Hierdoor zijn testen in een echt radiokanaal niet snel en eenvoudig uit te voeren. Bovendien krijgt men van de bevoegde instanties voor dit soort metingen slechts enkele zendfrequenties toegewezen en dan nog voor een beperkte tijd.

Om deze problemen te omzeilen is in de Telecommunicatiegroep gestart met het ontwikkelen van een testopstelling. Deze testopstelling simuleert een radiokanaal, waarvan de nagebootste propagatiefenomenen instelbaar zijn. Tevens is het mogelijk verschillende typen stoorsignalen toe te voegen. Hierdoor is het mogelijk het testen van radiocommunicatie-apparatuur lokaal te verrichten.

Deze testopstelling is een eerste stap naar een complete opstelling voor het testen van radiozend/ontvangers, modems en andere voor radiocommunicatie gebruikte apparatuur of schakelingen. Het compleet maken van de testopstelling dient in samenhang met toekomstige projecten te geschieden. Belangrijk is echter, dat deze testopstelling met toekomstige uitbreidingen intact blijft en vooral dat de documentatie toegankelijk is en "up to date" bewaard blijft. Op deze wijze hoeven niet voor elke opdracht opnieuw een testopstelling samengesteld en programma's ontwikkeld te worden, maar kan men standaard gebruik maken van de bestaande testopstelling.

Opbouw rapport:

Dit rapport geeft een beschrijving van de testopstelling.

In hoofdstuk 2 is beschreven hoe de opstelling er definitief zal gaan uitzien en wat op dit moment gerealiseerd is.

In hoofdstuk 3 wordt ingegaan op de besturing van de testopstelling, met name de toepassing van LabWindows, de opzet van programma en de presentatie van de testresultaten.

De bijlage bevat een handleiding voor de bediening van de testopstelling en de printout van het besturingsprogramma.

2 DE MEETOPSTELLING

Met deze testopstelling kan de kwaliteit en effectiviteit van radiozenders/ontvangers en van de modulator en demodulator gemeten worden. De metingen worden verricht in een gesimuleerd radiokanaal.

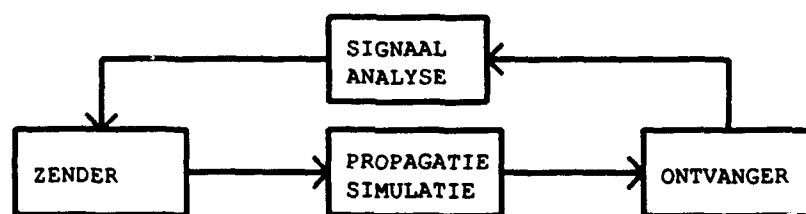


Fig. 1: Eenvoudig blokschema van de testopstelling

Een eenvoudig blokschema van de testopstelling is in figuur 1 weergegeven. Het bestaat uit vier funktiegroepen. De paragrafen 2.1 t/m. 2.3 geven een beschrijving van deze funktieblokken.

2.1 Simulatie van het radiokanaal

Een radioverbinding wordt gekarakteriseerd door de propagatie-effecten, zoals demping en reflectie en de interferentie en storing door andere zender(s).

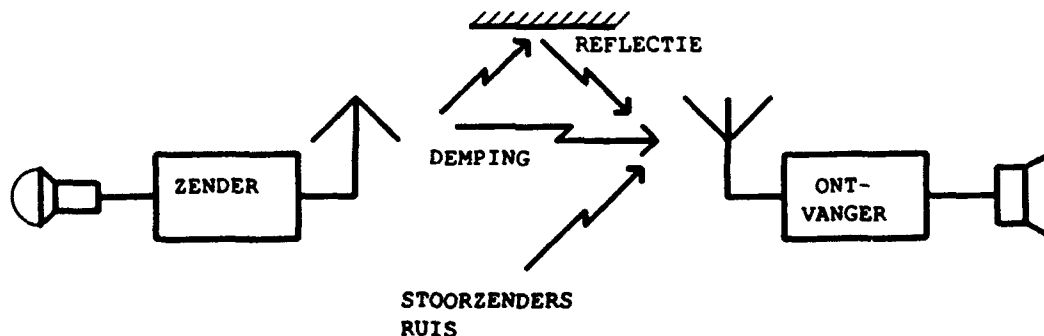


Fig. 2: Een radioverbinding

2.1.1 Propagatie-effecten

In een radiokanaal ondergaat het uitgezonden signaal in het propagatiepad de invloed van allerlei factoren die de kwaliteit van het signaal aantasten en daardoor het ontvangstbereik verkleinen. De belangrijkste faktor is natuurlijk de trajectdemping. Maar het bereik wordt niet alleen hierdoor bepaald. Het radiosignaal kan meerdere, onderling in lengte verschillende, wegen volgen. Dit kan het gevolg zijn van reflecties of door het optreden van meerdere propagatiepaden in de atmosfeer ten gevolge van troposferische en ionosferische variaties. Door meerwegpropagatie treden twee effecten op:

- Fading

Fading ontstaat als een signaal via meerdere wegen de ontvanger bereikt. Deze signalen arriveren met verschillende amplituden en fasen, waardoor frequentieselectieve versterking en uitdoving ontstaat. Dit verschijnsel noemt men fading. Fading is stabiel, indien het propagatiemedium niet verandert in de tijd. Bij een veranderend medium of veranderde propagatiepaden (bijvoorbeeld bij mobiele communicatie) treedt ook variërende frequentieselectieve fading op.

- Vervorming:

Naast het effect van fading kan bij meerwegpropagatie ook vervorming van het informatiesignaal optreden. Dit ontstaat doordat het radiosignaal meerdere keren met verschillende vertragingen wordt ontvangen. Dit heeft vervorming van het informatiesignaal tot gevolg. Met name bij digitale signaaltransmissie kunnen de optredende verschillen in vertragingstijd voor de verschillende paden groot worden t.o.v. de tijd van een databit waardoor de bits in elkaar overlopen (inter-symbol interference). Dit heeft vergroting van de BER (Bit Error Rate) tot gevolg.

De betrouwbaarheid van een radioverbinding wordt door het optreden van deze propagatiefenomenen in hoge mate bepaald door het werkingsprincipe en de kwaliteit van de gebruikte radiocommunicatie-apparatuur. Het is dan ook noodzakelijk dat de testinstallatie deze propagatiefenomenen kan nabootsen, zodat het gedrag van de te testen radiocommunicatie-apparatuur op deze punten nauwkeurig onderzocht kan worden.

In het gesimuleerd radiokanaal van het testsysteem zijn deze fenomenen apart of in combinatie in te schakelen en in te stellen, zodat het gedrag van de te testen radiocommunicatie-apparatuur op deze punten onderzocht kan worden.

2.1.2 Interferentie en storingen

In de meeste radioverbindingen heeft men niet alleen te maken met natuurlijke propagatiefenomenen, maar ook met storingen door andere zenders. Deze storingen kunnen zowel onopzettelijk als opzettelijk van oorsprong zijn. Een legale zender kan onopzettelijk storing veroorzaken in een radioverbindingen, ook als de frequenties verschillend zijn. De kwetsbaarheid voor deze storingen wordt in hoofdzaak veroorzaakt door een niet korrekte werking van de ontvanger. In een ontvanger wordt het gewenste signaal in frequentie geconverteerd naar een z.g. middenfrequent. Een filter op deze middenfrequent zorgt voor de frequentieselectiviteit van de ontvanger. Helaas kunnen door een reeks van oorzaken ongewenste radiosignalen na conversie ook frequentieprodukten hebben in het middenfrequentspectrum en daardoor interfereren met het gewenste signaal. Hoewel het dus ongewenste radiostations zijn die interfereren, zijn niet deze radiostations, maar is de ontvanger zelf de oorzaak van het probleem. De interferentieproblemen worden in de ontvanger veroorzaakt door resp. onvoldoende voorselectiviteit, harmonische van ingangssignalen t.g.v. niet lineariteit van versterkertrappen, onvoldoende spectrale reinheid van oscillatoren in de ontvanger en produkten verkregen uit ongewenste menging in de voortrappen van de ontvanger.

Bij militaire verbindingen moet men ook in hoge mate rekening houden met opzettelijk veroorzaakte storingen. Een tegenstander kan bewust op dezelfde frequentie uitzenden om daarmee de radioverbinding te storen of te "jammen".

In deze testopstelling kan een stoorzender of een ruisbron in de radioverbinding gebracht worden. Van de stoorzender is de modulatie, de frequentie en het niveau instelbaar. Van de ruisbron is alleen het niveau instelbaar.

2.2 De signaalanalyse

De verzonden informatie moet aan de ontvangstzijde op de juiste wijze kunnen worden geïnterpreteerd. Daarvoor is het vereist dat het ontvangen informatiesignaal een redelijke reproductie is van het verzonden signaal. De golfvorm van het over te dragen informatie- of audiosignaal wordt in elke schakel van de radioverbinding in meer of mindere mate aangetast. De mate van aantasting is voor een belangrijk deel afhankelijk van de kwaliteit en het werkingsprincipes van de gebruikte radiocommunicatie-apparatuur. De testopstelling is opgezet voor het quantificeren van de kwaliteit van radiocommunicatie-apparatuur bij het optreden van verschillende propagatiefenomenen. Voor het testen en onderling vergelijken van radiosystemen bepaalt men de betrouwbaarheid van de ontvangen informatie. Bij communicatie maakt men

onderscheid tussen analoge en digitale systemen, wat betekent dat de over te zenden informatie analoog of digitaal is. Voor een analoog systeem moet dus het percentage vervorming plus ruis in het ontvangen audiosignaal worden gemeten en voor een digitaal systeem de bitfoutkans.

De testopstelling gebruikt voor het testen van een analoog signaal een audio-analyzer en voor een digitaal signaal een BER-meter (Bit Error Rate).

2.2.1 SINAD

Voor het onderling vergelijken van de prestaties van analoge radiosystemen is het noodzakelijk van een norm gebruik te maken, waarmee de kwaliteit van een audiosignaal ondubbelzinnig wordt uitgedrukt. Veel gebruikt wordt de uitdrukking SNR (Signal to Noise Ratio). De SNR wordt uitgedrukt in decibels.

$$\text{SNR} = 10 \log(S/N);$$

S en N zijn de vermogens van resp. het signaal en de ruis.

Metten van de SNR is met relatief eenvoudige meetapparatuur uit te voeren en om die reden is deze norm dan ook populair.

Echter, de SNR geeft de kwaliteit van een signaal niet geheel correct weer, omdat de vervorming verwaarloosd wordt. Ook is het moeilijk het signaalniveau te meten zonder de ruis. Beter is daarom de definitie,

$$\text{SINAD} = 10 \log \left(1 + \frac{S}{N+D} \right);$$

hierin zijn S, N en D de vermogens van resp. signaal, ruis en vervorming.

Deze verhouding noemt men de "SINAD" waarde van een signaal. Het woord SINAD is gevormd uit de eerste letters van Signal Noise And Distortion. De SINAD-waarde is een bruikbare indicatie voor de verstaanbaarheid van een signaal.

In de testopstelling wordt de SINAD-waarde gemeten door de HP 8903B audio-analyzer.

De SINAD-waarde wordt gemeten bij slechts één audiofrequentie. Voor een goede indruk van de kwaliteit van een verbinding voor het gehele gehoor- of informatiespectrum moet men de SINAD-meting op meerdere frequenties in dit spectrum herhalen.

Heeft men te maken met spraak, dan zijn voor de verstaanbaarheid alleen de frequenties van ongeveer 300 tot 3600 Herz van belang. De signalen buiten deze frequentieband wegen minder mee voor de kwaliteit van de gesproken informatie en kunnen dan ook onderdrukt worden. Ook bij de meting van de SINAD-waarde houdt men hier dikwijls rekening mee door een zogenaamde gewogen meting te maken, d.w.z. het frequentiespectrum buiten de spraakband te filteren. Voor

de frequentiekenarakteristiek van dergelijke weegfilters bestaan officieel aanbevolen specificaties. Over het algemeen wordt de door CCITT ¹ voorgeschreven filterkarakteristiek gebruikt.

2.3 De zend- en ontvangstzijde

Met de testopstelling kunnen ook modems in een radioverbinding getest worden. Hiervoor is in de testopstelling aan de zenzijde een meetzender en aan de ontvangstzijde een meetontvanger opgenomen. De radiofrequentie, modulatie en middenfrequentbandbreedte van zowel de meetzender als van de meetontvanger zijn instelbaar en dus optimaal aan te passen aan de te testen modems.

2.4 Besturing en totaal overzicht van het testsysteem

Het produkt van de volledige test is een kromme, de grafische presentatie van de prestatie of kwaliteit van het geteste apparaat of schakeling. De kromme wordt verkregen uit een reeks opeenvolgende metingen. Het bepalen van deze krommes vereist de uitvoering van een reeks routinematige metingen, waarbij een parameter in de test steeds een gelijke opeenvolgende verandering ondergaat. In de testopstelling is een PC opgenomen met programmatuur voor het automatisch uitvoeren van deze routinematige metingen. De bediening via de PC voor het instellen van de metingvoorwaarden dient eenvoudig gehouden te worden, zonder dat dit de flexibiliteit benadeelt.

Tenslotte is in figuur 3 het gedetailleerde blokschema van het testsysteem zoals beschreven in de paragrafen 2.1 t/m 2.4 te zien.

¹ Consultative Committee in International Telephone and Telegraph - recommendation G.227 (Yellow book 1980)

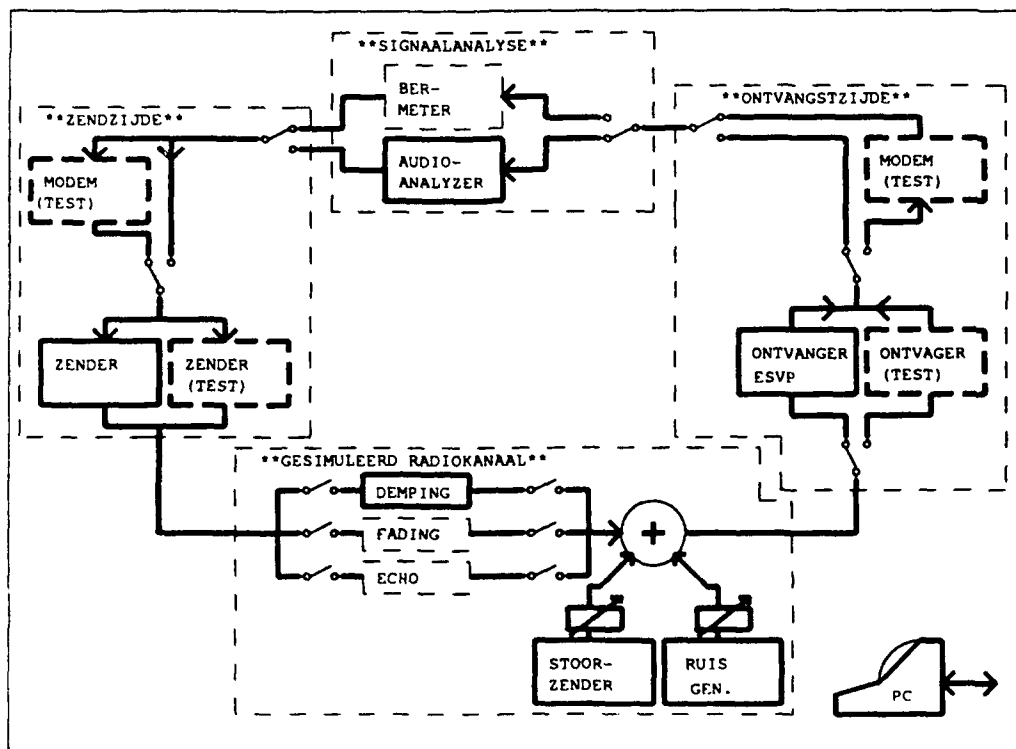


Fig. 3: Blokschema van de complete testopstelling

2.5 De gerealiseerde testopstelling

De simulatie-opstelling zoals hiervoor beschreven is voor een belangrijk deel gerealiseerd. Het blokschema van de gerealiseerde testopstelling is weergegeven in figuur 4. Het bestaat uit de volgende instrumenten:

- Marconi 2022 meetzender
- Rohde & Schwarz ESVP meetontvanger
- Marconi 2019 meetzender
- Rohde & Schwarz SUF2 ruisgenerator
- Hewlett and Packard 8903B audio analyzer
- Compaq model-2 PC.

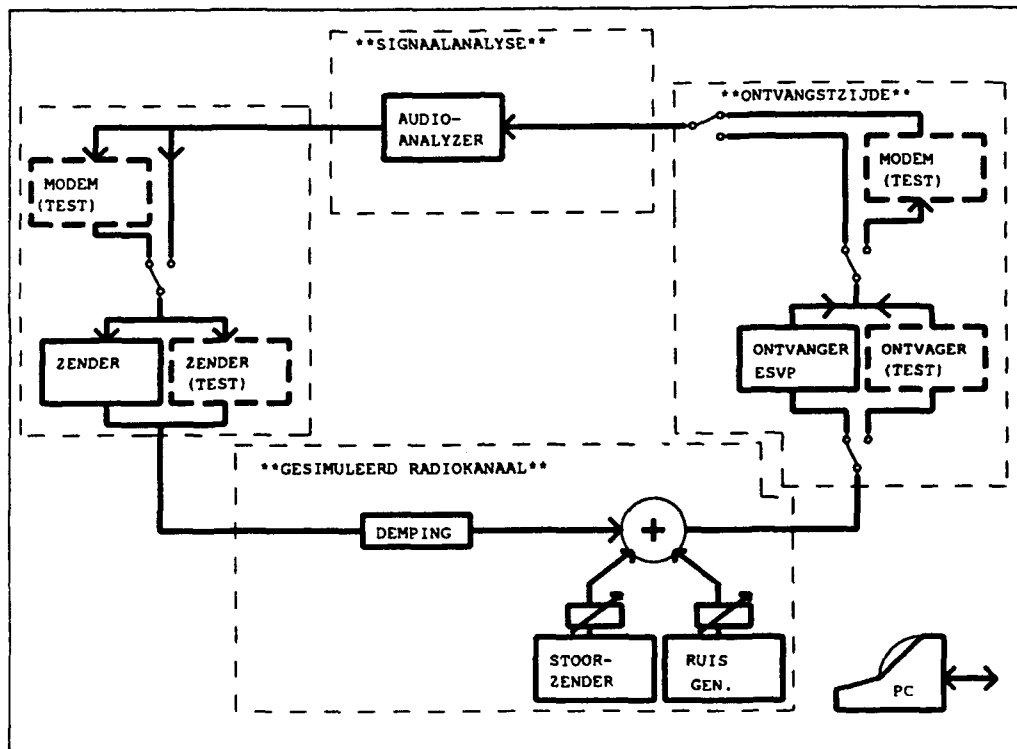


Fig. 4: Blokschema van de gerealiseerde opstelling

Het radiopad wordt gevormd door de Marconi 2022 meetzender en de Rohde & Schwarz ESVP meetontvanger. In het radiopad kan een tweede zender of stoorzender en een ruisbron opgenomen worden; dit zijn resp. de Marconi 2019 meetzender en de Rohde & Schwarz SUF2 ruisgenerator. Zender, stoorzender en ruisgenerator zijn via powersplitter/combiners van Mini Circuits gekoppeld met de ontvanger. De PC bestuurt en controleert het verloop van de metingen. De meetzenders hebben een frequentiebereik van 10 kHz tot 1 GHz. De ESVP meetontvanger heeft een frequentiebereik van 20 MHz tot 1.3 GHz, doch deze is gemakkelijk te vervangen door een type ESH eveneens van Rohde & Schwarz, met een frequentiebereik van 10 kHz tot 25 MHz. De bediening en besturing van beide apparaten zijn identiek.

In foto 1 is de bestaande testopstelling afgebeeld.

Van boven naar beneden zijn de volgende instrumenten te zien:

- 1- de audio-analyzer
- 2- de meetontvanger
- 3- de PC
- 4- propagatiepadverzwakkers en combiners
- 5- een plotter
- 6- van links naar rechts: de ruisgenerator en de meetzender voor het informatiesignaal
- 7- de meetzender voor het stoorzendersignaal.

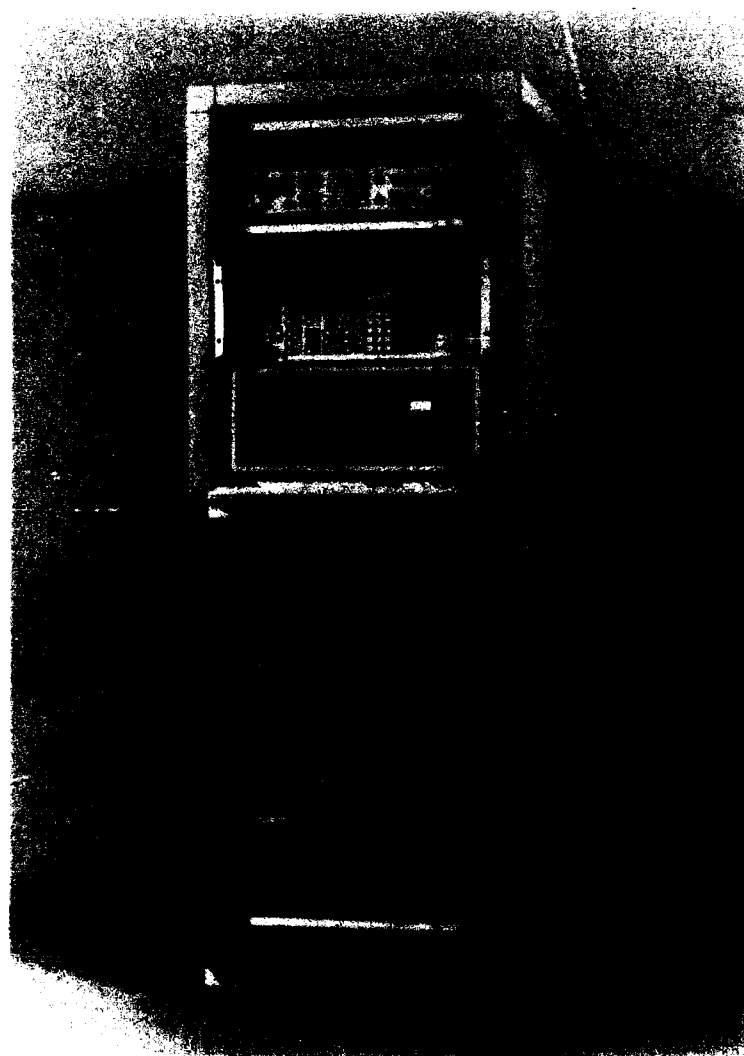


Foto 1: Het meetsysteem

3 HET BESTURINGSPROGRAMMA

3.1 Algemeen

Alle meetinstrumenten uit de testopstelling zijn voorzien van een zogenaamde GPIB (General Purpose Interface Bus)² aansluiting. Via deze aansluitingen is het mogelijk de instrumenten te besturen met behulp van een PC en de meetresultaten uit te lezen. Voor deze besturing is een programma ontwikkeld, waarmee op eenvoudige en snelle wijze metingen kunnen worden uitgevoerd. Dit besturingsprogramma is ontwikkeld met en maakt gebruik van het ontwikkelprogramma LabWindows.

3.2 LabWindows

Labwindows, een softwareprodukt van National Instruments, is een hulpmiddel bij het ontwikkelen van software voor de besturing van meetinstrumenten en het inlezen, analyseren en presenteren van de met deze meetinstrumenten verkregen meetresultaten. LabWindows is bestemd voor een PC met DOS operating system. Met LabWindows worden de functies van een instrument softwarematig ingesteld d.m.v. invoervelden op het scherm van de PC. Deze velden zijn per meetinstrument gegroepeerd in één of meer zg. panels. Een panel vertegenwoordigt één of meer programmaregels voor de besturing van het bijbehorende meetinstrument. Deze programmaregels roepen procedures aan. Door deze procedures worden de meetinstrumenten ingesteld, metingen uitgevoerd en meetgegevens uitgelezen.

De logische volgorde van de panels voor een instrument maakt het de gebruiker gemakkelijk stapsgewijs programmaregels te schrijven en deze te laten uitgroeien tot een werkend besturingsprogramma.

Voor elk in LabWindows ingevoerd instrument zijn in de bibliotheek of library de aparte instrumentmodules aanwezig. Een instrumentmodule bestaat uit twee files, de file met het programma voor de panel(s) en hieraan gekoppeld een file met het procedures voor de uitvoering van de panelcommando's. De bibliotheek van LabWindows bezit behalve instrumentmodules ook modules voor de analyse en grafische presentatie van meetresultaten. Met deze modules is

2 Komt overeen met de IEEE-488.1 specificatie

het mogelijk, op eenzelfde manier als bij de instrumentmodules, een programma te schrijven voor het analyseren van meetresultaten of het maken van grafieken.

Ingepulde panels kunnen op twee manieren toegepast worden: "program" of "interactive". "Program" wil zeggen, dat de ingepulde panels worden gebruikt voor het schrijven van een compleet programma en dat dit programma wordt opgeslagen in een file, terwijl "interactive" betekent, dat ingepulde panels direkt, dus interaktief, worden uitgevoerd.

"Program" en "interactive" zijn afwisselend te gebruiken, dus men kan via "interactive" programmaregels testen en goed werkende regels gebruiken voor het maken van een definitief programma.

Tijdens het "runnen" van een programma bewaart LabWindows automatisch de waarden van alle gebruikte variabelen. Deze waarden zijn later toegankelijk voor inspektie of te gebruiken voor een grafische presentatie.

Behalve de waarden van de variabelen bewaart LabWindows ook alle data die tussen PC en meetinstrument(en) wordt uitgewisseld. Omdat deze data zo karakteristiek is voor het verloop van een programma is dit een belangrijk hulpmiddel voor de controle van de werking van het programma en het lokaliseren van fouten.

Een figuurlijke voorstelling van de werking en mogelijkheden van LabWindows is te zien in figuur 5.

LabWindows wordt inmiddels zo veel gebruikt, dat de producenten van meetinstrumenten voor de meeste van hun nieuwe produkten de instrumentsmodules reeds ontwikkelen en deze als een optie bij het meetinstrument kunnen leveren aan de klant. Voor de in het verleden aangeschafte meetinstrumenten is dit helaas niet het geval, zodat deze modules zelf ontwikkeld dienen te worden. In de telecommunicatiegroep zijn voor een aantal meetinstrumenten deze modules ontwikkeld. Bovendien is voor de testopstelling uit deze al eerder ontwikkelde instrumentmodules een nieuw module ontwikkeld met behulp waarvan de totale testopstelling eenvoudig en gericht is te bedienen.

In LabWindows zijn twee programmatalen geïmplementeerd; Quick Basic en C. Het besturingsprogramma van de testopstelling is in C geschreven.

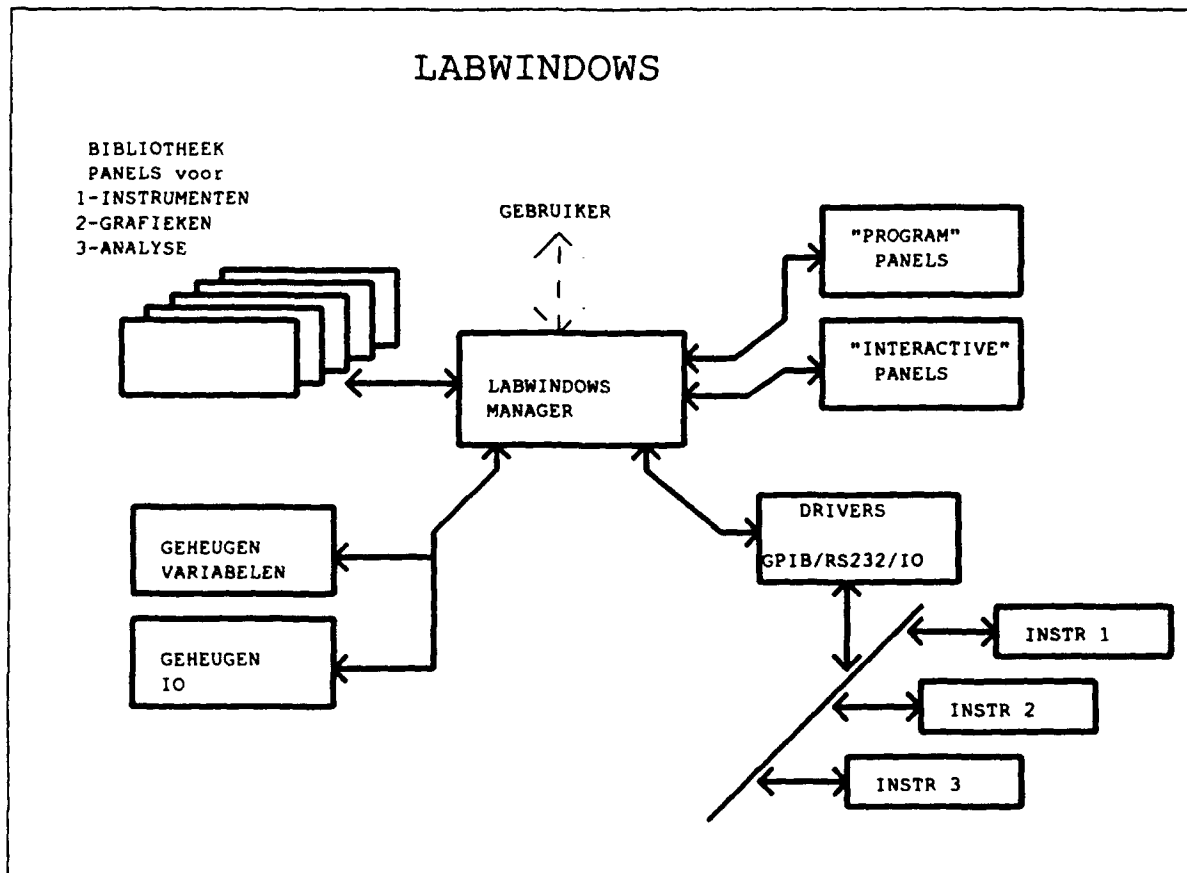


Fig. 5: Figuurlijke voorstelling van LabWindows

3.3 Overzicht van het programma

De testopstelling laat zich controleren door in totaal 6 opeenvolgende LabWindows-panels. Elk panel is verbonden met een procedure. Door het "runnen" van de met het panel verbonden procedure stellen de betrokken meetinstrumenten zich in volgens de via de velden van het panel ingevoerde parameters.

Indien tijdens het "runnen" van een panel een fout optreedt zal in het veld "error" een getal ongelijk aan nul te zien zijn. Aan de hand van dit getal is de oorzaak van de fout te lokaliseren.

Het meten van de SINAD-waarden wordt met panel 5 gestart. Met panel 2 kiest men de propagatiecondities, d.w.z. de frequentie en modulatie en de keuze of de verbinding gestoord wordt door ruis of door een stoorzender.

De meetinstrumenten die door het programma worden bestuurd zijn;

- Marconi 2022 meetzender
- Rohde & Schwarz ESVP meetontvanger
- Marconi 2019 meetzender
- Rohde & Schwarz SUF2 ruisgenerator
- Hewlett and Packard 8903B audio analyzer.

De 6 panels met een korte beschrijving zijn hieronder aangegeven. Voor een gedetailleerde bedieningsinstructie wordt de lezer verwezen naar bijlage A.

Panel 1: Initialisatie

Het eerste panel initialiseert de installatie. D.w.z., elk van de vijf instrumenten krijgt in LabWindows een poort toegewezen. Deze poort koppelt de instrumentmodule met het instrument via de opgegeven GPIB-adres. De initialisatieprocedure controleert ook of op het GPIB-adres inderdaad een instrument aanwezig is en bij sommige instrumenten herkent het door vraag en antwoord ook of dit het juiste type instrument is.

Panel 2: Instel_radiokanaal

Met dit panel kiest men de frequentie van het radiokanaal, de modulatiesoort en index en het zendniveau. Ook moet in dit panel de soort storing in de radioverbinding worden geselecteerd, d.w.z. een verbinding die gestoord wordt door een stoorzender, een ruisbron, of door een stoorzender plus ruisbron.

Panel 3: Instel_ruisgenerator

Heeft men in panel 2 als stoorbron ruis gekozen dan moeten hier de metingvoorwaarden worden ingesteld. Door het invullen van de velden voor resp. het beginniveau, het eindniveau en de stapgrootte wordt bepaald bij welke ruisniveaus SINAD-metingen zullen worden uitgevoerd. Het aantal metingen wordt door het programma uit deze gegevens berekend. De stapgrootte moet een geheel getal zijn.

Panel 4: Instel_stoorzender

Met dit panel kan van de stoorzender de frequentie, het niveau en de modulatie gekozen worden. Men kan kiezen tussen een metingreeks met steeds toenemende stoorfrequentie bij constant stoorniveau of, omgekeerd, met toenemend stoorniveau en constante frequentie.

Panel 5: SINAD_meting

Met dit panel kunnen vier audiofrequenties geselecteerd worden. Op deze vier geselecteerde frequenties worden per instelling van de propagatiesimulatie de SINAD-waarden gemeten. "Runnen" van dit panel start een procedure waarmee de SINAD-waarden worden gemeten voor alle instellingen van meetinstrumenten, die met de vorige panels gekozen zijn. De meetresultaten worden in vijf files opgeslagen op de harddisk van de PC. Het eerste file bevat de array met variabele instelwaarden. De overige vier files zijn voor de meetresultaten per audiofrequentie. De namen van de vijf files zijn data0.prn t/m data4.prn of ref0.prn t/m ref4.prn. In dit panel kiest men in het veld "grafiek" met de stand "meting" voor opslaan van de meetresultaten in de files data*.prn en met de stand "referentie" voor ref*.prn. Eventuele oude meetresultaten in deze files worden hierbij overschreven.

Het programma "grafiek.c" gebruikt deze files voor het maken van een grafische presentatie van de meetresultaten. In de grafiek komen twee krommes voor; een meting- en een referentiekromme. Beide krommes zijn ter herkenning voorzien van een eigen markering.

Panel 6: Close

"Runnen" van dit panel maakt alle door dit programma in LabWindows bezette GPIB-poorten weer beschikbaar.

In figuur 6 zijn schematisch de 6 panels met de instelmogelijkheden weergegeven.

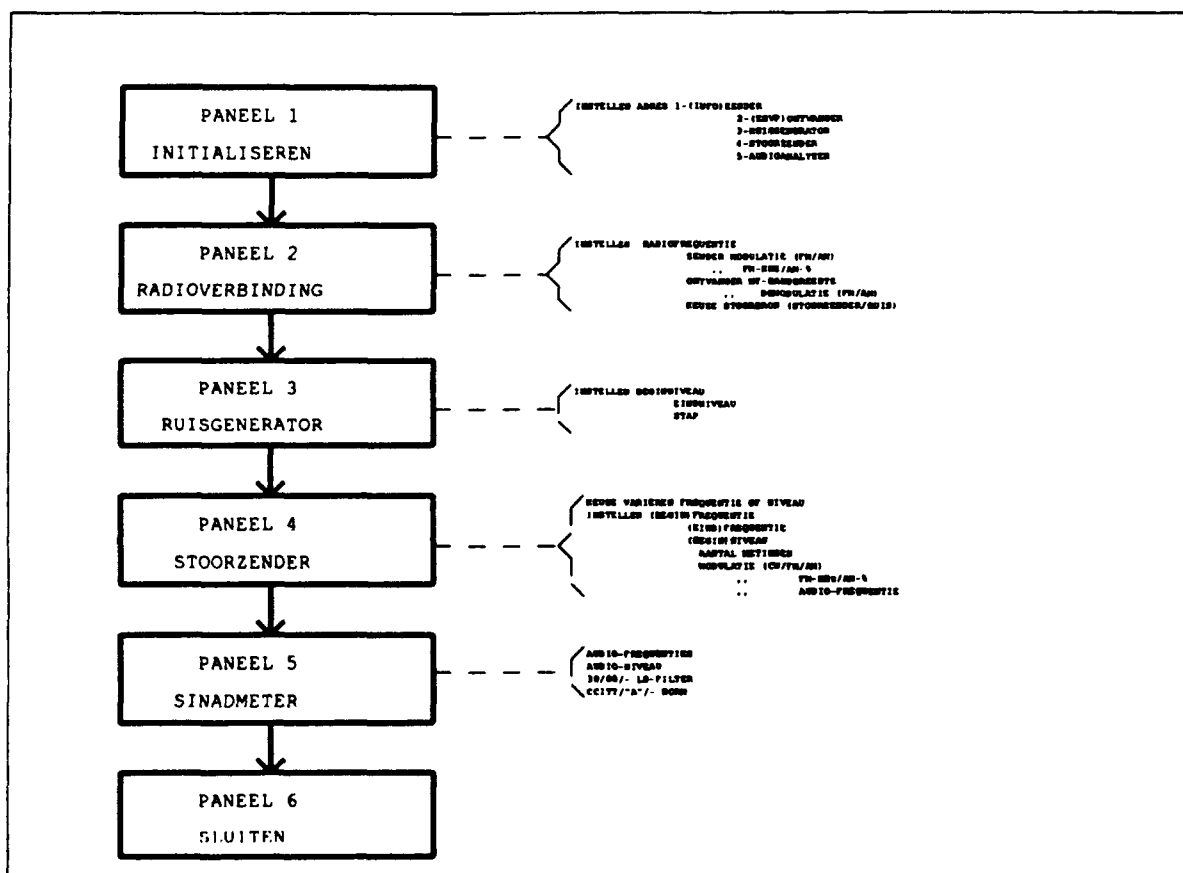


Fig. 6: Overzicht van het besturingsprogramma

4

BEVINDINGEN

Deze testopstelling is voor het eerst gebruikt voor het testen van de werking van een in de telecommunicatie groep ontwikkeld storingsonderdrukkingscircuit.³

Dit storingsonderdrukkingscircuit, dat is gebaseerd op het CCPLL-principe (Cross Coupled Phased Locked Loop), is een toevoeging voor bestaande radio-ontvangers. Het maakt een gewenst signaal verstaanbaar, ook als dit tegelijk wordt ontvangen met een veel sterker stoorsignaal in hetzelfde kanaal. Het storingsonderdrukkingscircuit vervangt de detector in een ontvanger, het is geplaatst achter de middenfrequenttrap van een ontvanger.

Voor het bepalen van de karakteristieken van de storingsonderdrukker is deze m.b.v. de testopstelling getest in een gesimuleerde radioverbinding met een stoorzender.

In figuur 7 en 8 zijn de grafische presentaties van de meetresultaten. Elk grafiek heeft twee krommes, resp. met CCPLL en zonder CCPLL. De kromme met CCPLL is het resultaat van SINAD-metingen van de ontvanger uitgerust met een CCPLL-storingsonderdrukker en de kromme zonder CCPLL geeft het resultaat weer van SINAD-metingen van dezelfde ontvanger zonder storingsonderdrukker.

Bij deze metingen was de frequentie van het radiokanaal 75 MHz, het zenderniveau -50 dBm, de modulatie FM met 5 KHz zwaai en het middenfrequentfilter van de ontvanger 25 KHz breed. Ook van de stoorzender was de modulatie FM met 5 KHz zwaai en een 1000 Hz audiofrequentie.

In figuur 7 zijn de SINAD-waarden gemeten waarbij de frequentie van de stoorzender in 40 stappen het frequentiegebied doorliep van (75 MHz - 150 KHz) tot (75 MHz + 150 KHz). Het niveau van de stoorzender was -40 dBm, 10 dB groter dus dan het niveau van het radiokanaal.

In de grafiek is te zien dat van de ontvanger zonder CCPLL de SINAD-waarde onaanvaardbaar laag wordt als de frequentie van de stoorzender binnen het frequentiespectrum van het middenfrequentbandfilter valt. In dit frequentiegebied geeft de ontvanger met CCPLL wel aanvaardbare SINAD-waarden (groter dan 8 à 10 dB). In de aangrenzende frequentiegebieden geeft de CCPLL juist een verslechtering. Dit is inherent aan de werking van de CCPLL; een stoorsignaal met een frequentie net buiten het middenfrequentfilter van de ontvanger zal een signaal kleiner dan de eigenlijke frequentie produceren waardoor het verkeerde signaal wordt onderdrukt. In die situatie zal men de CCPLL uitschakelen.

3 Zie het TNO-rapport "Een prototype van een storingsonderdrukker d.m.v. een Cross Coupled Phase Locked Loop", auteur P.Hoofslout

In figuur 8 is de stoofrequentie constant en gelijk aan de frequentie van het radiokanaal. Nu zijn de SINAD-waarden gemeten voor een oplopende reeks stoorniveaus.

Deze grafieken zijn gemaakt m.b.v. Lotus 123 uit datafiles, die verkregen zijn met het besturingsprogramma "TESTER.C". Met LabWindows kunnen ook grafieken gemaakt worden, maar vergeleken met Lotus 123 zijn de mogelijkheden voor het toevoegen van tekst geringer, zodat men voor grafieken die gepubliceerd moeten worden beter gebruik kan maken van Lotus 123.

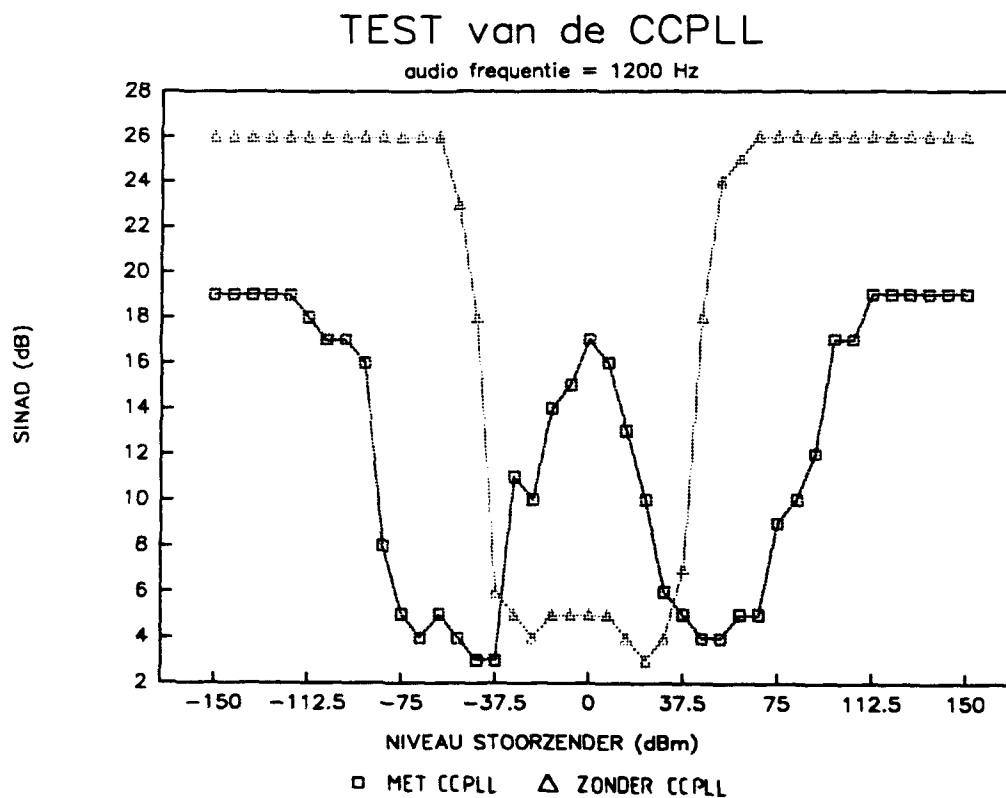


Fig. 7

TEST van de CCPLL

audio frequentie = 1200 Hz

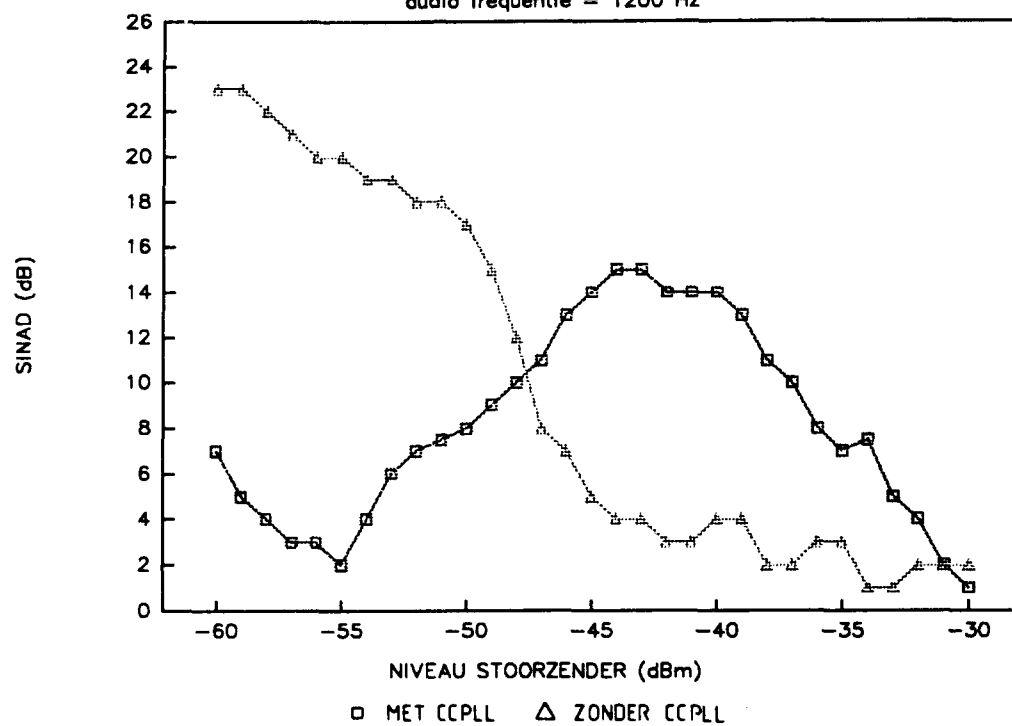
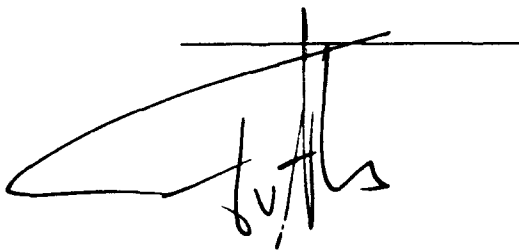


Fig. 8

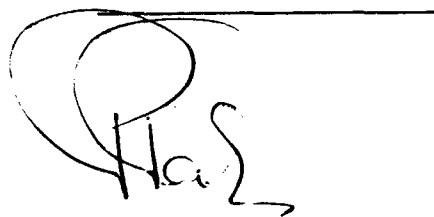
5 CONCLUSIE

Met de realisatie van dit meetsysteem beschikt het FEL over de mogelijkheid in korte tijd metingen uit te voeren, waarmee de kwaliteit van informatie-overdracht van radio-installaties bepaald kan worden. De gerealiseerde testopstelling is een eerste aanzet, het is nu mogelijk communicatie-apparatuur te testen in een gesimuleerde radioverbinding met een stoorzender of ruisbron. De groep Telecommunicatie heeft het voornemen de mogelijkheden van deze testopstelling geleidelijk te vergroten. Toegevoegd kunnen worden: de mogelijkheid om BER-metingen aan digitale radiosystemen uit te voeren en de simulatie van fading en meerwegpropagatie.

Het programma LabWindows bleek een geschikt hulpmiddel bij het ontwikkelen en gebruik van de programma's voor de controle van de meetinstrumenten en de analyse van de meetresultaten. Ook een grafische presentatie van de meetresultaten is met LabWindows snel en eenvoudig te verkrijgen.

A handwritten signature in black ink, featuring a large, sweeping loop on the left and several vertical strokes on the right, crossing a horizontal line.

Ir. F.G.J. van Aken
(groepsleider)

A handwritten signature in black ink, starting with a large, circular loop on the left, followed by several vertical strokes and a final flourish on the right, crossing a horizontal line.

P. Feenstra
(auteur)

DE BEDIENING VAN DE MEETOPSTELLING

A1 Woord vooraf

Deze beschrijving van de bediening gaat uit van de in de testinstallatie opgenomen COMPAQ model 2, personal computer. Besturing door een ander type PC is natuurlijk mogelijk mits aan de volgende voorwaarden is voldaan.

- a. Op de PC moet zijn geïnstalleerd het LabWindows ontwikkelpakket versie 1.2 van National Instruments.
- b. In de bibliotheek (library) van LabWindows moet onder de directory "Instrument" het programmamodule "TESTER" (bestaande uit de files "TESTER.FP" en "TESTER.LBW") zijn opgenomen en voor het presenteren van de meetresultaten via de plotter moet in de bibliotheek onder de directory "Programs" het programma "GRAFIEK.C" zijn opgenomen.

De vijf meetinstrumenten en eventueel de plotter moeten via de GPIB connectors aangesloten zijn met de PC.

De volgende beschrijving veronderstelt dat geen muis is geïnstalleerd. Is wel een muis aanwezig dan zijn alle commando's in de volgende beschrijving ook eenvoudig daarmee te selecteren; de beschreven commandoselectie met behulp van de <ALT> toets en oriëntatietoetsen is dan overbodig.

A2 De bediening

1. Schakel alle instrumenten aan.
2. Start LabWindows.
 - a. type <CD LW> en <return>.
 - b. type <LW> en <return>.
3. Laad het instrument-module TESTER.
 - a. type <ALT><I> en <Enter>, een nieuw venster verschijnt.
 - b. ga met de tabtoets naar de "list box", selecteer hierin met de oriëntatietoetsen "INSTRUMENTS" en type <Enter>.

Opm. Bij gebruik van de in de testopstelling aanwezige COMPAQ PC kan dit punt worden overgeslagen, omdat hierbij tijdens het opstarten van LabWindows al automatisch de instrumentmodule "TESTER" wordt geladen.

4. Selectie van de funktiepanels voor het besturingsprogramma.
 - a. Selecteer op de barbalk "INSTRUMENTS" en type <Enter>
Er verschijnt nu een dialoogvenster met daarin de namen van de zes panels (function panel tree).
 - b. Selecteer het eerste panel "Initialize" en type <Enter>. Het "initialize" panel verschijnt.

5. Panel 1
"Initialize"

Aan de linker kant van dit panel zijn vijf vensters voor de GPIB-adressen van de respectievelijke meetinstrumenten. In deze vensters zijn al default waarden ingevuld. Kloppen deze niet, vul dan het korrekte adres in door de markering m.b.v. de tabtoets op het gewenste venster te zetten en het adres in dit venster in overeenstemming te brengen met het GPIB-adres van het bijbehorende meetinstrument.

Selecteer op de commandobar "Go" door <ALT><g> te typen.

Na het "runnen" moeten de vijf errorvensters "0" gebleven zijn. Is dit niet het geval, hef dan eerst de storing op, voordat verder wordt gegaan. Het getal vermeld in het errorveld is een aanwijzing voor het type fout. Via de helpfunctie verkrijgt men een beschrijving van deze fout.

Type nu tegelijk de toetsen <shift> <PgDn>, waarna het tweede panel verschijnt.

GO: KEEP: Instruments Libraries View Options Help Return:

Tester Instel_radiokan. Page 1/1

SENDER NIVEAU -40 dBm

STOORBRON
RUIS
STOORZ R+S

MEETFREQUENTIE 25 MHz

MODULATIE
CW
FM
AM

FM-INDEX 5 KHz

AM-INDEX 70 %

ME FILTER
1 MHz
120 KHz
12 KHz
7.5 KHz

DEMOD
AM
AM-3KHz
FM
FM-3KHz

ERROR 0

TESTER_Radio (25,2,3,-40,1,5,70,1) :

6. Panel 2

"Instel_radiokanaal"

Via de vensters van dit panel kunnen de frequentie van het radiokanaal (zender en ontvanger) en de modulatie van de zender ingesteld worden. De vensters spreken voor zichzelf. Breng met de tabtoets de marker op het gewenste venster en wijzig de inhoud naar wens. In dit panel bevindt zich ook een venster waarmee de gewenste stoorbron gekozen kan worden. De mogelijkheden zijn: stoorzender, ruisgenerator of beide.

Type <ALT><g> en vervolgens <shift><PgDn>. Het volgende panel verschijnt.

GO: KEEP: Instruments Libraries View Options Help Return:

Tester Instel_ruis Page 1/1

START NIVEAU -80

STAP 5

STOP NIVEAU -20

ERROR 0

TESTER_Ruis (-80,-20,5) :

7. Panel 3

"Instel_ruisgenerator"

Heeft men in panel 2 "stoorzender" geselecteerd, ga dan met <shift><PgDn> door naar het volgende panel.

Dit panel stelt de ruisniveaus in het radiokanaal in, waarbij SINAD-waarden gemeten zullen worden. Vul de drie vensters in voor resp. het minimale ruisniveau, het maximale ruisniveau en de stapgrootte of ruisniveautoename per meting. De drie waarden moeten gehele getallen zijn.

Is "stoorzender + ruis" gekozen in panel 2, dan moet hier alleen het beginniveau worden ingevuld.

Type weer eerst <ALT><g> en daarna <shift><PgDn> voor het volgende panel.

8. Panel 4 "Instel_stoorzender"

Dit panel stelt de stoorzender in. Is in panel 2 "ruis" gekozen, ga kan met <shift><PgDn> direkt door naar het volgende panel.

Kies eerst via het venster "stoormode" tussen resp. a-"Frequentie", d.w.z. een metingserie met variabele frequentie bij een constant niveau van de stoorzender, of b- "Niveau", een metingserie met variabel niveau en constante frequentie. Vul vervolgens naar wens de begin- en eindwaarde van de frequentie- of niveaureeks in en het aantal metingen. De stapgrootte wordt uit deze gegevens berekend.

Tenstote kan de modulatiesoort en index van de stoorzender gekozen worden.

Type <ALT><g> en <shift><PgDn> voor het volgende panel.

GO: KEEP: Instruments Libraries View Options Help Return:

Tester SINAD_meting Page 1/1

FREQ 1 500 Hz

FREQ 2 800 Hz

FREQ 3 1200 Hz

FREQ 4 2400 Hz

LF NIVEAU 1000 mV

BAND-FILTER "A"

LD-FILTER 30 KHz

METING

REFERENTIE

ERROR 0

TESTER_Sinad (1000,500,800,1200,2400,2,1,1) :

9. Panel 5 "SINAD-meting"
"SINAD_meting"

Stel de vier audiofrequenties in waarop SINAD-waarden gemeten moeten worden. Selecteer eventueel ook een filter en weegfactor. Selecteer ook de "meting" of "referentie". In het eerste geval zullen de meetresultaten geschreven worden naar de files "data0" t/m "data4" en in het tweede geval naar de files "ref0" t/m "ref4". Bestaande meetresultaten in de files worden overschreven!

Type <ALT><g>, nu wordt de meting uitgevoerd volgens dit en vorige panels ingevoerde parameters. De tijd nodig voor de uitvoering van alle metingen zal afhankelijk van het aantal meetstappen enige minuten in beslag kunnen nemen. Pas als alle metingen zijn uitgevoerd kan met <shift><PgDn> naar het volgende panel worden gegaan.

GO! KEEP! Instruments Libraries View Options Help Return!

Tester Close Page 1/1

ESVP ERROR 0

SINAD ERROR 0

STOORE ERROR 0

RENDER ERROR 0

NOIS ERROR 0

TESTER_Close () :

10. Panel 6

"Close"

Type <Alt><g>. Hierna zijn alle GPIB-poorten weer vrij.

A3 Grafiek

Aanvullend aan de meting kan het resultaat grafisch worden bekeken.

Type <Esc> en selecteer met <Alt><F> in de commandobar "Files". Selecteer door <I> en <Enter> te typen "load".

Selecteer in het direktorie veld "programs". Type daarvoor tweemaal <tab> en vervolgens <p> en <Enter>.

Selecteer met behulp van de oriëntatietoetsen nu "grafiek.c" en type <Enter>, start vervolgens door <control> <R> het programma.

Indien een plotter bedrijfsklaar is aangesloten kan met de funktietoets <f4> een plot gemaakt worden.

PRINTOUTS

B1 Printout van het besturingsprogramma "TESTER.C"

```
/*== TESTER-Instrument Module ==*/  
/*==4 FEBRUARI 1991==*/
```

```
#include "lw\include\lwsystem.h"  
#include "lw\include\gpib.h"  
#include "lw\include\formatio.h"  
#include "TESTER.h"
```

```
int TESTER_read_data();  
int TESTER_write_data();  
int TESTER_device_closed();  
int TESTER_invalid_real_range();  
int TESTER_invaled_integer_range();
```

```
static int bd1;  
static int bd2;  
static int bd3;  
static int bd4;  
static int bd5;  
static int bdx;  
static char cmd[26];  
static int TUDRL_err;  
static int TESTER_RUISSTAP;  
static double TESTER_FREQSTAP;
```

```
void TESTER_Init (ESVPadres,ZENDERadres,SINADadres,RUISadres,STOORZENDadres)
```

```
int ESVPadres ;
```

```
int SINADadres ;
```

```
int ZENDERadres ;
```

```
int RUISadres ;
```

```
int STOORZENDadres ;
```

```
{ int addr ;
```

```
    addr = ESVPadres ;
```

```
    if (TESTER_invalid_integer_range (addr, 0, 30, -1) !=0)
```

```
    { TESTER_ESVP_err = -1 ; return ;
```

```
    }
```

```
    if (bd1 <= 0)
```

```
    { CloseInstrDevs("ESVP") ;
```

```
      bd1 = OpenDev ("","ESVP") ;
```

```
      if (bd1 <= 0)
```

```
      { TESTER_ESVP_err =220 ;
```

```
        return ;
```

```
      }
```

```
      if (ibpad (bd1,addr) < 0)
```

```
      { TESTER_ESVP_err = 233 ;
```

```
        return ;
```

```
      }
```

```
    addr = ZENDERadres ;
```

```
    if (TESTER_invalid_integer_range (addr, 0, 30, -1) !=0)
```

```
    { TESTER_ZEND_err = -1 ; return ;
```

```
    }
```

```
    if (bd2 <= 0)
```

```
    { CloseInstrDevs("ZENDER") ;
```

```
      bd2 = OpenDev ("","ZENDER") ;
```

```
if (bd2 <= 0)
{
    TESTER_ZEND_err = 220;
    return;
}
if (ibpad (bd2,addr) < 0)
{
    TESTER_ZEND_err = 233;
    return;
}

addr = SINAdres;
if (TESTER_invalid_integer_range (addr, 0, 30, -1) != 0)
{
    TESTER_SINAD_err = -1; return;
}

if (bd3 <= 0)
{
    CloseInstrDevs("SINAD");
    bd3 = OpenDev ("","SINAD");
    if (bd3 <= 0)
    {
        TESTER_SINAD_err = 220;
        return;
    }
    if (ibpad (bd3,addr) < 0)
    {
        TESTER_SINAD_err = 233;
        return;
    }
}

addr = RUISadres;
if (TESTER_invalid_integer_range (addr, 0, 30, -1) != 0)
{
    TESTER_RUIS_err = -1; return;
}

if (bd4 <= 0)
{
    CloseInstrDevs("RUIS");
    bd4 = OpenDev ("","RUIS");
```

```
    if (bd4 <= 0)
    { TESTER_RUIS_err = 220 ;
      return ;
    }
    if (ibpad (bd4,addr) < 0)
    { TESTER_RUIS_err = 233 ;
      return ;
    }

    addr = STOOZENDadres ;
    if (TESTER_invalid_integer_range (addr, 0, 30, -1) != 0)
    { TESTER_RUIS_err = -1 ; return ;
    }

    if (bd5 <= 0)
    { CloseInstrDevs("STOORZ") ;
      bd5 = OpenDev ("","STOORZ") ;
      if (bd5 <= 0)
      { TESTER_STOORZ_err = 220 ;
        return ;
      }
    }
    if (ibpad (bd5,addr) < 0)
    { TESTER_STOORZ_err = 233 ;
      return ;
    }
    TESTER_err = 0 ;
}
```

```
void TESTER_Radio(HFFREQ,MF,DEMODO,ZNIV,MOD,FM,AM,bron)
double HFFREQ, FM ;
int MF, ZNIV, MOD, AM ;
int DEMODO,bron ;
```

```
{  
    char *MFstr, *demodstr;  
  
    TESTER_STOORMODE = bron;  
  
    switch (MF)  
    { case 0 : MFstr = "B1"; break;  
      case 1 : MFstr = "B2"; break;  
      case 2 : MFstr = "B3"; break;  
      case 3 : MFstr = "B4"; break;  
    }  
  
    switch (DEMOMD)  
    { case 0 : demodstr = "D3"; break;  
      case 1 : demodstr = "D4"; break;  
      case 2 : demodstr = "D7"; break;  
      case 3 : demodstr = "D8"; break;  
    }  
  
    Fmt (cmd, "SF00,%s,%s\n", MFstr, demodstr);  
    bdx = bdl; Scan (cmd, "%s[di0t0]");  
    TESTER_write_data (bdx, cmd, NumFmtdBytes());  
  
    Fmt (cmd, "FR%f[p4]\n", HFFREQ);  
    bdx = bdl; Scan (cmd, "%s[di0t0]");  
    TESTER_write_data (bdx, cmd, NumFmtdBytes());  
  
    Fmt (cmd, "CF%f[p3]MZ\n", HFFREQ);  
    bdx = bd2;  
    TESTER_write_data (bdx, cmd, NumFmtdBytes());  
  
    Fmt (cmd, "LV%dDB,C1\n", ZNIV);  
    bdx = bd2;  
    TESTER_write_data (bdx, cmd, NumFmtdBytes());
```

```
if (MOD == 0) Fmt (cmd, "M0\n");
if (MOD == 1) Fmt (cmd, "FM,%f[p1],KZ,XM,M1\n", FM);
if (MOD == 2) Fmt (cmd, "AM,%d,PC,XM,M1\n", AM);
TESTER_write_data (bdx, cmd, NumFmtdBytes());

}
```

```
void TESTER_Stoorr(start, stop, stap)
```

```
int start, stop, stap;
```

```
{ int aantal;
```

```
if (TESTER_STOORMODE==0)
```

```
{ if (start > stop) aantal = ((start-stop)/stap)+1;
```

```
else aantal = ((-start+stop)/stap)+1; stap = -stap;
```

```
}
```

```
if (TESTER_STOORMODE == 0)
```

```
{ TESTER_RUISBEGIN = -start;
```

```
TESTER_RUISSTAP = stap;
```

```
TESTER_RUISEIND = -stop;
```

```
TESTER_AANTAL = aantal;
```

```
Fmt (cmd, "2P%dLL\n", -1*start);
```

```
bdx = bd4;
```

```
TESTER_write_data (bdx, cmd, NumFmtdBytes());
```

```
}
```

```
if (TESTER_STOORMODE == 1)
```

```
{ TESTER_RUISBEGIN = 0;
```

```
TESTER_RUISEIND = 0;
```

```
TESTER_AANTAL = aantal;
```

```
Fmt (cmd, "2P,99L,0\n");
```

```
bdx = bd4;
```

```

    TESTER_write_data (bdx, cmd, NumFmtdBytes());
}

if (TESTER_STOORMODE == 2)
{ Fmt (cmd, "2F,%dL\n",-1*start);

  bdx = bd4;

  TESTER_write_data (bdx, cmd, NumFmtdBytes());
}

void TESTER_Stoorz(modu,fm,am,mode,frstart,frstop,aantal,nistart,nistop,lffr)

int modu, am, mode, aantal, nistart, nistop;

int lffr;

double fm, frstart,frstop;

{ char *xlf;

  bdx=bd5;

  switch (lffr)

  { case 0 :xlf = "M0"; break;

    case 1 :xlf = "M1"; break;

    case 2 :xlf = "M2"; break;

    case 3 :xlf = "M3"; break;

    case 4 :xlf = "M4"; break;

  }

  if (modu == 0) Fmt(cmd, "FM OF,AM OF\n");

  if (modu == 1) Fmt(cmd, "FM,ON,%f[p1]KZ,IT,%s\n",fm,xlf);

  if (modu == 2) Fmt(cmd, "AM,ON,%dPC,IT,%s\n",am,xlf);

  TESTER_write_data (bdx,cmd,NumFmtdBytes());

  if (TESTER_STOORMODE == 0)

  { Fmt (cmd,"CP%f[p3]MZ,LV%dDB,OF\n",frstart,nistart);

    TESTER_write_data (bdx,cmd,NumFmtdBytes());

  }

```

```
if (TESTER_STOORMODE != 0)
{ if (mode == 1)
{ TESTER_AANTAL = aantal ;
  TESTER_STOORFREQBEGIN = frstart ;
  TESTER_STOORFREQSTAP = (frstop-frstart) / aantal ;
  TESTER_STOORFREQEIND = frstop ;
  TESTER_STOORNIVBEGIN = nistart ;
  TESTER_FREQOFNIV = 1 ;
  Fmt (cmd, "CF%f[p4]MZ,LV%dDB,ON\n",frstart,nistart) ;
  TESTER_write_data (bdx,cmd,NumFmtdBytes()) ;
}
else
{ TESTER_AANTAL = aantal ;
  TESTER_STOORFREQBEGIN = frstart ;
  TESTER_STOORNIVSTAP = (nistop-nistart) / aantal ;
  TESTER_STOORNIVBEGIN = nistart ;
  TESTER_STOORNIVEIND = nistop ;
  TESTER_FREQOFNIV = 0 ;
  Fmt (cmd, "CF%f[p4]MZ,LV%dDB,ON\n",frstart,nistart) ;
  TESTER_write_data (bdx,cmd,NumFmtdBytes()) ;
} } }
```

```
void TESTER_Sinad (niv,FR1,weeg,kdf,REF)
```

```
int niv, FR1,weeg, kdf, REF ;
```

```
{ int FR, lus1, lus2, t ;
```

```
int ruismiv ;
```

```
double ampl ;
```

```
double freq ;
```

```
double result ;
```


D. ga met de tabtoets naar de list box, selecteer item met de waarde "INSTRUMENTS" en type <Enter>.

TNO rapport

Bijlage B

Pagina
B.9

```
char *weegstr, *ldfstr;
double dotarray[102];
int Schijf1, Schijf2;

bdx = bd3;
Fmt (cmd, "M2,L2,LG,RR,AP%dMV\n", niv);
TESTER_write_data (bdx, cmd, NumFmtdBytes());
if (weeg==0) weegstr = "H0";
if (weeg==1) weegstr = "H2";
if (weeg==2) weegstr = "H1";
if (ldf==0) ldfstr = "L0";
if (ldf==1) ldfstr = "L1";
if (ldf==2) ldfstr = "L2";
Fmt (cmd, "%s%s\n", weegstr, ldfstr);
TESTER_write_data (bdx, cmd, NumFmtdBytes());

TESTER_SINADTABEL[0][0] = 0;
TESTER_SINADTABEL[0][1] = FR1;
TESTER_SINADTABEL[0][2] = 2222;
TESTER_SINADTABEL[0][3] = 3333;
TESTER_SINADTABEL[0][4] = 4444;

if (TESTER_STOORMODE == 0)
{ for (lus1=0; lus1<=(TESTER_AANTAL-1); lus1++)
{ ruismiv = (TESTER_RUISBEGIN + lus1*TESTER_RUISSTAP);
Fmt (cmd, "%d\n", ruismiv);
bdx = bd4;
TESTER_write_data (bdx, cmd, NumFmtdBytes());
TESTER_SINADTABEL[lus1+1][0] = ruismiv;

FR=FR1;
Fmt (cmd, "FR%dHZ,T3\n", FR);
bdx = bd3;
TESTER_write_data (bdx, cmd, NumFmtdBytes());
```

```
TESTER_read_data (bdx, cmd, 11);  
Scan (cmd, "%f", &result);  
if ((result<0)||(result>100)) result = 1313;  
  
TESTER_SINADTABEL[lus1+1][1]= result;  
TESTER_SINADTABEL[lus1+1][2]= result;  
TESTER_SINADTABEL[lus1+1][3]= result;  
TESTER_SINADTABEL[lus1+1][4]= result;  
}  
  
if (TESTER_STOORMODE != 0)  
{ if (TESTER_FREQOFNIV == 1)  
{ for (lus1=0; lus1<=TESTER_AANTAL; lus1++)  
{ freq = TESTER_STOORFREQBEGIN + lus1*TESTER_STOORFREQSTAP;  
Fmt (cmd, "CF%f[p5]MZN", freq);  
bdx = bd5;  
TESTER_write_data (bdx, cmd, NumFmtdBytes());  
TESTER_SINADTABEL[lus1+1][0] = freq;  
  
FR=FR1;  
  
Fmt (cmd, "FR%dHZ,T3n", FR);  
bdx = bd3;  
TESTER_write_data (bdx, cmd, NumFmtdBytes());  
TESTER_read_data (bdx, cmd, 11);  
Scan (cmd, "%f", &result);  
if ((result<0)||(result>100)) result = 1313;  
TESTER_SINADTABEL[lus1+1][1]= result;  
TESTER_SINADTABEL[lus1+1][2]= result;  
TESTER_SINADTABEL[lus1+1][3]= result;  
TESTER_SINADTABEL[lus1+1][4]= result;  
}}}
```

```

if (TESTER_STOORMODE != 0)
{ if (TESTER_FREQOFNIV == 0)
{ for (lus1=0; lus1 <= TESTER_AANTAL; lus1++)
{ ampl = TESTER_STOORNIVBEGIN + lus1*TESTER_STOORNIVSTAP ;
Fmt (cmd, "LV%f[p1]DB,ON\n", ampl) ;

bdx = bd5 ;

TESTER_write_data (bdx, cmd, NumFmtdBytes());
TESTER_SINADTABEL[lus1+1][0] = ampl ;

FR=FR1;

Fmt (cmd, "FR%dHZ,T3\n", FR) ;

bdx = bd3 ;

TESTER_write_data (bdx, cmd, NumFmtdBytes());
TESTER_read_data (bdx, cmd, 11) ;

Scan (cmd, "%f", &result) ;

if ((result<0)|| (result>100)) result =1313 ;

TESTER_SINADTABEL[lus1+1][1]= result ;
TESTER_SINADTABEL[lus1+1][2]= result ;
TESTER_SINADTABEL[lus1+1][3]= result ;
TESTER_SINADTABEL[lus1+1][4]= result ;

} } }

```

```

TESTER_SINADTABEL[98][0] = TESTER_STOORMODE ;
TESTER_SINADTABEL[98][1] = TESTER_FREQOFNIV ;
TESTER_SINADTABEL[98][2] = TESTER_AANTAL ;
TESTER_SINADTABEL[98][3] = TESTER_RUISBEGIN ;
TESTER_SINADTABEL[98][4] = TESTER_RUISEIND ;
TESTER_SINADTABEL[99][0] = TESTER_STOORFREQBEGIN ;
TESTER_SINADTABEL[99][1] = TESTER_STOORFREQEIND ;
TESTER_SINADTABEL[99][2] = TESTER_STOORNIVBEGIN ;
TESTER_SINADTABEL[99][3] = TESTER_STOORNIVEIND ;

```

```
if (REF == 1)
{
    Schijf1=OpenFile ("c:/tw/data0.prn",0,0,1);
    for (lus1=0;lus1<=99;lus1++)
    {
        dotarray[lus1] = TESTER_SINADTABEL[lus1][0];
    }
    FmtFile (Schijf1,"%100f[w10]",dotarray);
    CloseFile (Schijf1);

    Schijf1=OpenFile ("c:/tw/data1.prn",0,0,1);
    for (lus1=0;lus1<=99;lus1++)
    {
        dotarray[lus1] = TESTER_SINADTABEL[lus1][1];
    }
    FmtFile (Schijf1,"%100f[w10]",dotarray);
    CloseFile (Schijf1);

    Schijf1=OpenFile ("c:/tw/data2.prn",0,0,1);
    for (lus1=0;lus1<=99;lus1++)
    {
        dotarray[lus1] = TESTER_SINADTABEL[lus1][2];
    }
    FmtFile (Schijf1,"%100f[w10]",dotarray);
    CloseFile (Schijf1);

    Schijf1=OpenFile ("c:/tw/data3.prn",0,0,1);
    for (lus1=0;lus1<=99;lus1++)
    {
        dotarray[lus1] = TESTER_SINADTABEL[lus1][3];
    }
    FmtFile (Schijf1,"%100f[w10]",dotarray);
    CloseFile (Schijf1);

    Schijf1=OpenFile ("c:/tw/data4.prn",0,0,1);
    for (lus1=0;lus1<=99;lus1++)
    {
        dotarray[lus1] = TESTER_SINADTABEL[lus1][4];
    }
}
```

```
    FmtFile (Schijf1,"%100f[w10]",dotarray);  
    CloseFile (Schijf1);  
}
```

```
if (REF == 0)  
{ Schijf1=OpenFile ("c:/tw/ref0.prn",0,0,1);  
  for (lus1=0;lus1<=99;lus1++)  
  { dotarray[lus1] = TESTER_SINADTABEL[lus1][0];  
  }  
  FmtFile (Schijf1,"%100f[w10]",dotarray);  
  CloseFile (Schijf1);
```

```
  Schijf1=OpenFile ("c:/tw/ref1.prn",0,0,1);  
  for (lus1=0;lus1<=99;lus1++)  
  { dotarray[lus1] = TESTER_SINADTABEL[lus1][1];  
  }  
  FmtFile (Schijf1,"%100f[w10]",dotarray);  
  CloseFile (Schijf1);
```

```
  Schijf1=OpenFile ("c:/tw/ref2.prn",0,0,1);  
  for (lus1=0;lus1<=99;lus1++)  
  { dotarray[lus1] = TESTER_SINADTABEL[lus1][2];  
  }  
  FmtFile (Schijf1,"%100f[w10]",dotarray);  
  CloseFile (Schijf1);
```

```
  Schijf1=OpenFile ("c:/tw/ref3.prn",0,0,1);  
  for (lus1=0;lus1<=99;lus1++)  
  { dotarray[lus1] = TESTER_SINADTABEL[lus1][3];  
  }  
  FmtFile (Schijf1,"%100f[w10]",dotarray);  
  CloseFile (Schijf1);
```

```
Schijf1=OpenFile ("c:/tw/ref4.prn",0,0,1);  
for (lus1=0;lus1<=99;lus1++)  
{ dotarray[lus1] = TESTER_SINADTABEL[lus1][4];  
}  
FmtFile (Schijf1,"%100f[w10]",dotarray);  
CloseFile (Schijf1);  
}
```

```
void TESTER_close()
```

```
{ int control; control = 0;  
  bdx = bd1;  
  if (TESTER_device_closed())  
  { control = 1;  
    TESTER_ESVP_err = 232;  
  }
```

```
  bdx = bd2;  
  if (TESTER_device_closed())  
  { control = 1;  
    TESTER_ZEND_err = 232;  
  }
```

```
  bdx = bd3;  
  if (TESTER_device_closed())  
  { control = 1;  
    TESTER_SINAD_err = 232;  
  }
```

```
  bdx = bd4;  
  if (TESTER_device_closed())
```

```
{ control = 1 ;  
  TESTER_STOORZ_err = 232 ;  
}  
  
bdx = bd5 ;  
if (TESTER_device_closed())  
{ control = 1 ;  
  TESTER_RUIS_err = 232 ;  
}  
if (control == 0)  
return ;  
  
if (CloseDev (bd1) < 0 )  
  TESTER_ESVP_err = 221 ;  
else TESTER_ESVP_err = 0 ;  
bd1 = 0 ;  
if (CloseDev (bd2) < 0 )  
  TESTER_ZEND_err = 221 ;  
else TESTER_ZEND_err = 0 ;  
bd2 = 0 ;  
if (CloseDev (bd3) < 0 )  
  TESTER_SINAD_err = 221 ;  
else TESTER_SINAD_err = 0 ;  
bd3 = 0 ;  
if (CloseDev (bd4) < 0 )  
  TESTER_RUIS_err = 221 ;  
else TESTER_RUIS_err = 0 ;  
bd4 = 0 ;  
if (CloseDev (bd5) < 0 )  
  TESTER_STOORZ_err = 221 ;  
else TESTER_STOORZ_err = 0 ;  
bd5 = 0 ;  
}
```

```
int TESTER_invalid_integer_range (val,min,max,err_code)
```

```
int val,min,max,err_code;
```

```
{ if ((val < min) || (val > max))
```

```
{ TUDEL_err = err_code;
```

```
  return 1;
```

```
}
```

```
return 0;
```

```
}
```

```
int TESTER_invalid_real_range (val,min,max,err_code)
```

```
double val,min,max;
```

```
int err_code;
```

```
{
```

```
  if ((val < min) || (val > max))
```

```
  { TUDEL_err = err_code;
```

```
    return 1;
```

```
  }
```

```
  return 0;
```

```
}
```

```
int TESTER_write_data (bdx,cmd,cnt)
```

```
char *cmd;
```

```
int cnt;
```



```
{ int x ;  
  x=cnt ;  
  FmtOut("%s",cmd) ;  
  breakpoint();  
  x=cnt ;  
  if (ibwrt(bdx, cmd, cnt) <= 0)  
    TESTER_err = 230;  
  else  
    TESTER_err = 0;  
  return TESTER_err;  
}
```

```
int TESTER_read_data (bdx,cmd,cnt)  
char *cmd ;  
int cnt ;
```

```
{ if (ibrd(bdx, cmd, cnt) <= 0)  
  TESTER_err =231;  
  else  
  { TESTER_err = 0;  
    FmtOut ("##%s",cmd) ;  
  }  
  return TESTER_err ;  
}
```

```
int TESTER_device_closed()  
{ if (bdx <= 0)  
  { TLJDEL_err =232 ;  
    return 1 ;  
  }  
}
```

```
return 0;
```

```
}
```

```
/*= THE END =====*/
```

B2 Printout van de include file "TESTER.H"

```
/*= CCPLL TESTER Include File =====*/
```

```
/*= GLOBAL FUNCTION DECLARATIONS =====*/
```

```
void TESTER_Init();  
void TESTER_Radio();  
void TESTER_Stoor();  
void TESTER_Stoorz();  
void TESTER_Sinad();  
void TESTER_close();
```

```
/*= GLOBAL VARIABLE DECLARATIONS =====*/
```

```
/* Global error variable for the instrument module */
```

```
int TESTER_err;  
int TESTER_ESVP_err;  
int TESTER_ZEND_err;  
int TESTER_SINAD_err;  
int TESTER_RUIS_err;  
int TESTER_STOORZ_err;
```

```
/*=GLOBAL VARIABLE DECLARATIONS=====*/
```

```
int TESTER_STOORMODE; /* ruis=1 stoorzender=0 */  
int TESTER_FREQOPNIV; /* freq=0 niveau=1 */  
int TESTER_RUISBEGIN;  
int TESTER_RUISEIND;  
int TESTER_AANTAL;  
double TESTER_STOORFREQBEGIN;  
double TESTER_STOORFREQEIND;
```

```
double TESTER_STOORFREQSTAP;  
int TESTER_STOORNIVBEGIN;  
int TESTER_STOORNIVEIND;  
int TESTER_STOORNIVSTAP;  
double TESTER_SINADTABEL[100][6];  
double TESTER_REFTABEL[100][6];
```

UNCLASSIFIED

REPORT DOCUMENTATION PAGE

(MOD-NL)

1. DEFENSE REPORT NUMBER (MOD-NL) TD91-2232	2. RECIPIENT'S ACCESSION NUMBER	3. PERFORMING ORGANIZATION REPORT NUMBER FEL-91-A171
4. PROJECT/TASK/WORK UNIT NO.20561 20561	5. CONTRACT NUMBER A89KL641	6. REPORT DATE AUGUST 1991
7. NUMBER OF PAGES 53 (INCL. 2 APPENDICES + RDP, EXCL. DISTRIBUTION LIST)	8. NUMBER OF REFERENCES -	9. TYPE OF REPORT AND DATES COVERED INTERIM
10. TITLE AND SUBTITLE EEN MEETSISTEEM VOOR HET TESTEN VAN RADIOCOMMUNICATIE-APPARATUUR (A MEASURING FACILITY FOR TESTING OF RADIO COMMUNICATION EQUIPMENT)		
11. AUTHOR(S) P. FEENSTRA		
12. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) TNO PHYSICS AND ELECTRONICS LABORATORY, P.O. BOX 96864, 2509 JG THE HAGUE OUDER WAALSDORPERWEG 63, THE HAGUE, THE NETHERLANDS		
13. SPONSORING/MONITORING AGENCY NAME(S) HWO-KL, P.O. BOX 90701, 2500 ES THE HAGUE, THE NETHERLANDS		
14. SUPPLEMENTARY NOTES		
15. ABSTRACT (MAXIMUM 200 WORDS, 1044 POSITIONS) THIS REPORT DESCRIBES A TEST FACILITY FOR MEASURING THE BEHAVIOUR AND QUALITY OF RADIO COMMUNICATION EQUIPMENT IN A SIMULATED OPERATIONAL ENVIRONMENT. THE WORK AS DESCRIBED IN THIS REPORT WAS PERFORMED IN THE SCOPE OF THE ASSIGNMENT "ECCM JAMMER SUPPRESSION" (A89KL641). THE TESTBED ALLOWS TO SELECT THE SIMULATED PROPAGATION CHARACTERISTICS, SUCH AS ATTENUATION, FADING AND THE OCCURRENCE OF MULTIPATH EFFECTS. BESIDES IT IS POSSIBLE TO INSERT VARIOUS JAMMING SIGNALS. THE CRITERION FOR CLASSIFYING RADIO EQUIPMENT UNDER TEST IS THE QUALITY OF TRANSFERRED INFORMATION, THAT IS THE SINAD VALUE IN AN ANALOG SYSTEM AND THE BER FOR A DIGITAL SYSTEM. THE TEST SET-UP IS CONTROLLED BY A PC, THE OPERATION IS EASY BY USE OF WINDOWS. THIS TEST FACILITY CAN BE EMPLOYED FOR DEVELOPING NEW EQUIPMENT AND FOR EXAMINING SPECIFICATIONS OF RADIO COMMUNICATION EQUIPMENT IN EXISTENCE. WITH THIS TEST FORMATION FEL HAS THE DISPOSAL OF A FACILITY TO TEST THE QUALITY OF RADIO EQUIPMENT IN A SIMULATED OPERATIONAL SITUATION.		
16. DESCRIPTORS MEASUREMENT MEASURING INSTRUMENTS RADIO EQUIPMENT COMMUNICATION EQUIPMENT TEST MEASUREMENT		IDENTIFIERS
17a. SECURITY CLASSIFICATION (OF REPORT) UNCLASSIFIED	17b. SECURITY CLASSIFICATION (OF PAGE) UNCLASSIFIED	17c. SECURITY CLASSIFICATION (OF ABSTRACT) UNCLASSIFIED
18. DISTRIBUTION/AVAILABILITY STATEMENT UNLIMITED		17d. SECURITY CLASSIFICATION (OF TITLES) UNCLASSIFIED

UNCLASSIFIED