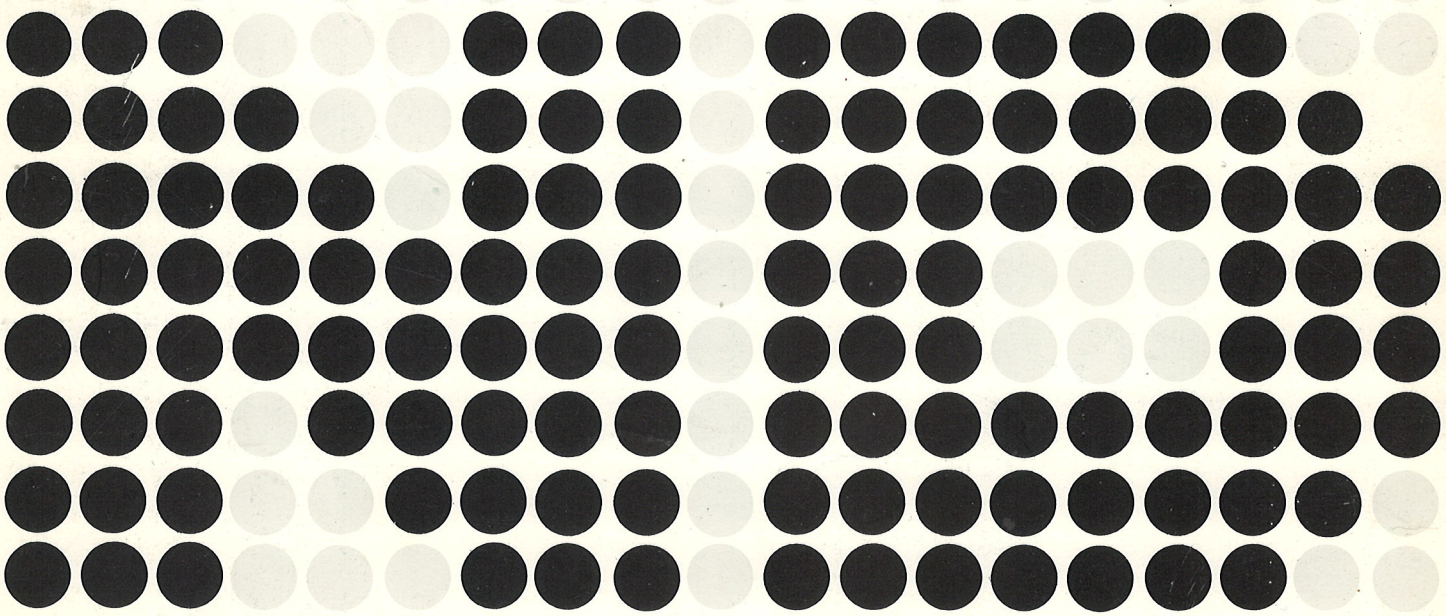
NORD - 50

Test System

NORSK DATA A.S



NORD - 50

Test System

REVISION RECORD

Revision	Notes
09/75	Original Printing
05/77	Revision A - Revised page 5-7, A-12, A-13, A-15, A-17.

The NORD-50 Test System - Reference Manual
Publication No. ND-62.008.01

A/S NORSK DATA-ELEKTRONIKK
Lørenvoien 57, Oslo 5 - Tlf.: 21 73 71

PURPOSE

This manual describes the NORD-50 test system. The NORD-50 test system is an integrated debugging system for the complete NORD-50 CPU and memory hardware.

This manual should be read together with the NORD-50 Reference Manual and the NORD-10 ⟷ NORD-50 Communication System.

--oOOoo--

TABLE OF CONTENTS

--oOOoo--

1 INTRODUCTION

The NORD-50 computer is under complete control of a NORD-10 computer. The debugging and maintenance of the NORD-50 is therefore based on the principle of having a computer testing another computer. This makes it possible to isolate an error, and loop on this error almost regardless of how serious the error is. This system is used in production of the NORD-50, starting with the simple tests to check data paths and ends with a complete test of the CPU and memory.

The system is designed to be an efficient debugging system. However, it is also used for ordinary maintenance and error checking.

The system requires a NORD-50 to be tested, a NORD-10 with preferably 16K memory and an alphanumeric display; (if this is not available, a Teletype may be used).

2 HOW TO USE THIS MANUAL

This manual is the reference manual for the NORD-5 test system and it should be used by:

a) The people who debug NORD-50 CPU's,

b) NORD-50 maintenance engineers,

c) The people doing modifications and extensions to the NORD-50 test system.

Readers in category a) and b) should read Chapters 4 and 5, and then use the Appendix when the system detects an error.

3 THEORY OF OPERATION

Before reading this chapter, you should have read the manual: The NORD-10/NORD-50 Communication System.

The NORD-50 is tested by supplying data and instructions from the NORD-10 program and then reading the result from this operation into the NORD-10 and comparing the result with a computed or stored correct result. When an error is detected, this is displayed to the operator, which then through the keyboard may analyze the error further.

For every error detected, the system also provides a complete loop so that the operator immediately may use the oscilloscope to locate the error. For every test the scope triggering signal is found in the Appendix.

Three different types of tests are supplied with the system:

a) Transfer Tests

This is typically a data path test. In a transfer test, the output or correct result equals the input data. The word length is 32 bits.

b) Two Argument Type Tests

This test uses two input operands called Argument A, and Argument B. The system compares the result of the NORD-5 with a correct result which is computed by the NORD-1. The word length is 32 bits.

c) Floating Point Tests

This is also two argument type tests, but the word length is 64 bits.

Input Data

The system contains a table of data, which is used for input during the tests. This table contains 64 32-bit words.

During transfer tests, both the data and the one's complement of data are used. Therefore each transfer test uses 128 different data words.

For two argument type test this table is used for all possible combinations of Argument A and Argument B, and also for all combinations of data and one's complement of data, giving a total of $64 \cdot 64 \cdot 4 = 16384$ combinations.

The floating point tests use special tables for each operation, where both 64 bit input operands are stored and the 64 bit correct result.

4 SYSTEM USAGE

The complete system is contained in a 16K NORD-10 program, which also includes an ordinary NORD-10 MAC assembler. After loading the system either from paper tape or from mass storage, the system is ready to be used. Select one of the preselected program tables, see Chapter 5, or write your own, see Section 4.2.

A preselected program table is read by typing $ to MAC, with the correct paper tape in the paper tape reader. The system will then display a message.

When everything is O.K., the message ALL AVAILABLE TESTS COMPLETED will occur after each complete cycle of the programs contained in the current test program table.

4.1 Display Commands

Messages will only be written on the display when an error has been detected or when all programs have passed the test successfully.

When an error has been detected, the program loops on this error (without any further tests of the error still being there), so that an oscilloscope may be used to isolate and fix the error. In this loop the operator may use the display keyboard to guide the tests further.

All commands are single letter commands. The following commands are used in the system:

D The command D gives further information about the current error; a short description and the name of the test program, followed by the code for the instruction. For example:

```
D
T45    ARGUMENT EXCLUSIVE OR
INSTRUCTION  10000000  10000001  01011110  01110111
```

C The command C, continue, continues the test with the next input data, if or when the next error occurs, the error is displayed again.

B The command B, back, will restart the particular test program, so that the input data is taken from the beginning of the input data table. This command is particularly useful when trying to determine whether an error is permanent or transient.

N The command N, next, will continue the tests starting with the next program in the program table.

R The command R, reset, will reset the program table pointer and the test will restart from the first program in the test table.

P The command P will test the previous program in the program table.

Note that the commands are echoed on the display. All other commands are ignored, but the echo may be used to erase the display screen if this is desired. Typically the display will be used in "Roll" mode.

4.2 The Program Table

The NORD-50 test system is controlled by a program table which contains the start addresses of all the test programs which are to be run.

The program table is a sequential table of these start addresses. The program table has room for $240_8 = 176$ different test programs. The program table should always be terminated with the termination program T99, which displays the message

ALL AVAILABLE TESTS COMPLETED

and then starts from the beginning of the program table again.

During the running of the NORD-50 test system the contents of the NORD-10, B-register are a pointer to the current program in the test table being executed.

An error message will always start with the sequence number (in octal) in the test table of the current test program.

All test programs are written as subroutines (called by the JPL I ,B instruction), they have the following specifications:

Transfer test type programs:

Input data	(A-D) register
Output data	(A-D) register
	(T), (X), (L) are lost
	(B) must be saved

Two argument type test programs:

Input data:	Operand A	(A-D) register
	Operand B	(X) is a pointer to operand B
Output data:	Result	(A-D) register
		(X) is a pointer to the computed correct result.
		(T), (L) lost, (B) must be saved.

The program table sequence should always start with the transfer type tests followed by 0 and again followed by the two argument type tests and terminated by T99.

The first zero in the program table tells the system that the following programs are two argument type programs. There must be at least one transfer type test in the program table.

Example of a correct program table:

```
PTAB/  T 0          % TRANSFER TEST
         0
       T20          % TWO ARGUMENT TEST
       T21          % TWO ARGUMENT TEST
       T99
```

Associated with the program table there are two other tables; one DISPT for a pointer to the name and description string of the test, and one IRTAB for a pointer to the current instruction being tested.

In the preselected program tables, all these tables are completed. If you are writing your own program tables, you may look into the Appendix to find the corresponding entries into these tables, or you can ignore the DISPT and IRTAB if you are not using the "D" command (If your are writing your own program tables, you are supposed to know what the programs are doing).

4.3 Bit Masking

A special feature has been built into the system so that during the compare test of the correct result and the achieved result, this test may be performed through a bit mask.

If for example, a test fails for one bit or a number of bits contained on the same PC board, the test may be run in a mode ignoring these bits by setting the corresponding bits in the MASK register to zero. The MASK register is contained in location:

MASK1	bit 31-16
MASK1+1	bit 15- 0

Normally the MASK contains a 1 in all 32 bits.

5 PRESELECTED PROGRAM TABLES

A set of program tables has been made so that it is easy to do a complete check-out of the machine, and also so that special purpose tables exist for example for specific tasks such as floating point debugging or memory debugging.

For most of the tests it is required that the LDR/STR instructions operate satisfactorily, because these are used to initialize a test and to read the results. The main arithmetic test should therefore be run before any other tests are run. If you know there is something wrong, but do not know what, try the fast complete test. See Section 5.10.

5.1 Main Arithemetic Tests

This program table consists of the following test programs:

PTAB/T0	% NORD-10/NORD-50 Communication
T1	% Cycle Counter Test
T2	% Datapaths
T3	% Address paths
T7	% LDR/STR
T7B	% LDR/STR
T7BREP	% LDR/STR for all register numbers
T10	% Register block test
T11	% Register zero test
T11B	% Transfer to/from overflow reg.
T11C	% Transfer to/from remainder reg.
T12	% LDF, Most Significant
T13	% LDF, Least significant
T14	% STF, Most significant
T15	% STF, Least significant
0	% Transfer to two argument tests
T20	% Register add
T21	% Register subtract
T22	% Register and
T23	% Register or
T24	% Register exclusive or
T25	% Register clear
T26	% Register and with complement of B
T27	% Register exclusive or with complement of A

```
T30     % Register or, with complements of A and B
T31     % Add
T32     % Sub
T33     % And
T34     % ADM
T35     % XMR, register test
T36     % XMR, memory test
T37     % MIN
T40     % Set argument
T41     % Set negative argument
T42     % Add argument
T43     % Add negative argument
T44     % Logical clear argument
T45     % Logical exclusive or argument
T46     % Logical and argument
T47     % Logical or argument
T99     % End main arithmetic tests
```

5.2 Address Arithmetic Tests

The address arithmetic is tested in the following special purpose test table:

```
PTAB/ T65     % Instruction address in C4
      T66     % Instruction address in C0
        0     % Transfer to two argument
      T61     % Data reference, Displacement
      T62     % Data reference, X + D
      T63     % Data reference, B + D
      T64     % Data reference, X + B
      T67     % Indirect addressing, indirect address
      T70     % Indirect addressing, D
      T71     % Indirect addressing, X + D
      T72     % Indirect addressing, B + D
      T73     % Indirect addressing, X + B
      T74     % Multilevel indirect addressing, 2 lev.
      T75     % Multilevel indirect addressing, 7 lev.
      T99     % Termination
```

5.3 Jump and Skip Tests

The sequence controlling instructions are tested by means of the following programs:

```
PTAB/  T60      % RTJ, Return address test
       T76      % RTJ, Jump test
          0
       T150     % JPM, Count test
       T151     % JNM, Count test
       T152     % JZM, Count test
       T153     % JFM, Count test
       T154     % JPM, Jump test
       T155     % JNM, Jump test
       T156     % JZM, Jump test
       T157     % JFM, Jump test
       T160     % JRP, Jump test
       T161     % JRN, Jump test
       T162     % JRZ, Jump test
       T163     % JRF, Jump test
       T51      % CRG, Skip test
       T52      % CRL, Skip test
       T53      % CRE, Skip test
       T54      % CRD, Skip test
       T55      % MIN, Skip test
       T130     % Register Skip, GRE
       T131     % Register Skip, Less
       T132     % Register Skip , Equal
       T133     % Register skip, Different
       T140     % Argument skip, GRE
       T141     % Argument skip, Less
       T142     % Argument skip, Eual
       T143     % Argument skip, Different
       T144     % Bit skip zero
       T145     % Bit skip one
       T99      % Termination
```

5.4 Single Shift and Bit Instructions Test

The single length shift and bit instructions are tested by this test table:

```
PTAB/  T0              % One transfer test required
         0
       T100            % Bit set
       T100REP         % Repeat  for all bit numbers
       T101            % Bit clear
       T100REP         % Repeat  for all bit numbers
       T102            % Left rotate
       T100REP         % Repeat  for all shift counts
       T103            % Left arithmetic
       T100REP         % Repeat
       T104            % Left logical
       T100REP         % Repeat
       T105            % Right rotate
       T100REP         % Repeat
       T106            % Right arithmetic
       T100REP         % Repeat
       T107            % Right logical
       T100REP         % Repeat
       T113            % Bit complement
       T100REP         % Repeat
       T99             % Terminate
```

5.5 Double Precision Shift Tests

These programs require the NORD-50 interpreter to compute the correct results. The test table also uses two special purpose programs FLPINN and FLPUT to modify the test system in order to handle 64 bit numbers. If this program has been run without completion, reset the system by means of this program table:

```
PTAB/  T0
         0
       FLPUT
       T99
```

The double length shift table is:

```
PTAB/  T0                %
         0               %
       FLPINN            %  Change mode
       T120              %  Double left rotate
       T120REP           %  Repeat  for all shift counts
       T121              %  Double left arithmetic
       T120REP           %  Repeat
       T122              %  Double left logical
       T120REP           %  Repeat
       T123              %  Double right rotate
       T120REP           %  Repeat
       T124              %  Double right arithmetic
       T120REP           %  Repeat
       T125              %  Double right logical
       T120REP           %  Repeat
       FLPUT             %  Change to normal mode
       BEGIN             %  STOPS
```

5.6 Floating Point Tests

5.6a This program table tests all the double (64 bits) floating point operations.

```
PTAB/  T0
         0
       T110              %  FIXD (convert to integer) without rounding
       T111              %  FIXD without rounding
       T112              %  FLOD (convert to floating)
       T164              %  Register DIV
       T165              %  Register MUL
       T166              %  DIV
       T167              %  MUL
       T170              %  Register RAFD
       T171              %  Register RSBD
       T172              %  Register RMFD
       T173              %  Register RDFD
```

T174	% FAD
T175	% FSB
T176	% FMU
T177	% FDV
T99	% Terminate

5.6.b This program table tests all the single (32 bits) floating point operations.

PTAB/TO

0

T215	% FLO
T216	% FIX
T217	% FIX Rounded
T220	% FAD
T221	% FMU
T222	% FMV
T223	% FSB
T224	% RAF (floating reg. add.)
T225	% RMF (floating reg. multiply.)
T226	% RDV (floating reg. divide.)
T227	% RSF (floating reg. sub.)

5.7 Execute and Breakpoint Test

PTAB/ T0

0

T135	% Execute
T136	% Execute immediate
T200	% Any reference, inside
T201	% Any reference, outside
T202	% PC reference, inside
T203	% PC reference, outside
T204	% Data reference, inside
T205	% Data reference, outside
T206	% Store reference, inside
T207	% Store reference, outside
T99	% Terminate

5.8 Memory Test

The NORD-50 memory may be tested with the following test table:

```
PTAB/  PMRT     % Prepare memory test, initialize
       T16EL    % Data path, equal low addresses
       T16UL    % Data path, unequal low addresses
       T16EU    % Data path, equal upper addresses
       T16UU    % Data path, unequal upper addresses
       T17      % Address in address test
         0
       T50      % Two argument memory tests
       T99      % Terminate
```

5.9 Complete Test

A complete thorough check of the NORD-50 CPU uses the following test table:

```
PTAB/  T0       % NORD-10/NORD-50 Communication
       T1       % Cycle Counter Test
       T2       % Data paths
       T3       % Address paths
       T7       % LDR/STR
       T7B      % LDR/STR
       T7BREP   % LDR/STR for all register numbers
       T10      % Register block test
       T11      % Register zero test
       T11B     % Transfer to/from overflow reg
       T11C     % Transfer to/from remainder reg.
       T12      % LDF, most significant
       T13      % LDF, least significant
       T14      % STF, most significant
       T15      % STF, least significant
       T64      % Data reference, X + B
       T65      % Instruction address in C4
       T60      % RTJ, return address test
       T76      % RTJ, jump test
       PMRT     % Prepare memory test, initialize
       T16EL    % Data path, equal low addresses
```

T16UL	%	Data path, unequal low addresses
T16EU	%	Data path, equal upper addresses
T16UU	%	Data path, unequal upper addresses
T17	%	Address in address test
0		
T20	%	Register add
T21	%	Register subtract
T22	%	Register and
T23	%	Register or
T24	%	Register exclusive or
T25	%	Register clear
T26	%	Register and with complement of B
T27	%	Register exclusive or with complement of A
T30	%	Register or, with complements of A and B
T31	%	Add
T32	%	Sub
T33	%	And
T34	%	ADM
T35	%	XMR, memory test
T36	%	XMR, register test
T37	%	MIN
T40	%	Set argument
T41	%	Set negative argument
T42	%	Add argument
T43	%	Add negative argument
T44	%	Logical clear argument
T45	%	Logical exclusive or argument
T46	%	Logical and argument
T47	%	Logical or argument
T61	%	Data reference, displacement
T62	%	Data reference, X + D
T63	%	Data reference, B + D
T64	%	Data reference, X + B
T67	%	Indirect addressing, indirect address
T70	%	Indirect addressing, D
T71	%	Indirect addressing, X + D
T72	%	Indirect addressing, B + D

```
T73       % Indirect addressing, X + B
T74       % Multilevel indirect addressing, 2 lev.
T75       % Multilevel indirect addressing, 7 lev.
T150      % JPM, count test
T151      % JNM, count test
T152      % JZM, count test
T153      % JFM, count test
T154      % JPM, jump test
T155      % JNM, jump test
T156      % JZM, jump test
T157      % JFM, jump test
T160      % JRP, jump test
T161      % JRN, jump test
T162      % JRZ, jump test
T163      % JRF, jump test
T51       % CRG, skip test
T52       % CRL, skip test
T53       % CRE, skip test
T54       % CRD, skip test
T55       % MIN, skip test
T130      % Register skip, GRE
T131      % Register skip, less
T132      % Register skip, equal
T133      % Register skip, different
T140      % Argument skip, GRE
T141      % Argument skip, less
T142      % Argument skip, equal
T143      % Argument skip, different
T144      % Bit skip zero
T145      % Bit skip one
T100      % Bit set
T100REP   % Repeat for all bit numbers
T101      % Bit clear
T100REP   % Repeat for all bit numbers
```

T102	% Left rotate
T100REP	% Repeat for all shift counts
T103	% Left arithmetic
T100REP	% Repeat
T104	% Left logical
T100REP	% Repeat
T105	% Right rotate
T100REP	% Repeat
T106	% Right arithmetic
T100REP	% Repeat
T107	% Right logical
T100REP	% Repeat
T113	% Bit complement
T100REP	% Repeat
T110	% FIX without rounding
T111	% FIX with rounding
T112	% FLO
T164	% Register DIV
T165	% Register MUL
T166	% DIV
T167	% MUL
T170	% Register FAD
T171	% Register FSB
T172	% Register FMU
T173	% Register FDV
T174	% FADD
T175	% FSBD
T176	% FMUD
T177	% FDVD
T215	% FLO (single)
T216	% FIX
T217	% FIX ROUNDED
T220	% FAD
T221	% FMU
T222	% FDV
T223	% FSB
T224	% RAF
T225	% RMF

T226	% RDV
T227	% RSF
T135	% Execute
T136	% Execute immediately
T200	% Any reference, inside
T201	% Any reference, outside
T202	% PC reference, inside
T203	% PC reference, outside
T204	% Data reference, inside
T205	% Data reference, outside
T206	% Store reference, inside
T207	% Store reference, outside
T99	% Terminate

5.10 Fast Complete Test

The completed test in Section 5.9 may be speeded up by reducing the number of input data combinations for two argument type tests with the following modifications to the system:

```
SPEED/AAX 20
```

This modification from AAX 2 until AAX 20 reduces the number of input data combinations from 16384 until 4·64·8 = 2048.

APPENDIX

This appendix gives a short description of each program supplied by the NORD-50 test system. The appendix should be used whenever an error has been detected. It is organized sequentially by T number and gives information about:

a) Scope triggering signal(s)

b) Actions during the test loop

c) Special hints so that the user may isolate the error fast

d) Entries in the DISPT and IRTAB for those who are writing their own test program tables.

The appendix does not supply the information about which registers are used during the test. This information is available through the display keyboard by typing D and looking at the instructions.

For further information the user should consult the program listing, or even contact the designer of the NORD-50 test system, Mr. Rolf Skår, A/S Norsk Data-Elektronikk.

Note: The scope triggering signal is a logic signal which only occurs once during the test loop and therefore gives stable triggering. It is tried to specify this signal so that it is active when the action tested takes place. However, this is not true in every case. The scope triggering signals marked by *, identifying those cases where the triggering signal is not active during the most interested part of the scope picture.

Name and description		Action	Debugging hints	Oscilloscope triggering signal	DISPT entry	IRTAB entry
T0	SA communication	Tests data paths NORD-10/NORD-50 by setting SA register and reading it again.	Memory power on. NORD-50 in READY.	IOSTR1 0 IOSTR2 0	T0D	0
T1	Cycle counter test	NORD-50 Master Clear Tests NORD-50 BUSY/READY, tests mem. req. and write status flip-flops, tests the cycle counter in an RTJ instruction.	Data output has the following meaning: Bit 17: Wrong SKIP Bit 18: Wrong SKIP Bit 19: BUSY/READY Bit 20: BUSY/READY Bit 21: Wrong SKIP Bit 22: Mem. req. missing Bit 23: Write signal wrong Bit 24: BUSY/READY Bit 25: Mem. req. missing Bit 26: Write wrong Bit 27: BUSY/READY Bit 28: Mem. req. missing	DC7 0	T1D	0
T2	SD/TD Data paths communication	NORD-50 Master Clear Sets SD register. Starts NORD-50C7→C4, and reads TD register	Check cycle counter C7→C4, timing of $TDS1_0$. Data enabling signals.	DC7 0 DC4 0	T2D	0

Name and description		Action	Debugging hints	Oscilloscope triggering signal		DISPT entry	IRTAB entry
T3	SA/PC Communication	NORD-50 Master Clear Sets SA register. Starts NORD-50, C7→C4 and reads the program counter.	Check MAS3 0 and address enabling signals	DC7 DC4	0 0	T3D	0
T7	LDR/STR	NORD-50 Master Clear. Starts a program LDR STR and reads TD register.	Check chip select and WP signals and timing. Check IR decoding.	DC7 DC4 WRITE	0* 0* 0	T7D	TP7
T7B	LDR/STR with subroutines	NORD-50 Master Clear LDR subroutine Master Clear STR subroutine	Same as T7	WRITE	0	T7BD	TPSETR
T7BREP	Increment register No. for T7B	No NORD-50 action					
T10	Register block test	All registers are loaded (by subroutine LDR), with increasing contents, then all registers are checked (by subroutine STR).	Least significant part of data output tells which register that first failed. Check chip select. Try T7B with this register.	WRITE	0*	T10D	TPEXAR

Name and description		Action	Debugging hints	Oscilloscope triggering signal	DISPT entry	IRTAB entry
T11	Register zero test	Master Clear Stores register 0 and checks whether it is zero	Check enabling signals $ABE1_1$, $ABE2_1$ $RGE1_0$, $MBE1_0$ $BBE1_1$, $BBE2_1$	WRITE 0* PC2 0	T11D	TPEXAR
T11B	Overflow reg. test.	Increasing reg. is loaded with content. Then transferred to overflow reg (2) and then read back and checked.	ABE1 FRE	WRITE SRS1	T11BD	IRREG.
T11C	Remainder reg. test.	Same as T11B both remainder reg(3).	ABE1 FRE	WRITE SRS2	T11CD	IRREG.
T12	LDF, most significant	Master Clear LDF STR (most sign.)	Check chip select	IR21 1	T12D	TP12
T13	LDF, least significant	Master Clear LDF STR (least sign.)	Check DC3 cycle Chip select	IR21 1 DC3 0	T13D	TP12
T14	STF, most significant	Master Clear LDR (most sign.) STF	Chip select	IR21 1	T14D	TP14+2
T15	STF, least significant	Master Clear LDR (least sign.) STF	DC3 cycle	IR21 1 DC3 0	T15D	TP14+2
T17	Address in address memory test	Memory check program. The address of each location is written into the location, and afterwards this pattern is checked.	Check the address bits where the error first occurs.	WRITE 0	T17D	0

Name and description		Action	Debugging hints	Oscilloscope triggering signal	DISPT entry	IRTAB entry
T20	Inter-register add	Reg. A loaded with operand A by means of the subroutine LDR, then reg. B is loaded with operand B. Operation performed followed by STOP. Results read by STR subroutine.	Check Chip Select and arithmetic control signals.	IR30 1 IR29 1	T20D	TP20
T21	Inter-register subtract	Same as for T20	Same as for T20	IR30 1 IR29 1	T21D	TP20
T22	Inter-register logical and	"	"	"	T22D	"
T23	Inter-register logical or	"	"	"	T23D	"
T24	Inter-register logical exclusive or	"	"	"	T24D	"
T25	Inter-register logical clear	"	"	"	T25D	"
T26	Inter-register logical and with complement B	"	"	"	T26D	"

Name and description		Action	Debugging hints	Oscilloscope triggering signal		DISPT entry	IRTAB entry
T27	Inter-register logical exclusive or with complement A	Same as for T20	Same as for T20	IR0 IR29	1 1	T27D	TP20
T30	Inter-register logical or with complement of A and B	"	"	"		T30D	"
T31	Memory ref, ADD	Register is loaded with operand B (by means of subroutine LDR) Master Clear Operation performed with operand A as memory operand. Results read by STR subroutine.	Arithmetic control signals.	FS4 FS3	0 0	T31D	TP30
T32	Memory ref, SUB	Same as T31	Same as T31	FS0	0	T32D	T30D
T33	Memory ref, AND	"	"	WRITE	0*	T33D	T30P
T34	Memory ref, ADM (Add to memory)	Same as T31 except results are read by reading TD	Check C3 cycle	DC3 WRITE	0 0	T34D	T30P

Name and description		Action	Debugging hints	Oscilloscope triggering signal	DISPT entry	IRTAB entry
T35	XMR, register test	Same as T31	Check C3 cycle	WRITE 0	T35D	T30P
T36	XMR, memory test	Same as T34		WRITE 0	T36D	T30P
T37	MIN,	Same as T34	Check C3 cycle	WRITE 0	T37D	T30P
T40	Set argument	Reg. is loaded with operand A Master Clear Argument instruction performed with the 16 least sign. bits of operand B as argument. Results read by STR subroutine.	Check enabling signals, and arithmetic control signals.	IR31 1	T40D	ARGIN4
T41	Set negative argument	Same as T40	Arithmetic control signal	IR31 1	T41D	ARGIN4
T42	Add argument	"	"	"	T42D	"
T43		"	"	"	T43D	"
T44		"	"	"	T44D	"
T45		"	"	"	T45D	"

Name and description		Action	Debugging hints	Oscilloscope triggering signal	DISPT entry	IRTAB entry
T46		Same as T40	Arithmetic control signal	IR31 1	T46D	ARGIN4
T47		"	"	"	T47D	"
T50	Two argument memory test	Operand A is the input data, operand B is the address.		WRITE 0	T50D	0
T51	CRG, Memory compare greater, skip test	Register is loaded with operand A. Reg. 77_8 is set to +3 Master Clear A program CRG followed by ADD NEGARG 1 ADD NEGARG 1 is started, if skip, the result is 2, if not it is 1. The result is measured by reading reg. 77 (with subroutine STR).		WRITE 0*	T51D	P51
T52	CRL, Memory compare less, skip test	Same as T51		WRITE 0*	T52D	P51
T53	CRE, Memory compare equal, skip test	"		"	T53D	"

Name and description		Action	Debugging hints	Oscilloscope triggering signal		DISPT entry	IRTAB entry
T54	CRD, Memory compare different, skip test	Same as T51		WRITE	0*	T54D	P51
T55	MIN, Skip test	"		MIN	0	T55D	"
T60	RTJ, Return address transfer test	Master Clear SA is set, a RTJ with return address in reg. 4 is performed, result is obtained by subroutine STR	Enabling signals	RTJ	0	T60D	TP60
T61	Address test, displacement alone	First index register is loaded (with subroutine LDR) Then base register is loaded. Master Clear The instruction CRG with the actual address arithmetic specification to be tested, is performed. The result is measured by reading the TA-register.		IR20	1	T61D	T62P
T62	Address test index + displacement	Same as T61, operand A = index operand B = displacem.		IR20	1	T62D	T62P

Name and description		Action	Debugging hints	Oscilloscope triggering signal		DISPT entry	IRTAB entry
T63	Address test, base + displacement	Same as T61 operand A = base operand B = displacement		IR20	1	T63D	T62P
T64	Address test, index + base	Same as T61 operand A = index operand B = base		IR20	1	T64D	T62P
T65	Address transfer test, C4	Master Clear Set SA-register, start once, C7→C4, read TA-register.	Check PC-enabling	DC7 DC4	0 0	T65D	T62P
T66	Address transfer test C0	Master Clear Set SA-register Start twice, C7→C4→C0 read TA-register	Check program counter counting	DC4 DC0	0 0	T66D	T62P
T67	Address of indirect word	First index register is loaded, then base register (by means of subroutine LDR). Master Clear. The appropriate program sequence is started.		IR20	1	T67D	TP67
T70	Indirect address, displacement only	Same as T67	Cycle counter DC1	DC1	0	T70D	TP70

Name and description		Action	Debugging hints	Oscilloscope triggering signal		DISPT entry	IRTAB entry
T71	Indirect address, index - displacement	Same as T67 operand A = displacement operand B = index		DC1	0	T71D	TP70
T72	Indirect address, base + displacement	Same as T67 operand A = displacement operand B = base		DC1	0	T72D	TP70
T73	Indirect address, index + base	Same as T67 operand A = base operand B = index		DC1	0	T73D	TP70
T74	Indirect addressing, two levels, displacement + index in second level	Same as T67 operand A = displacement operand B = index	Check cycle counter	DC1	0*	T74D	TP74
T75	Indirect addressing, 7 levels, displacement + base in last level	Same as T67 operand A = displacement operand B = base		DC1	0*	T75D	TP75
T76	RTJ, jump transfer test	Base register loaded by subroutine LDR. Master Clear RTJ ,B performed, TA is then read.		RTJ	0	T76D	TP76

Name and description		Action	Debugging hints	Oscilloscope triggering signal	DISPT entry	IRTAB entry
T100	Bit set	Source register(s) loaded with subroutine LDR. Master Clear Destination register(s) read by subroutine STR.	Probably error in the C-rack or in the drive/receive circuits in the CPU (B-rack)	EXOP	T100D	TP100
T101	Bit clear	Same as T100		"	T101D	"
T102	Shift left rotate	"		"	T102D	"
T103	Shift left arithmetic	"	Same as T100	"	T103D	"
T104	Shift left logical	"		"	T104D	"
T105	Shift right totate	"		"	T105D	"
T106	Shift right arithmetic	"		"	T106D	"
T107	Shift right logical	"	Same as T100	"	T107D	"
T110	Convert to integer (FIXD)	"		"	T110D	TP170
T111	Convert to integer with rounding	"	Same as T100	"	T111D	TP170

Name and description		Action	Debugging hints	Oscilloscope triggering signal	DISPT entry	IRTAB entry
T112	Convert to floating (FLOD)	Same as T100	Same as T100	EXOP	T112D	TP170
T113	Bit complement	"	Same as T100	"	T113D	TP100
T120	Double shift, left rotate	"	Same as T100	"	T120D	TP120
T121	Double shift, left arithmetic	"	Same as T100	"	T121D	"
T122	Double shift, left logical	"	Same as T100	"	T122D	"
T123	Double shift, right rotate	"	Same as T100	"	T123D	"
T124	Double shift, right arithmetic	"	Same as T100	"	T124D	"
T125	Double shift, right logical	"	Same as T100	"	T125D	"
T130	Inter-register skip, GRE	The two registers loaded with sub-routine LDR, the last part equals T51		IR30, IR29 1	T130D	P51
T131	Inter-register skip, less than	Same as T130, T51		"	T131D	"
T132	Inter-register skip, equal	"		"	T132D	"

Name and description		Action	Debugging hints	Oscilloscope triggering signal	DISPT entry	IRTAB entry
T133	Inter-register skip, unequal	Same as T130, T51		IR30, IR29 1	T133D	P51
T135	Execute	Reg. 1 and 43_8 loaded with subroutine LDR, EXC of the following instruction RAND 77, 1, 43. Result by storing register 77.		EXC 0	T135D	TP135
T136	Execute immediate	Reg. 1 and 43_8 loaded, reg. 3 and 7 loaded (used as base and index). EXC immediate, with effective address = RAND 77, 1, 43 Result by storing register 77.		"	T136D	TP136
T140	Argument skip, greater than	Same as P51		IR31 1	T140D	P51
T141	Argument skip, less than	"		"	T141D	"
T142	Argument skip, equal	"		"	T142D	"
T143	Argument skip, unequal	"		"	T143D	"

Name and description		Action	Debugging hints	Oscilloscope triggering signal	DISPT entry	IRTAB entry
T144	Bit skip, zero	Same as P51	Same as T100	EXOP	T144D	P51
T145	Bit skip, one	"	Same as T100	"	T145D	"
T150	JPM, Content test	SA register is made zero. Reg. 2 is set to operand A (by subroutine LDR) Reg. 22 (modification register) is set to operand B. Reg. 1 (B-register) is set to operand B. Master Clear A program JPM ,B is started. The result is 2 if no jump, or operand B if jump. Result measured by reading TA-register. Two types of tests, either jump tests or content type tests. If content type test, result is read by subroutine STR.		IR21 1	T150D	TP150
T151	JNM, Content test	Same as T150		IR21 1	T151D	TP150

Name and description		Action	Debugging hints	Oscilloscope triggering signal	DISPT entry	IRTAB entry
T152	JZM, Content test	Same as T150		IR21 1	T152D	TP150
T153	JFM, Content test	"		"	T153D	"
T154	JPM, Jump test	"		"	T154D	"
T155	JNM, Jump test	"		"	T155D	"
T156	JZM, Jump test	"		"	T156D	"
T157	JFM, Jump test	"		"	T157D	"
T160	JRP, Jump test	"		"	T160D	"
T161	JRN, Jump test	"		"	T161D	"
T162	JRZ, Jump test	"		"	T162D	"
T163	JRF, Jump test	"		EXOP	T163D	"
T164	Register Division (integer)	Same as T100		"	T164D	TP170

Name and description		Action	Debugging hints	Oscilloscope triggering signal	DISPT entry	IRTAB entry
T165	Register multiply (integer)	Same as T100	Probably error in the A-rack or in the drive/ receive circuits in CPU (B-rack)	EXIN	T165D	TP170
T166	DIV (integer)	"	Same as T165		T166D	"
T167	MUL (integer)	"	Same as T165	EXIN	T167D	"
T170	RAFD (register add floating double)	"	Same as T165	EXOP	T170D	"
T171	RSFD (register sub floating)	"	Same as T165	"	T171D	"
T172	RMFD (register multiply floating)	"	Same as T165	"	T172D	"
T173	RDFD (register divide floating)	"	Same as T165	"	T173D	"
T174	FADD	"	Same as T165	EXIN	T174D	"
T175	FSBD	"	Same as T165	"	T175D	"
T176	FMUD	"	Same as T165	"	T176D	"
T177	FDVD	"	Same as T165	"	T177D	"

Name and description		Action	Debugging hints	Oscilloscope triggering signal	DISPT entry	IRTAB entry
T200	Breakpoint Any reference inside	BP and BQ are set, as 28 bits positive integers. BP is set to the lowest of operand A and B (bits 0 - 27), and BQ the highest. They are tested against an address equal to exclusive or of BP and BQ. (Displayed in bits 1 - 27 of result). If a stop condition occurs, status bit 1 is set. This is then displayed in bit 0 of result meaning: Bit 0=0 No stop Bit 0=1 Stop Instruction is RAD 1, 2, 2. Before the test SA is set to address tested against.		IR301	T200D	TP202
T201	Breakpoint test, Any reference outside	Same as T200		IR301	T201D	TP202
T202	Breakpoint test, PC reference inside	"		"	T202D	"

Name and description		Action	Debugging hints	Oscilloscope triggering signal	DISPT entry	IRTAB entry
T203	Breakpoint test, PC reference outside	Same as T200		IR301	T203D	TP202
T204	Breakpoint test. Data reference inside	Same as T200, except SA is made zero and the instruction LDR ,B is performed after (B) has been loaded with test address.		M6 1	T204D	TP204
T205	Breakpoint test. Data reference outside	Same as T204		M6 1	T205D	"
T206	Breakpoint test, Store reference inside	Same as T204, except a STR ,B is performed		M7 1	T206D	TP206
T207	Breakpoint test, Store reference outside	Same as T206		M7 1	T207D	"
T16EL	Memory transfer test, equal low address	A memory deposit is performed then a memory examine and results compared Address tested 32		WRITE 0	T16ELD	

Name and description		Action	Debugging hints	Oscilloscope triggering signal	DISPT entry	IRTAB entry
T16UL	Memory transfer test, unequal low address.	Same as T16EL Address tested 33		WRITE 0	T16ULD	0
T16EU	Memory transfer test equal upper address.	Same as T16EL Address tested 100032		WRITE 0	T16EUD	0
T16UU	Memory transfer test unequal upper address.	Same as T16EL Address tested 100033		WRITE 0	T16UUD	0
T215	FLO (single)	Same as T112 (32bits)			T215D	
T216	FIX	T110			T216D	
T217	FIX ROUNDED	T111			T217D	
T220	FAD	T174			T220D	
T221	FMU	T176			T221D	
T222	FDV	T177			T222D	
T223	FSB	T175			T223D	
T224	RAF Reg. floating add	T170			T224D	
T225	RMF	T172			T225D	
T226	RDV	T173			T226D	
T227	RSV	T171			T227D	

A/S NORSK DATA-ELEKTRONIKK
Lørenveien 57, Oslo 5 - Tlf. 21 73 71

COMMENT AND EVALUATION SHEET

The NORD-50 Test System - Reference Manual
September 1975

In order for this manual to develop to the point where it best suits your needs, we must have your comments, corrections, suggestions for additions, etc. Please write down your comments on this pre-addressed form and post it. Please be specific wherever possible.

FROM: _____

_____

_____

– we make bits for the future

NORSK DATA A.S LØRENVEIEN 57 OSLO 5 NORWAY PHONE: 21 73 71 TELEX: 18284