

---

Title:

RC8200  
Disc Testpackage

---

 **REGNECENTRALEN**

RC SYSTEM LIBRARY: FALKONERALLE 1 DK-2000 COPENHAGEN F

---

RCSL No: 30-M241

Edition: February 1979

Author: Finn Grønning

---

Keywords:

RC8000, test program, RC8200 Disc.

---

Abstract:

Description of four test programs for the RC8200 line discs. The first two programs enable initialization of disc kit and reliability testing. The last two are intended for use during initial factory testing and repair. They are used under OPCOM/testadm system.

CONTENTS:		NO OF PAGES
1.	<u>Testprograms, Parameter Values</u> .....	3
2.	<u>Configuration, Requirements, Aecessories, Documents</u> ...	3
	Table with RC 8000 Disctypes .....	
	Selection of Disctype .....	
3.	<u>Address Mark Test (test a)</u> .....	8
	Purpose .....	
	Test Strategy, parameters .....	
	Caution .....	
	Errorhandling .....	
	Channelprograms .....	
	Tetsdata .....	
	Errormessages .....	
	Operator messages .....	
	Test messages .....	
4.	<u>Write and Read (reliability test), test b</u> .....	10
	Purpose .....	
	Testing Strategy, parameters .....	
	Testdata .....	
	Addressingmodes .....	
	Channelprogram .....	
	Errorhandling .....	
	Errormessages .....	
	Remarks .....	
5.	<u>Procedure for Initializing and Checking of Disc Kits</u> ....	6
6.	<u>Examples</u> .....	6
7.	<u>Test Loop Builder (test c)</u> .....	9
	7.1 Purpose .....	
	7.2 Channelprogram, bufferarea .....	
	7.3 Insertion, timer .....	
	7.4 Testcycle, switch control .....	
	7.5 Errormessages .....	

CONTENTS:	NO OF PAGES
8. <u>Move Heads (test d)</u> .....	2
8.1 Purpose .....	
8.2 Initialization .....	
8.3 Seek .....	
8.4 Errormessages .....	

1. Testprogram. parameter values

## Address mark test (test a)

param no.	text	default	min	max
000	select testprogram:	b	a	d
001	number of runs =	19	1	2xx23-1
002	deviceno =	4	4	64
003	rc82no, no =	23	21	54
004	write ?	yes	-	-
005	read ?	yes	-	-
007	statuscheck ?	yes	-	-
008	datacheck ?	yes	-	-
011	first cylinder =	0	0	822
012	first head =	0	0	18
014	number of tracks =	-1	-1	15637
016	first bufferword =	19990	19990	memtop
017	buffersize (tracks) =	10	1	74
019	odd cyl. shift =	0	0	20
023	max messages per block =	2	1	1000
024	cleantrack on ?	yes	-	-
049	terminate on max messages =	10	-1	1000

## Write and read data (test b)

param no	text	default	min	max
000	select testprogram:	b	a	d
001	number of runs =	19	1	2xx23-1
002	deviceno =	4	4	64
003	rc82no, no =	23	21	54
004	write ?	yes	-	-
005	read ?	yes	-	-
007	statuscheck ?	yes	-	-
008	datacheck ?	yes	-	-
011	first cylinder =	0	0	822
012	first head =	0	0	18
013	first sector =	0	0	20
015	number of segments =	-1	-1	328377
016	first bufferword =	19990	19990	memtop
018	buffer size (segments) =	1024	1	1024
019	odd cyl. shift =	0	0	20
021	data kind =	4	0	4
022	addressing mode =	c	a, c or r	
023	max messages per block =	2	1	1000
025	no of retries =	2	0	5
049	terminate on max messages =	10	-1	1000

## test loop builder (test c)

param no.	text	default	min	max
000	select testprogram:	b	a	d
001	number of runs =	10000	1	2xx23-1
002	deviceno =	4	4	64
003	rc82no, no =	23	21	54
026	1. command =	sk	see description	
027	1. first address =	19990	19990	32768
028	1. bytecount =	768	0	24576
029	2. command =	sk	see description	
030	2. first address =	16384	19990	32768
031	2. bytecount =	12	0	24576
032	start in command no:	1	1	3
033	insert from cell:	19990	19990	32768
034	insert no. of words:	1	0	8192
035	insert octal contents:	00200000	8-digit octal	
036	timer (mS) =	200	1	2xx23-1

## move heads (test d)

param no.	text	default	min	max
000	select testprogram:	b	a	d
001	number of runs =	10000	1	2xx23-1
002	deviceno =	4	4	64
003	rc82no, no =	23	21	54
037	1. cylinder =	0	0	822
038	seek length =	100	-822	822





2. Configurations requirements, accessories, documents

The configuration needed is the minimum standard configuration + disc channel consisting of DSC801/CU33 or DSC803/DSA802 and one or more drives.

Accessories needed:

See list, page 2.2

Applicable documents:

DSC 801, Disc Storage Channel, Reference Manual, RCSL 30-M43

DSC803/DSA802, Reference Manual, RCSL 30-M110

RC 8000, Testprogram system, Users Manual, RCSL 30-M216

RC 8200, Disc Testprogram (Slang), RCSL 30-M237.

## DISC TYPES USED BY RC 8000

Type	Sectors	Heads	Cylinders	Tracks	Segments	Words (24bit)	Bytes (MB)		Density
8221	21	5	147	735	15435	3951360	12	Moveable	
22	-	5	219	1095	22995	5886720	18	-	
23	-	5	411	2055	43155	11047680	33	-	
24	-	5	823	4115	86415	22122240	66	-	High
25	-	19	411	7809	163989	41981184	124	-	
26	-	19	823	15637	328377	84064512	248	-	High
8230	-	2	320	640	13440	3440640	10	Fixed	
31	-	4	320	1280	26880	6881280	20	-	
32	-	2	344	688	14448	3698688	11	-	
33	-	4	332	1328	27888	7139328	21	-	
8244	-	5	411	2055	43155	11047680	33	Moveable	
45	-	5	823	4115	86415	22122240	66	-	High
46	-	19	411	7809	163989	41981184	124	-	
47	-	19	823	15637	328377	84064512	248	-	High
8251	-	5	411	2055	43155	11047680	33	-	
52	-	5	823	4115	86415	22122240	66	-	High
53	-	19	411	7809	163989	41981184	124	-	
54	-	19	823	15637	328377	84064512	248	-	High

Selection of disctype.

By means of the parameter:

003 RC 82NO, NO =

a specific disctype may be selected.

NO is the last two digits in the RC Sales Number for the disc. If NO = 23, the RC 8223 disc is chosen.

If a nonexisting disctype is specified (see page 2.2 for a list of current disctypes) the operator message:

xxx illegal discspecification

is printed.

The parameters:

014 number of tracks =

and

015 number of segments =

are both initialized to -1.

With this initialization, the parameters will be given their maximum value corresponding to the disctype parameter (P003), see also page 2.2.

For disc RC 8223 "number of tracks" will be set to 2055 and "number of segments" will be set to 43155.

Before the test is started, the number of heads, and maximum values of number of tracks and segments, is printed.



### 3. Address mark test (test a)

#### Purpose

This test is used for disc systems, where hardware failure causes malfunction of the built-in address mark mechanisms.

It is also used for initializing and testing of virgin disc kits, before these are put into operation.

#### Test strategy, parameters

The test reserves a memory area, which is twice the buffersize specified. The test area start address is defined by the "first bufferword" parameter. If the testarea is larger than the free area, it will be cut, and a check message is output on operators console.

For every run the test writes the number of tracks desired in double buffered mode, and with a blocksize corresponding to the "buffersize". When all the tracks specified have been written they will be check read.

This standard mode of operation may be modified :

If the "clean track" param is "yes", every write operation will be preceded by clean-track operations on the tracks to be written in that operation. Reading or writing may be suspended by means of the parameters read ?, and write ?, respectively.

Every address mark written consists of 4 words (2 for address field, 2 for next field). Thus, the marks in 1 track will occupy a bufferarea of  $21 \times 4 = 84$  words. The smallest buffersize is one track, because address marks can only be written one track at a time.

Normaly, the "clean track" operations should be included when initializing kits, as there otherwise will be a risk that the track contains "forbidden data" in blockgaps making normal operation impossible.

However, it can be useful to disable the cleantrack, e.g. if the address-marks need a refreshment in an attempt to rescue data from a partly spoiled kit.

## CAUTION

If the address test is broken during write, and clean-track is used, there is a risk that the program will be stopped between a clean track and a write-address operation, leaving one or more tracks totally erased.

The odd cyl. shift param defines the physical startsegment (first logical segment) for the tracks on odd cylinders. On even cylinders this will always be segment 0, but to speed transfer of blocks across cylinder shifts, the odd cylinders may be "screwed". In this case "odd cyl. shift" should be set to 10. The shift must always correspond to the driver software operating the disc.

### Errorhandling

Every operation on the disc should be terminated by an interrupt. As interrupt device specified in the devicedescription is not the deviceaddress of the CPU, but a memory cell, in which the interrupt number is stored upon interrupt.

If no interrupt is received within 2 seconds after the operation was started, an errormessage is output, and, if the operation was a seek operation, the controller is reset, and a recalibrate operation is started, and the seek operation is repeated. Statuserror after an initialize operation will terminate the current run.

Whenever a statuserror is detected, an errormessage containing information of the bad statuswords is output on the selected output device. This output may be skipped by setting the "statuscheck" param to "no".

If the data read back are wrong, an errormessage is output, and the wrong words are printed on selected output device.

To avoid entire blocks printed in this way, only a number of bad words in the block will be printed, as defined by the "max messages pr block" param. The checking of data may be skipped entirely by means of the "datacheck" param.

The disc channel is operated by the "do" instruction with an argument address of either start or reset type. If a bus exception is detected upon this "do" instruction, an errormessage will be output, and the instruction repeated. This will go on until the exceptions disappear, or the test is break'ed.

### Channelprograms

The channelprograms used are the simplest possible : an operation followed by a stop command. Status is transferred to statusarea upon interrupt.

Five different types of channelprogram are used :

1. initialize, makes the drive recalibrate. Used in the beginning of each run, and after seek error.

2. seek, is always used before a read or write.
3. write address.
4. read address.
5. clean track.

### Testdata

The testdata used will be the correct address marks. Notice, that when checkreading is performed, the "odd cyl. shift" param must be set correctly. If not, some segments will be skipped, because only the first addressmark will be read from a physical sector number computed by the testprogram, whereas the sequence of the following marks, (in that block), will be defined from mark to mark by the contents of the nextfield. Before read-operations the input buffer is filled with words of all zeroes alternating with words of all ones.

### Error messages

Although OPCOM has made an initial check on the parameters, a detailed checking is performed at test start. Parameter errors may cause these error-messages to appear on the operation console :

bufferstart changed to <new bufferstart >

The bufferstart address was too little. The testprogram proceeds.

bufferlength cut to 2 x <new bufferlength >

The double buffer area desired was larger than the area between bufferstart and memory top.

The testprogram proceeds.

buffer size too small

The buffer cannot hold even one track.

Testprogram terminates.

neither write nor read selected

both "write?" and "read?" has been answered "no".



Operator messages:

heads = x

max tracks = y

max segments = z

this message is printed at the start of each test, informing the operator of the specifications of the disctype chosen.

xxx illegal discspecification

the parameter 003 RC 82NO, NO = specifies an illegal disctype.

disc size exceeded

the number of tracks to be tested exceeds the size of the disc specified

Test messages:

During run the following error messages may be output on the selected output device:

xxx initialize give up. Current run terminated.

a status error was detected after an initialize (recalibrate) operation.

do-exception bus parity / bus timeout / bus nack

do address: <binary address (0:23)>

bus exception is detected upon a do instruction.

The binary address also gives the type of operation:

bit (21:22) = 00 ~ start, bit (21:22) = 01 ~ reset.

blockstart on cylinder <cyl> head <head> sector <sector>

<command> <statuserror>

received : <status received>

expected : <status expected>

Status error has been detected after an operation to the <cyl> , <head> and <sector> position of the disc.

The <command> is the channelprogram-command active.

The possible commands are :

read-addr, (1<8 + 1)

write-addr, (3<8 + 1)

seek, (2<8 + 0)

clean-track, (3<8 + 3)

initialize, (6<8 + 0)

<statuserror> is the statustype in which the error was found.

There will be one complete message for each statusword.

<statuserror> may be :

prg-counter

rem chcount

curr-status

event-status

timer 2 secs (indicating no interrupt received).

<status received> and <status expected> will appear as integers for the two first kinds of status. The next two kinds will be output as texts or, in special cases in binary form.

Notice, that if <status received> contains all ones this usually means, that no status at all has been received.

The possible status texts are :

current status word :

pow-low	(bit 0 )
local	(bit 1 )
seek-err	(bit 5 )
wr-prot	(bit 8 )
high-den	(bit 9 )
kind 4	(bit 21 )
kind 1	(bit 23 )

event status word :

pow/intv	(bit 0 )
data-err	(bit 1 )
overrun	(bit 3 )
hard-err	(bit 4 )
pos-err	(bit 5 )
bus-com	(bit 20 )
int-err	(bit 21 )
bus-tim	(bit 22 )
bus-par	(bit 23 )

Status bits not mentioned here will be printed by the bitnumber.

block begins at cylinder <cyl> , head <head> , sector <sector>

read addressmarks, data error, field <field>

received : <mark received> <cyl> <head> <sector>

expected : <mark expected> <cyl> <head> <sector>

The addressmark read does not match the one computed (expected).

The <cyl> , <head> , <sector> of the blockstart is the physical sector addressed when the blocktransfer was started.

The <field> will either be pointing to the actual address part of the addressmark, field (0 : 1), or the nextfield, field (2 : 3).

Field (0 : 1) always must correspond to the actual (physical) sector whereas the nextfield will point to the following logical segment, which may be found on another track and/or another cylinder.

Notice, that the reading sequence of addressmarks in the block will always be guided by the nextfields.

If the number of wrong data words in a block exceeds the value given by "max messages pr. block" the rest of the block is skipped and the symbol :

```

x   x
x   x
x   x
x   x
x   x
x   x

```

is printed.

If the leftmost word is "all zeroes" and the rightmost "all ones", this is an indication of no data word received, e.g. because of an addressing error.

---

Time consumption, 1 complete run, no errors, RC8223 :

default values of parameters, except

read? = no :      0 : 42      [minutes]

read? = yes :     1 : 23      [minutes]

Time consumption pr run for RC8224 will be twice these values.

4. Write and read (reliability test), test bPurpose

This test is used for long-time testing of the disc-drive. It executes high-speed data transport in double buffered mode, and various addressing and data modes may be selected.

It may also be used for testing the surfacequality of disc kits.

Testing strategy, parameters

The test reserves a memory area which is twice the buffersize specified.

The startaddress of the testarea is defined by the "first bufferword" parameter. If the testarea exceeds the free memory area, it is cut, and a checkmessage is output on operators console.

In every run the test writes the number of segments specified in double buffered mode, and with a blocksize corresponding to the "buffersize". When all the segments wanted have been written, they will be read back in the same way.

This standard mode of operation may be modified by means of the "write?" or "read?" parameters, which may be set to "no" for skipping write or read operations.

Further, statuserrors may cause repetition of the operations a number of times defined by the "no. of retries" parameter.

Every time a status error is detected, the bad statuswords will be printed on selected output device, and a status errorflag will be set. This errorflag is used to signal that the operation may be repeated. If the "status-check?" param is set to "no" no statuserrormessages will appear, but the errorflagging is unaffected.

Similarly, the data read back are checked, word by word. If the "data-check?" param is answered "no" this checking is skipped, speeding the test somewhat.

The "odd cyl. shift" param must be set to the same value as used when addressmarks were written. The standard values are : 0 and 10.

Testdata.

The data contents used are selected by the parameter "datakind".

Five different types of data may be selected :

- 0     all zeroes :   0000 ---- 00
- 1     all ones   :   1111 ---- 11
- 2     alternating zeroes / ones : 0101 ---- 01
- 3     000000101010111111010101
- 4     wordnumber

Type 0 - 3 fills all words with identical information. Datakind 0 and 1 (constant string of zeroes or ones) will result in the highest bit frequency in the read-write channel. Datakind 2 (alternating ones and zeroes) will give the lowest possible bit frequency. Datakind 3, shifting between max and min frequency will form a worst-case test with respect to magnetic distortion (bit shift) and the compensation circuits in the readdetector.

Datakind 4 is the wordnumber written in the word, starting with 0 on first segment / first cylinder. This means, that practically no words will be equal, making this test good for checking sequencing or addressing errors. Also wrong screwing of addressmarks is detected.

The errormessages are relative easily interpreted, as the contents appear counting. Segment starts will always contain a number, which is an integer multiple of 256, and consecutive segments can be recognized by the contents, increasing 256 from segment to segment. Also, the absolute segmentno may easily be calculated from the contents.

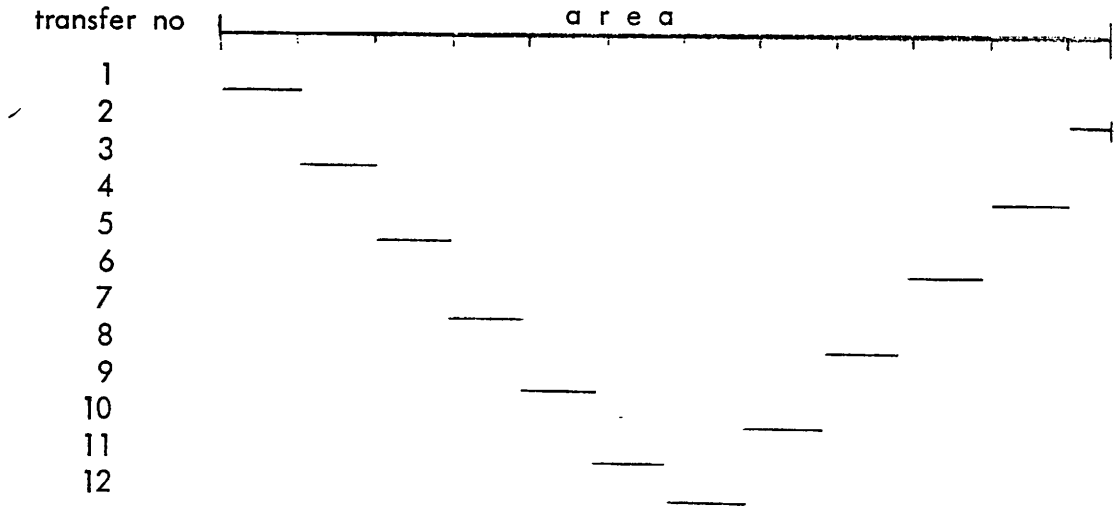
Addressingmodes

Three modes may be selected :

- a     alternating
- c     continous
- r     random

In continous mode the write and read operations are carried out block for block, starting at the beginning of the area specified by the "First cylinder", "First head", "First sector" params. The last block transferred may be cut in size to match the area-bound.

In alternating mode the blocks are transferred from / to the beginning of the area and the end of the area alternately. If the area consists of  $n$  blocks, the sequence is : block 1,  $n$ , 2,  $n-1$ , 3,  $n-2$  etc. This may be visualized so :



Notice, that if the blocksize does not match the areazise, always the highest block will be cut.

In random mode the blocks are selected by means of a random number generator. This means, that during the first run not all segments in the testarea will be written, this will give errorindications during the read operations. To prevent this, the area must have been initialized in continuous or alternating mode.

The random generator generates true random numbers, that is, in two consecutive runs different segments will be affected.

The alternating and random modes perform a hard test of the head-positioning mechanism of the drive, but the run time is much longer than in continuous mode.

### Channelprogram

Four types of channelprograms are used. They consist of two commands of which the second is the stopcommand. The first command will be one of these :

1. initialize, makes the drive recalibrate, used in the beginning of each run and after seek error.
2. seek, is always used before read and write.
3. write data.
4. read data.

If "ECC CHECK? YES" the read command will be followed by a sense command.

### Errorhandling

Every operation on the disc should be terminated by an interrupt. As the interrupt device address in the channelprogramdescription is defined a memory cell, in which the devicenummer is stored upon interrupt.

If no interrupt is received within two seconds after the operation was started, an errormessage is output, and, if the operation was a seek operation, the controller is reset, a recalibrate operation is initiated and the seek operation is repeated. This is done independently on the "retries" param. If a status error is detected after a recalibrate (initialize) operation, the current run is terminated.

Whenever a status error is detected, an errormessage containing the bad statuswords is output on the selected outputdevice.

This output may be skipped, if the parameter "statuscheck" is set to "no".

Statuserror after read data or write data operations may cause repetition of the operation. The number of repetitions allowed is given by the parameter "no. of retries". If the last try executed was unsuccessful, the message

"xxxRetries = 0, give up"

is printed as an give-up indication.

This is printed independently on the "statuscheck?" param.

If the data read back are wrong, an errormessage is output for every bad word. The message contains information about the position and the halfwordnumber relative to the segmentstart, and the datawords received and expected, printed as binary words.

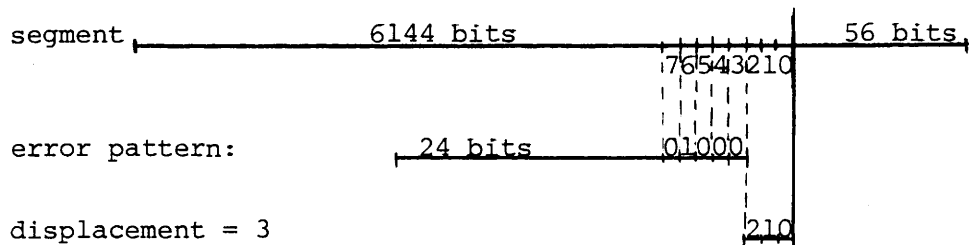


To limit the bulky error messages with totally wrong blocks, only a number of bad words per block—corresponding to the parameter "max messages pr block" will be printed. The time consuming data checking may be skipped by means of the "datacheck?" parameter. The disc channel is operated by means of the "do" instruction with an address specifying either start or reset. If a bus exception is detected upon this "do" instruction, an error messages will be output, and the instruction repeated.

This will go on until the exceptions disappear, or the test is break'ed.

If "ECC CHECK? YES" and the error is correctable, no status message is output. In this case the exact bit-position of the bad spot will be specified. This is no serious error, as monitor release 4 and newer is able to handle error correction. If the error is not correctable, the reason why and the detailed status will be output. At the end of the test the total number of succeeded ECC operations will be output.

The datafield of a segment on the disc surface consists of 768 data characters (= 6144 bits) and 7 check characters (= 56 bits).



In the example above, with a displacement of 3, the bad spot is databit no. 7, counted from the end of the databits.

If the displacement is negative, the error may be in the check character itself. At the end of the test the total number of errors in ECC check character is

output. If the whole errorburst is inside the check-character, it is obvious that there are no errors in the databits, and no correction in the databuffer is needed.

If one of the four factors (N0, N1, N2, N3) in the sense status is negative, or the displacement is more than 6143 bits, the error is not correctable.

#### Errormessages.

When started the parameters are checked. If the parameters are inconsistent, errormessages will appear on operators console as described on page 3.4.

During run the following errormessages may be output on selected outputdevice:

```
"ECC PROG: cylinder = <xx>, Head = <xx>, Sector = <xx>,
          Displacement (bit) = <xx>
          Errorpattern: <.....>"
```

This is the detailed information given at every succeeded ECC operation. There will normally be a number of this message in every normal run.

```
"ECC PROG: check FCO 17-019, 17-020"
```

```
"ECC PROG: Rem byte count error expected"
```

Eventstatus (Dataerror, Headerror) without Rem byte count error.

```
"ECC PROG: ECC SENSE STATUS ERROR"
```

Unexpected standard status on sense command.



Only the blockstart of the discoperation is output.  
A dataerror can be expected too, so look there to find  
the exact segment on the disc.

```
<Command> =      (first command in channelprogram active)
  initialize (6<8 + 0)
  seek       (2<8 + 0)
  write-data (3<8 + 0)
  read-data  (1<8 + 0)
```

```
<statuserror> = (statustype in which the error was found)
  prg-counter
  rem-chcount
  curr-status
  event-status
  timer 2 secs (indicating no interrupt received).
```

<status received> and <status expected> will appear as integers  
for the two first kinds of status. The next two kinds will be  
output as texts, or in special cases in binary forms.

Notice, that if <status received> contains all ones this usually  
means, that no status at all has been received.

The possible status texts are :

current status word :

pow-low	( bit 0 )
local	( bit 1 )
seek-err	( bit 5 )
wr-prot	( bit 8 )
high-den	( bit 9 )
kind 4	( bit 21 )
kind 1	( bit 23 )

event status word :

pow/intv	( bit 0 )
data-err	( bit 1 )
overrun	( bit 3 )
hard-err	( bit 4 )
pos-err	( bit 5 )
bus-com	( bit 20 )
int-err	( bit 21 )
bus-tim	( bit 22 )
bus-par	( bit 23 )

Status bits not mentioned here will be printed by the bitnumber.

Dataerror will cause the following message:

```
"
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
Dataerror at cylinder <xx>, head <xx>, sector <xx>,
Hword <xx>
received <received data in binary format>
expected <expected data in binary format>
xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx"
```

The dataword read back differs from the one computed (expected). The cyl, head and sector is the physical sector addressed. The Hword gives the halfwordnumber in the segment, i.e. words are numbered 0, 2, 4 etc. from segmentstart.

If the number of wrong datawords in a block exceeds the value given by "max messages pr. block", the rest of the block is skipped and the symbol:

```

x  x
x  x
x  x
x  x
x  x
x  x

```

is output.

### Remarks

Notice that the input buffers are initialized to words of all zeroes alternating with words of all ones. This is used to indicate dataerrors in modes of error, where no data is received, or the data are written in a wrong place in the memory.

time consumption, 1 complete run, no errors. RC8223 :

1. buffersize = 110 segments, datakind = 4, addressmode = C

datacheck?	yes	no	
odd cyl shift = 0	2 : 17	1 : 33	[minutes]
odd cyl shift = 10	2 : 16	1 : 30	[minutes]

2. datacheck? yes or no (unimportant) :

buffersize (segments)	1	21	
odd cyl shift = 0	24 : 40	2 : 40	[minutes]
odd cyl shift = 10	24 : 40	2 : 40	[minutes]

The time consumption for RC8224 will be twice these values.

3. addressing mode : alternating

disc drive	RC8223	RC8224	
blocksize = 1 segment	65 : 12	n/a	[minutes]
blocksize = 110 segments	2 : 26	5 : 25	[minutes]

## 5. Procedure for Initializing and Checking of Disc Kits.

5.

Note 1: Monitor release 4 and newer versions are able to handle disc kits formatted with "odd cylinder shift". This is, however, only advantageous for some disc types.

```
RC8221/22/23/24/25/26      odd cyl.shift = 10
RC8244/45/46/47
```

```
RC8230/33                  odd cyl.shift = 0
RC8251/52/53/54
```

To initialize a new disc kit proceed as follows:  
(Both testA and testB must be run).

1. Run the addressmark test with the following non-default parameters:
 

```
000 Select testprogram: a
001 Number of runs = 1
002 Device no = <XX>
003 RC82NO, no = <YY>
019 Odd cyl.shift = <see note1>
```

When "\*\*\*Test terminated" appear on the console without error messages, the addressmarkers have been written.

2. When the addressmarks have been written, the data segments must be initialized before normal operation takes place. Otherwise statuserror may occur when unfilled areaslices are to be moved. Further it is a good practice to read back the segments to check for possible bad spots on the kit surface. This is done by running the reliability test with the following non-default parameters:
 

```
000 Select testprogram: b
001 Number of runs = as desired, min. 1
019 Odd cyl.shift = same value as in test a.
```

Now also the addressmarks have been thoroughly checked. To speed up the test, the maximum blocksize is automatically used. In case of bad spots in the datafield of a segment, the error will be corrected. A message on the console will tell the exact position of every bit, which may be wrong. At the end of the test the total number of succeeded errorcorrections is written.

3. Criteria to reject a disc kit.

- a) TestA: If any error, then start testA again. If 6 starts are made without an errorfree run, the kit cannot be used.
- b) TestB: If any error on cylinder 000, heads 00 or 01, the kit cannot be used.
- c) TestB: If dataerror without statuserror, the errorburst is more than 22 bit (the errorburst may continue into the ECC check character), and it is not even possible for the ECC to discover the error. This is very serious, and the kit cannot be used. (Even if the test succeeds when started again, the kit cannot be used).
- d) TestB: If "ECC PROG: ERROR UNCORRECTABLE" appear, then start testB again, with parameter 025 No of retries = 5. If the error is still uncorrectable, the kit cannot be used.
- e) TestB: If "Number of succeeded ECC operations" is greater than a certain number, the kit will be exchanged by the factory.



<u>Kit size:</u>	<u>Number of correctable errors:</u>
33 Mbytes	15
66 Mbytes	30
124 Mbytes	100
248 Mbytes	100

This kind of error is not serious, because the monitor does not spend much extra time on an ECC operation.

#### C A U T I O N

If the addresswriting is interrupted before the run is terminated, there is a risk that the program is stopped between a cleantrack and a write address operation leaving one or more tracks totally erased.

#### Examples:

On page 5.4 it is shown how to start up the testsystem, the selection of testA and an errorfree run.

On page 5.5 testB is run once with datakind 4. The number of correctable errors is 3, which is less than the acceptable 30 for a 66 Mbyte kit.

SYSTEM: 3000  
 BASIC REV 02.01

>S  
 INT SYSS

>P0000  
 P0000 TEST - OPERATOR COMMUNICATION PROGRAM (VERS. 79.02.00)

AUTOCLEAR ON  
 AUTORESET ON  
 TESTTIME ON

TESTTIME VERSION: 79.02.00  
 MEMORY SIZE = 1000000

OUTPUT DEVICE - TPX

SELECT FUNCTION: NEW DISC  
 RC 0000 DISC TEST FEB 79

SELECT FUNCTION: P0000 LIST  
 RC 0000 DISC TEST FEB 79

000 SELECT TESTPROGRAM: P  
 001 NUMBER OF RUNS = 19  
 002 DEVICE# = 4  
 003 RECORD# MB = 29  
 004 WRITE? YES  
 005 READ? YES  
 007 STATUSCHECK? YES  
 008 DATACHECK? YES  
 011 FIRST CYLINDER = 0  
 012 FIRST HEAD = 0  
 014 NUMBER OF TRACKS = -1  
 016 FIRST BUFFERWORD = 19999  
 017 BUFFER SIZE (TRACKS) = 10  
 019 ODD CYL. SHIFT = 0  
 023 MAX MESSAGES PER BLOCK = 2  
 024 CLEAR TRACK ON? YES  
 049 TERMINATE ON MAX MESSAGES = 10

SELECT FUNCTION: P001/1 P002/5 P003/24 P019/10 START  
 SELECT FUNCTION:

HEADS	=	5
MAX TRACKS	=	4115
MAX SEGMENTS	=	86415

FIRST TRACK = 0  
 RESERVED TRACKS = 4115  
 RUN NO. 1  
 TEST TERMINATED

SELECT FUNCTION:

P000/B LIST  
RC 8000 DISC TEST FEB 79

```

000 SELECT TESTPROGRAM: B
001 NUMBER OF RUNS = 1
002 DEVICENO = 4
003 RC82NO, NO = 24
004 WRITE? YES
005 READ? YES
006 ECC CHECK ? YES
007 STATUSCHECK? YES
008 DATACHECK? YES
011 FIRST CYLINDER = 0
012 FIRST HEAD = 0
013 FIRST SECTOR = 0
015 NUMBER OF SEGMENTS = -1
016 FIRST BUFFERWORD = 19990
018 BUFFERSIZE (SEGMENTS) = 1024
019 ODD CYL. SHIFT = 10
021 DATA KIND = 4
022 ADDRESSING MODE = A/C/R? C
023 MAX MESSAGES PR BLOCK = 2
025 NO OF RETRIES = 0
049 TERMINATE ON MAX MESSAGES = 10

```

SELECT FUNCTION: START  
SELECT FUNCTION:

```

HEADS           =           5
MAX TRACKS     =         4115
MAX SEGMENTS   =         86415

FIRST SEGMENT  =           0
RESERVED SEGMENTS =         86415
BUFFERSIZE OUT TO (SEGMENTS) :         236
RUN NO.        1
RUNNO.         1

```

```

ECC PROG: CYLINDER= 196, HEAD= 2, SECTOR= 2, DISPLACEMENT(BIT)= 64
ERROR PATTERN: .....1

```

```

ECC PROG: CYLINDER= 197, HEAD= 2, SECTOR= 2, DISPLACEMENT(BIT)= 64
ERROR PATTERN: .....1.1

```

```

ECC PROG: CYLINDER= 327, HEAD= 1, SECTOR= 11, DISPLACEMENT(BIT)= 207
ERROR PATTERN: .....11

```

```

NUMBER OF ERRORS IN ECC CHECK CHARACTER = 0
NUMBER OF SUCCEEDED ECC OPERATIONS = 3
***TEST TERMINATED

```

SELECT FUNCTION:

This page is intentionally left blank.

6. Examples:

	page
testA: Normal run	5.4
- : Statuserror during write-addr.	6.2
- : Status- and dataerror during write-addr.	6.3
testB: Normal run	5.5
- : Statuserror during write-data	6.4-6.5
- : Dataerror without statuserror	6.6

P000/A LIST  
RC 8000 DISC TEST FEB 79

000 SELECT TESTPROGRAM: A  
001 NUMBER OF RUNS = 19  
002 DEVICENO = 4  
003 RC82ND, NO = 23  
004 WRITE? YES  
005 READ? YES  
007 STATUSCHECK? YES  
008 DATACHECK? YES  
011 FIRST CYLINDER = 0  
012 FIRST HEAD = 0  
014 NUMBER OF TRACKS = -1  
016 FIRST BUFFERWORD = 19990  
017 BUFFERSIZE (TRACKS) = 10  
019 ODD CYL. SHIFT = 0  
023 MAX MESSAGES PR BLOCK = 2  
024 CLEANTRACK ON? YES  
049 TERMINATE ON MAX MESSAGES = 10

SELECT FUNCTION: P001/1 P002/5 P003/24 P019/10 START  
SELECT FUNCTION:

HEADS = 5  
MAX TRACKS = 4115  
MAX SEGMENTS = 86415

FIRST TRACK = 0  
RESERVED TRACKS = 4115  
RUN NO. 1

RUNNO. 1

-----  
BLOCKSTART ON CYLINDER: 566, HEAD: 0, SECTOR: 0  
WRITE\_ADDR PRG\_COUNTER  
RECEIVED: 10046  
EXPECTED: 10052

WRITE\_ADDR REM\_LCHCOUNT  
RECEIVED: 1516  
EXPECTED: 0

WRITE\_ADDR CURR\_STATUS  
RECEIVED: SEEK\_LERR HIGH\_DEN KIND4 KIND1  
EXPECTED: HIGH\_DEN KIND4 KIND1

WRITE\_ADDR EVENT\_STATUS  
RECEIVED: HARD\_LERR  
EXPECTED:

-----  
BLOCKSTART ON CYLINDER: 568, HEAD: 0, SECTOR: 0  
SEEK PRG\_COUNTER  
RECEIVED: 9920  
EXPECTED: 9926

SEEK CURR\_STATUS  
RECEIVED: SEEK\_LERR HIGH\_DEN KIND4 KIND1  
EXPECTED: HIGH\_DEN KIND4 KIND1

SEEK EVENT\_STATUS  
RECEIVED: HARD\_LERR  
EXPECTED:

-----  
BLOCKSTART ON CYLINDER: 568, HEAD: 0, SECTOR: 0  
SEEK PRG\_COUNTER  
RECEIVED: 9926  
EXPECTED: 9968

\*\*\*TEST TERMINATED

SELECT FUNCTION:

```

000 SELECT TESTPROGRAM: A
001 NUMBER OF RUNS = 1
002 DEVICENO = 5
003 RC82NO, NO = 24
004 WRITE? NO
005 READ? YES
007 STATUSCHECK? YES
008 DATACHECK? YES
011 FIRST CYLINDER = 0
012 FIRST HEAD = 0
014 NUMBER OF TRACKS = -1
016 FIRST BUFFERWORD = 19990
017 BUFFERSIZE (TRACKS) = 10
019 ODD CYL. SHIFT = 10
023 MAX MESSAGES PR BLOCK = 2
024 CLEANTRACK ON? NO
049 TERMINATE ON MAX MESSAGES = 10

```

RUNNO. 1

\*\*\*\*\*

```

BLOCKSTART ON CYLINDER: 0, HEAD: 0, SECTOR: 0
READ_ADDRSS PRG_COUNTER
RECEIVED: 10104
EXPECTED: 10110

```

```

READ_ADDRSS REM_CHCOUNT
RECEIVED: 1516
EXPECTED: 0

```

```

READ_ADDRSS EVENT_STATUS
RECEIVED: HARD_ERR POS_ERR
EXPECTED:

```

```

BLOCK BEGINS AT CYLINDER: 0, HEAD: 0, SECTOR: 0
READ ADDRESSMARKS, DATAERROR, FIELD( 0:1)
RECEIVED: .....11111111 11111111 11111111 0 0 255
EXPECTED: .....1 .....1.1 ..... 1 0 10

```

```

READ ADDRESSMARKS, DATAERROR, FIELD( 2:3)
RECEIVED: .....11111111 11111111 11111111 0 0 255
EXPECTED: .....1 .....1.1 ..... 1 0 11

```

```

* *
* *
* *
* *
* *
* *
* *
* *

```

\*\*\*\*\*

```

BLOCK BEGINS AT CYLINDER: 2, HEAD: 0, SECTOR: 0
READ ADDRESSMARKS, DATAERROR, FIELD( 0:1)
RECEIVED: .....1 .....11 .....1.1 ..... 259 0 10
EXPECTED: .....1 .....11 .....1.1 ..... 3 0 10

```

```

READ ADDRESSMARKS, DATAERROR, FIELD( 2:3)
RECEIVED: .....1 .....11 .....1.1 ..... 259 0 11
EXPECTED: .....1 .....11 .....1.1 ..... 3 0 11

```

```

* *
* *
* *
* *
* *
* *
* *

```

\*\*\*\*\*

```

BLOCK BEGINS AT CYLINDER: 4, HEAD: 0, SECTOR: 0
READ ADDRESSMARKS, DATAERROR, FIELD( 0:1)
RECEIVED: .....1 .....1.1 .....1.1 ..... 261 0 10
EXPECTED: .....1 .....1.1 .....1.1 ..... 5 0 10

```

```

READ ADDRESSMARKS, DATAERROR, FIELD( 2:3)
RECEIVED: .....1 .....1.1 .....1.1 ..... 261 0 11
EXPECTED: .....1 .....1.1 .....1.1 ..... 5 0 11

```

CLOSE  
 OUTPUT DEVICE= TTY  
 SELECT FUNCTION: NEW/DISC  
 RC 8000 DISC TEST FEB 79

SELECT FUNCTION: NEW/DISC LIST  
 RC 8000 DISC TEST FEB 79

RC 8000 DISC TEST FEB 79

000 SELECT TESTPROGRAM: B  
 001 NUMBER OF RUNS = 19  
 002 DEVICEND = 4  
 003 RCSENO, NO. = 23  
 004 WRITE? YES  
 005 READ? YES  
 006 ECC CHECK ? YES  
 007 STATUSCHECK? YES  
 008 DATACHECK? YES  
 011 FIRST CYLINDER = 0  
 012 FIRST HEAD = 0  
 013 FIRST SECTOR = 0  
 015 NUMBER OF SEGMENTS = -1  
 016 FIRST BUFFERWORD = 19990  
 018 BUFFER SIZE (SEGMENTS) = 1024  
 019 ODD CYL. SHIFT = 0  
 021 DATA KIND = 4  
 022 ADDRESSING MODE = A/C/R? C  
 023 MAX MESSAGES PR BLOCK = 2  
 025 NO OF RETRIES = 0  
 049 TERMINATE ON MAX MESSAGES = 10

SELECT FUNCTION: P001/1 P002/5 P003/24 P019/10 CLOSE

OUTPUT DEVICE= LPT  
 SELECT FUNCTION: START  
 SELECT FUNCTION:

HEADS = 5  
 MAX TRACKS = 4115  
 MAX SEGMENTS = 86415

FIRST SEGMENT = 0  
 RESERVED SEGMENTS = 86415  
 BUFFER SIZE CUT TO (SEGMENTS) : 236

RUN NO. 1  
 BREAK  
 \*\*\* TEST TERMINATED

SELECT FUNCTION:



```

000 SELECT TESTPROGRAM: B
001 NUMBER OF RUNS = 1
002 DEVICENO = 5
003 RCB2NO, NO = 24
004 WRITE? YES
005 READ? YES
006 ECC CHECK ? YES
007 STATUSCHECK? YES
008 DATACHECK? YES
011 FIRST CYLINDER = 0
012 FIRST HEAD = 0
013 FIRST SECTOR = 0
015 NUMBER OF SEGMENTS = -1
016 FIRST BUFFERWORD = 19990
018 BUFFERSIZE (SEGMENTS) = 1024
019 ODD CYL. SHIFT = 10
021 DATA KIND = 4
022 ADDRESSING MODE = A/C/R? C
023 MAX MESSAGES PR BLOCK = 2
025 NO OF RETRIES = 0
049 TERMINATE ON MAX MESSAGES = 10

```

RUNNO. 1

\*\*\*\*\*

```

BLOCKSTART ON CYLINDER: 0, HEAD: 0, SECTOR: 0
WRITE_DATA PRG_COUNTER
RECEIVED: 14396
EXPECTED: 14402

```

```

WRITE_DATA REM_CHCOUNT
RECEIVED: 166144
EXPECTED: 0

```

```

WRITE_DATA EVENT_STATUS
RECEIVED: POS_ERR
EXPECTED:

```

\*\*\*RETRIES=0, GIVE UP RETURN= 15134

\*\*\*\*\*

```

BLOCKSTART ON CYLINDER: 2, HEAD: 1, SECTOR: 5
WRITE_DATA PRG_COUNTER
RECEIVED: 14458
EXPECTED: 14464

```

```

WRITE_DATA REM_CHCOUNT
RECEIVED: 186112
EXPECTED: 0

```

```

WRITE_DATA EVENT_STATUS
RECEIVED: POS_ERR
EXPECTED:

```

\*\*\*RETRIES=0, GIVE UP RETURN= 15134

\*\*\*\*\*

```

BLOCKSTART ON CYLINDER: 4, HEAD: 2, SECTOR: 10
WRITE_DATA PRG_COUNTER
RECEIVED: 14396
EXPECTED: 14402

```

```

WRITE_DATA REM_CHCOUNT
RECEIVED: 140544
EXPECTED: 0

```

```

WRITE_DATA EVENT_STATUS
RECEIVED: POS_ERR
EXPECTED:

```

\*\*\*RETRIES=0, GIVE UP RETURN= 15134

\*\*\*\*\*

```

BLOCKSTART ON CYLINDER: 6, HEAD: 3, SECTOR: 15
WRITE_DATA PRG_COUNTER
RECEIVED: 14458
EXPECTED: 14464

```

```

WRITE_DATA REM_CHCOUNT
RECEIVED: 160512

```



## 7. Test loop builder (test c)

### 7.1 Purpose

This test enables the operator to define simple channelprograms to exercise the disc system. No checking of status or data is performed.

### 7.2 Channelprogram, bufferarea

The parameters 026 - 031 defines the two first commands in the channelprogram. The third command always will be a stop command. The command code itself is specified by one of these shorthands :

shorthand	command	code
ss	<u>s</u> ense	0 < 8 + 0
rd	<u>r</u> ead <u>d</u> ata	1 < 8 + 0
ra	<u>r</u> ead <u>a</u> ddress	1 < 8 + 1
sk	<u>s</u> ee <u>k</u>	2 < 8 + 0
sm	<u>s</u> et <u>m</u> ode	2 < 8 + 1
wd	<u>w</u> rite <u>d</u> ata	3 < 8 + 0
wa	<u>w</u> rite <u>a</u> ddress	3 < 8 + 1
ct	<u>c</u> lean <u>t</u> rack	3 < 8 + 3
wt	<u>w</u> ait	4 < 8 + 0
in	<u>i</u> n <u>i</u> t	6 < 8 + 0
st	<u>s</u> t <u>o</u> p	15 < 8 + 0

The "first address" params must point to an address in the buffer area.

The buffer area is located inside the basic 32 kiloword memory module :

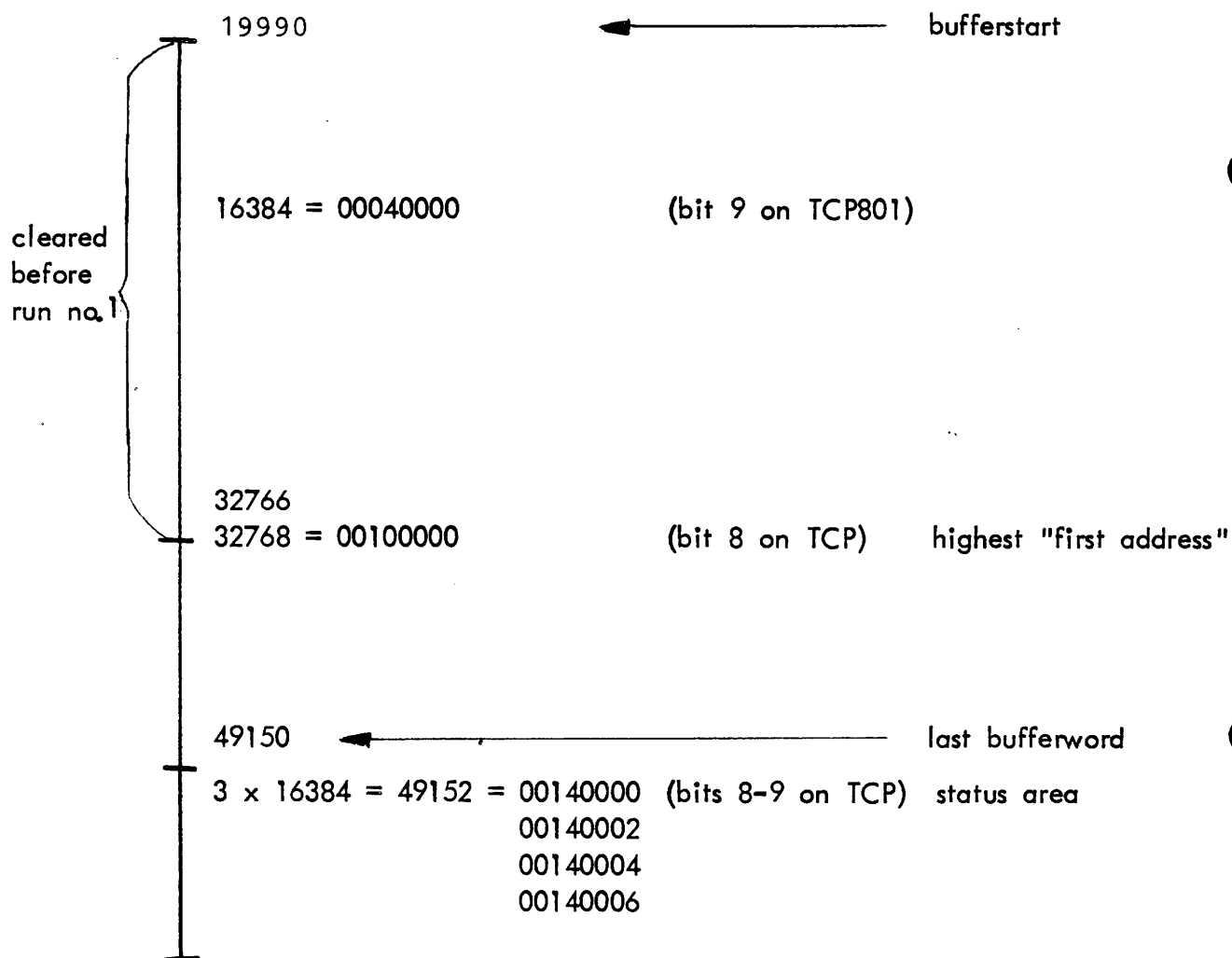


fig. 1: memory buffer

The maximum value of bytcount is 24576 bytes (= 8192 words) to ensure that when the maximum "first address" is used the statusarea will not be overwritten.

example 1 : first address = 32768

last address in the biggest block possible will be :

$$32768 + 8192 \times 2 - 2 = 49150$$

example 2 : reading of the 21 addressmarks of cylinder 0, track 0 to address 16384 and up may be performed by setting these parameters :

```

026 in
027 0      (irrelevant)
028 0      "
029 ra
030 16384
031 252    (= 21 x 4 x 3)

```

Notice, that the channelprogram automatically is supplied with a stop command as the third command.

The channelprogram is executed from the first command on.

It is, however possible to skip one or two commands by changing the parameter

032 start in command no.:

If the value is changed to 3 the devicedescription is pointing on the third command, i.e. only the stop command will be executed, causing status to be transferred, and interrupt received immediately.

During initialization, (before first run) the buffer area from start to 32766 (incl) is cleared.

### 7.3 Insertion, timer

When the test is started an initialization is executed before run no. 1.

During this, the buffer area from 19990 to 23766 is cleared.

After this, information is inserted in cells specified by the parameters

```
033  insert from cell
034  insert no. of words
035  insert octal contents
```

Notice, that the octal contents must be specified by typing 8 octal digits.

By these commands it is possible to fill an area with a specific contents for use during write-data operations.

If the insertion is made in the area 32768 - 49150 a complex pattern forming e.g. address mark information may be set up by repeated starts of the test with various values for no. of words and contents ("insert from cell" must be 32768 in all these cases). Also information which is put in normally by means of the TCP801 should be set in this area to avoid destruction upon start.

On the other hand read operations should preferably be destined to the lower area (19990 - 32766) to eliminate misleading information from earlier testprogram runs.

Notice, that the dabs command provides a convenient possibility for printout of the bufferarea.

When an operation is started, interrupt is expected to signal the termination of that operation. This is checked by means of a software timer based on the digital clock.

When no interrupt has been received before timeout an error message is output and the next run is initiated.

The standard setting of the timer is app. 200 mS, but to facilitate scoping it may be decreased. On the contrary, if the operation sequence takes more than 200 mS the time may be increased.

The timeinterval is set by the parameter

$$036 \quad \text{timer (mS)} =$$

As the timing function counts timer interrupts the precision will be 25.6 mS. Consequently, the time specified will be rounded to the nearest integer multiple of 25.6 mS :

time specified	actual time
0 - 12.7	0 mS
12.8 - 38.3	25.6 "
38.4 - 63.9	51.2 "
64.0 - 89.5	76.8 "
89.6 - 115.1	102.4 "
etc.	

#### 7.4 Testcycle, switch control

In figure 2 is shown the flow during one single run.

Before the first run an initialization is carried out. In this phase the first part of the buffer is cleared, a possible insertion command is executed, a reset and an initialize (recalibrate) is sent to the disc, and the channelprogram is set up.

Then the message

run no. 1

is output, and the test runs, now controlled by the register selector switches on the TCP.

The function of the switches is :

○	○	○	○	○	○
toggle	unused	reset	unused	skip status	toggle mode

If the "togglemode" switch is on (up), each run will be delayed until the "toggle" switch is moved.

For each run the four standard statuswords will be printed. This may be skipped by turning the "skip status" key on.



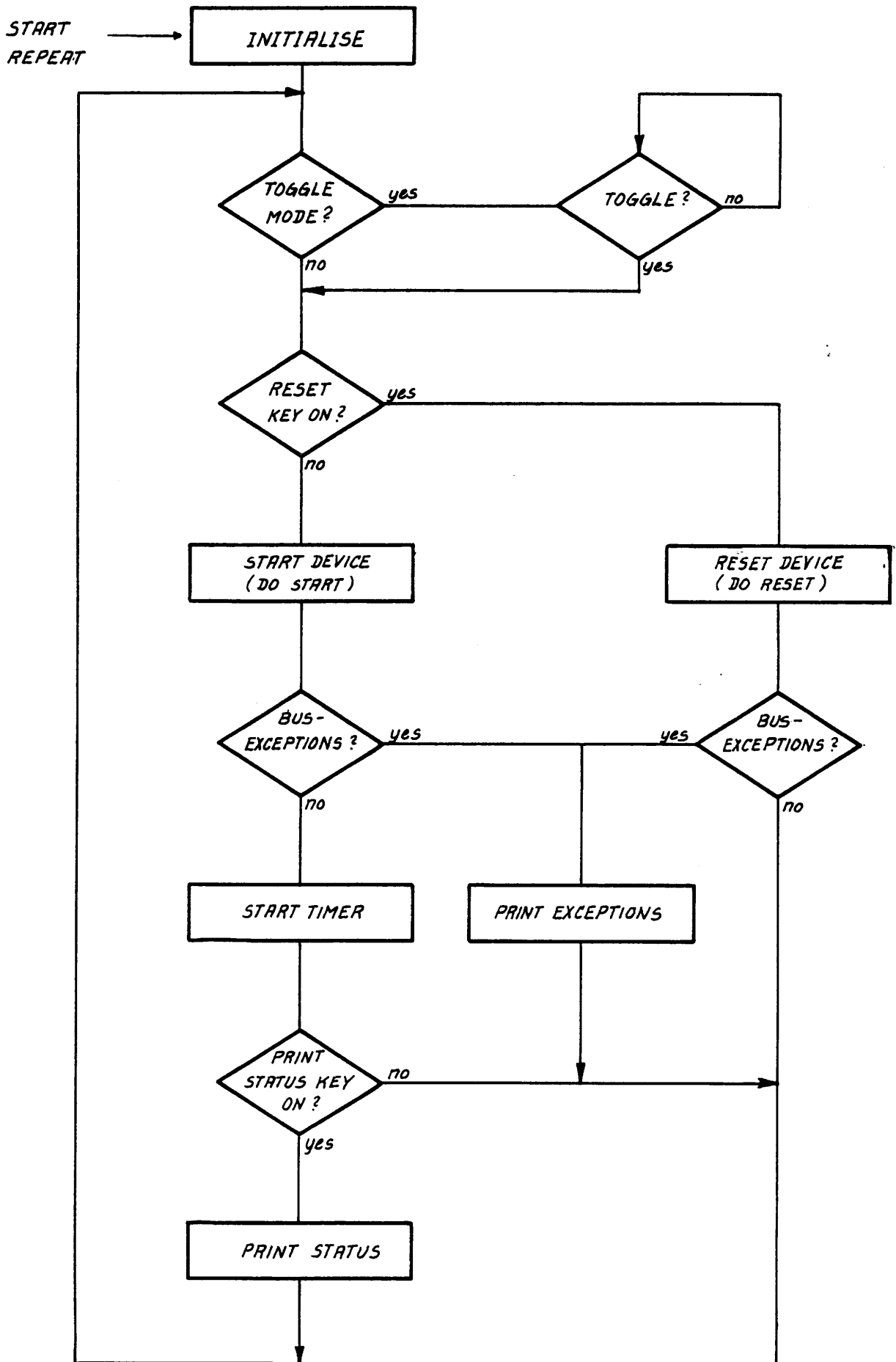


figure 2: flowchart for one run

If the "reset" switch is turned on, the normal channelprogram execution will be suspended, and instead do-reset will be fired.

Normal, continuous operation with statusprinting will be achieved by all switches off or all switches on (corresponding to TCP not present).

For every run the current runnumber will be output on the TCP display.

## 7.5 Errormessages

In the initialization phase these messages may occur on operator's console :

xxx wrong octal

in param 035 a number has been specified which is not an octal or is not consisting of 8 digits (spaces not allowed).

xxx illegal command

in param 026 or 029 a command has been specified, which is not contained in the command list.

During run the following messages may appear on selected output device :

start exceptions <exceptions>

reset exceptions <exceptions>

indicates bus exceptions after a do instruction. May eventually indicate a wrong devicenummer.

xxx timeout

The time specified in param 036 has been exceeded.

After each run the four standard statuswords are output in this layout :

<integer>

<integer>

<text>

<text>

The texts describing current status and event status are described in e.g. RCSL 30-M43, DSC 801 Reference Manual.

If the statusword is empty the text

nil

is output.

Notice, that before any operation is started, the status area is initialized to :

-1  
-1  
0  
0



8. Move heads (test d)8.1 Purpose

This test is intended for exercising the seek mechanism of the disc to enable scoping and adjustments. No statuschecking or datatransfer is performed.

8.2 Initialization

Before the first run the disc is initialized as follows :

The controller is reset and an init (recalibrate) command is executed.

If busexceptions or software timeout is detected during initialization an errormessage is output and the current run terminated.

8.3 Seek

In every run two seeks are performed corresponding to the parameters 037 and 038. The first seek is executed to the cylinder specified as 1. cylinder (037), the second seek is performed to the cylinder with the number first cylinder + seek length (038) .

Notice, that seek length may be negative.

example :

seeks from cylinder 100 to cylinder 90 may be obtained by the parameter settings :

037 1. cylinder = 100

038 seek length = -10

The parameters are checked against the total number of cylinders as defined from the high density bit in the current status word.

No status checking is performed, but in case of busexceptions on the do-start or timeout (no interrupt after 1 second) an errormessage will be output on selected outputdevice.

8.4 Errormessages

Startexceptions <exceptions>

resetexceptions <exceptions>

indicates exceptions after a do instruction.

May eventually indicate a wrong devicenumber.

xxx timeout

The time limit of one second (for the two seek operations totally) is exceeded.

xxx parameter off cylinders

The parameters specify a pair of cylinders, of which one or both are outside the number of cylinders available.



