

NOCUS NEWS

No. 1

September 1981

NORD COMPUTER USERS SOCIETY

CONTENTS

Channel One	2
NOCUS Information	3
SINTRAN III VS Timing Measurements	4
Mailbox	17
News & Rumours	19
Advertisements	21

NOCUS NEWS is published 8-10 times per year by NOCUS.

Subscription is included in NOCUS membership fee.

Editorial correspondence, manuscripts etc. should be addressed to NOCUS NEWS, c/o Norsk Data AS, Box 25 Bogerud, Oslo 6, Norway. Unacceptable manuscripts will be returned if the author's address is submitted.

NOCUS is not responsible for lost manuscripts, figures or photos.

Opinions expressed by the authors are not necessarily those of NOCUS if not specifically indicated.

Copying permitted for other than personal or internal reference only if the source is stated.

NOCUS is a politically, religiously and economically independent organization.

The aims of NOCUS are:

to establish contact between users of computer systems delivered by Norsk Data A/S with subsidiaries (hereafter called ND) and to bring around knowledge and information within the professional area to the members,

to attend to common interests among the members against ND,

to assist the users of computer systems delivered by ND to make the best possible use of them.

This issue is printed in 900 copies and mailed not later than September 21st, 1981.

CHANNEL ONE

We believe that this first issue of NOCUS NEWS marks an advent that has been eagerly awaited. Mostly because NOCUS for a number of years has lacked a suitable communication channel for the members, but also because the board since almost a year has spread rumours about its imminent publication.

It was, when NOCUS proportions were less, with a manageable number of members mainly in Norway, presumably adequate with the opportunities for contacts that our spring and fall meetings provided. But the organization has grown, and it had by the end of this summer 238 members. The quotient NOCUS members/ND customers has not developed accordingly which, on the other hand, indicates that we have difficulties with the communications. Specific member information has been published in the column "Nocus News" in the ND Bulletin. There is, however, a lot that tells against that NOCUS' communication channel is a part of a ND publication. Criticism and debate, or advertising of used equipment and competing software, can hardly find a natural place in a ND publication.

Consequently we think that the World, despite Datamation, Byte and other well-coloured magazines, as well as dozens of IEEE-publications, suffers from the absence of a high quality journal which particularly covers the Science of computing with ND-systems.

This means that we hope that NOCUS NEWS, after a few fumbling issues, stabilises in a state where it spends about half the number of pages on technical articles concerning the use and abuse of ND hardware and software. The other half will be used for departments such as: NOCUS Information, NOCUS Library, Mailbox, Famous Patches, News & Rumours and Advertisements. Titles may change until we have found our style, but the mentioned indicates what we think NOCUS NEWS will cover in addition to technical articles. The "Advertisements" department will be strictly devoted to non-commercial ads until NOCUS Board has formed a well defined policy - if at all possible. The sale of used equipment will until then by definition be regarded as "non-commercial".

NOCUS NEWS is NOCUS' own publication, and, although ND employees may be invited to publish their thesises, the authors and writers should normally be found at NOCUS member institutions.

Until further the Board's Working Committee will act as editorial staff. This means that editorials, if not signed otherwise, expresses opinions represented by NOCUS' Board.

NOCUS INFORMATION

SIBUG (Sibas User Group) Meeting in the ND/France-region

We are planning to arrange a SIBUG Meeting this autumn for the SIBAS-users in this area. During the years several Scandinavian SIBUG Meetings have been arranged successfully, and we have experienced that SIBAS-users outside Scandinavia also are interested.

We hope that the SIBAS-users in the ND/France-region take interest in this meeting, and during the autumn we will come back to you with detailed information.

NOCUS International Meeting in Brighton, 3rd-6th November 1981

There are still places available for this meeting, so if you are interested in attending, please contact Eva Hestnes for registration at telephone No. (02) 29 54 00 during the office hours (not on Wednesdays).

We are for the time being working on the program for the Brighton meeting, and a more detailed version will be sent you as soon as possible.

A confirmation for your participation will also be sent you in the near future.

There will be a ND-500 system available at the hotel in Brighton. The included ND-100 can be run as a stand-alone system. Please feel free to bring your most interesting software for demonstration among the members.

SIBUG Scandinavia

There will be a Scandinavian SIBUG Meeting on November 24th-25th 1981 at Olrud Hotell, Hamar, Norway.

The main topics for this meeting will be data base security and related matters. A detailed invitation will be sent to the members in the Scandinavian countries.

Sintran III VS

Timing Measurements

Richard Herzog, JET/ CODAS Division, Abingdon, OX14 3EA, England

1. Introduction

NORSK DATA has not yet given any information on the timing of the various monitor calls in the SINTRAN III system. Such information, even if not perfectly accurate, is very useful when designing and planning software to be installed on NORD computers. The present note presents measurements conducted on some NORD 100s at JET, using special CAMAC equipment for time measurements. As all system timing figures, these measurements are very dependent of configuration, and load of the system. As much as possible, the main effects on the timing were isolated and repeated whenever feasible. For example, a disc read operation may involve a head movement the first time, but once the head is on the proper track, only disc latency and system overheads are involved. This note presents some information on the monitor calls most commonly used when developing real time systems.

2. Method

A SENSION timer 1009 CAMAC module [1] was used for most of the short times. Such a toggle timer can be counting on 32 bit a 50 MHz internal clock, thus capable of 20 nanosecond resolution, which cannot be achieved as operating the timer via software is slow (ms). On a CAMAC order, or on an external hardware trigger, the count is stopped on one timer, and started on an other counter. This allows to read at leisure the elapsed time since the previous operation. It takes about 1.6 millisecond to perform a reading using the parallel CAMAC interface, but this value must be measured by each program and automatically reflected in the final results.

Analysis of the measurements is prone to errors and discussion ! At first, the raw data was simply printed out, as well as computed mean, minimum, maximum and variance. Quite often, the first or last sample were quite different from the others, and if a reasonable explanation was found (head movement, segment brought to core, etc) it was discarded from the tally, unless a special test was devised to measure only this effect. Other measurements come with a marked two peak distribution, such as the time to access an element in a page: it is short if the page is in core, or it is rather long. It is easy to separate between two disjoint peaks by simply placing threshold limits for lower and upper values. These limits were set for a number of tests, wherever applicable. Such figures will be reported as a number of "bad samples", meaning really rejected samples: they are not used to compute anything. Finally, there are unfortunately many measurements where a broad distribution is found, and cannot be simply resolved into components. These will show a rather large variance, and simply mean that the analysis was not refined further.

RTDSC/GETRT/GRTDA

RTDSC has been consistent from version to version, but GRTDA was on some system timed at 45 milliseconds. This led to further study, showing that GRTDA time depends a lot on the position of the RT program in the system's table.

action	mean	max.	min.	var.	stdev.	samples
RTDSC(rtadrs,array)	6.12	6.33	6.04	0.11	2.%	99
GETRT(0)	0.31	0.52	0.28	0.08	25.%	99
GRTDA("DUMMY" 1st RT program"	2.33	2.62	2.28	0.10	4.%	99
GRTDA("BC031" 20th RT program"	5.23	5.47	5.15	0.12	2.%	99
GRTDA("MDUMMY" 36th RT program"	12.25	12.42	12.08	0.12	1.%	99

HOLDS

This test was simply meant as a variance test to see how accurate is SINTRAN timing. Note that as the process is synchronized to the clock via HOLDS, the function HOLD with a zero value still acts as HOLD with value of 1. This is probably right, but should be noted in the manuals. All the values are too short, as the processing program removed the timing call time from these values, hence one can say that the clock tick is 20 ms within 100 microseconds.

action	mean	max.	min.	var.	stdev.	samples
HOLD 0 clock tick	18.60	18.87	18.30	0.13	1.%	19
HOLD 1 clock tick (20 ms)	18.60	18.87	18.42	0.11	1.%	19
HOLD 2 clock ticks (40 ms)	38.62	38.82	38.38	0.13	0.%	19

5. Segment handling

The RT program STIME was developed to investigate some of the problems related with segment handling. A number of non-demand segments were set up of sizes 1, 2, 4, 8 and 32 pages, as well as two demand segments of size 32 pages, mapping from the upper address range of STIME, accessible via COMMON. A special SWAPPER program was also made, which simply writes into every page of its logical space and starts the program STIME. This is an attempt to force pages to be written back if needed, and to simulate various memory occupation conditions.

As the total memory size was changed, it was difficult to adjust all parameters (one would need a SWAPPER program writing to more than one segment, to write into all the available space). Page faults were only measured with 128 kw memory, and some FIX operations are significant only with 128 kw memory.

Discussion follows:

MEXIT/REENT

MEXIT is fairly constant in its operation, although it was significantly shorter when it had already been called on the same segment. REENT came to be a surprise not yet entirely explained. If the SWAPPER program is run before calling REENT, then REENT takes a large time on the first and second call, while the third call is short. This was repeated on various installations (128 kw only),

and only by actually calling the SWAPPER program could such a behaviour be reproduced. Fixing and unfixing segments did not change REENT timing. On larger memory systems, this behaviour was not found, indicating that some parts of SINTRAN are being swapped in to perform MON REENT. The same behaviour on a faster MEXIT after the third call was found on all systems (80A/E and memory sizes). MON REENT was found to be slow on first call only on the 128 kw systems, provided that the first sample was removed, clearly indicating some swap in of SINTRAN.

action	mean	max.	min.	var.	stdev.	samples
MEXIT 32p ND seg. fixed	1.57	2.56	1.39	0.16	10.%	60
MEXIT 32p DM 1st seg.	1.62	1.98	1.45	0.10	6.%	60
MEXIT 32p DM 2nd seg.	1.22	2.32	1.04	0.22	18.%	60
REENT 32p seg. after SWAPP	67.56	90.10	65.00	7.13	11.%	11 128 kw
REENT 32p seg. 1st repeat	71.16	89.25	64.01	11.05	16.%	11
REENT 32p seg. 2nd repeat	0.98	1.19	0.89	0.10	10.%	11

REENT times	80E	128 kw	memory			196 kw					256 kw
after SWAPP	149	165	140	12	8.%	1.11	1.19	0.96	0.05	5.%	1.08
1st repeat	73	89	64	12	16.%	1.16	1.28	1.13	0.06	5.%	1.30
2nd repeat	1	1	1	.1	9.%	1.31	2.25	1.13	0.31	24.%	1.20

Page faults

In order to measure page faults, read access to each page of the COMMON segment is measured, and all short accesses are discarded. To force swap out, the SWAPPER program is run, while a large segment is fixed in core: this actually forces swapping to occur when the number of pages available for swapping is smaller than the combined sizes of SWAPPER and of the fixed segment. Then, one can measure the time it takes to bring in a page, while swapping one out (remember that SWAPPER writes into each of its pages).

To measure access with no swap out, the whole of core is "cleaned up" by fixing segments, and unfixing them: they have not been written into, hence no swap out is required. Our measurements relate well to the disc rotation: one page per rotation! Such a measurement could not be easily moved to larger memory configurations: one has to fix extra segments to block part of the larger space, so as to force a swap-out when the SWAPPER program is called.

action	mean	max.	min.	var.	stdev.	samples
Page fault/page, with swap out	86.06	155.49	70.89	15.85	18.%	32
Page fault/page, no swap out	30.39	43.19	26.74	4.12	14.%	11

FIX/UNFIX segments already in core

The attempt here was to measure only the time to perform actions on the tables, but not to move the segments from disc. For this, the first sample is discarded from the tallies, as it is on the first fix that the pages are brought from disc. A loop fixing 5 segments, then unfixing them is timed, with fairly consistent results: UNFIX is constant to 0.6 ms, while FIX takes 2.4 ms and about 26 microseconds/page. A previous measurement on another system gave 0.8 ms for UNFIX and 5.1 ms + 20 us for the FIX. Memory size changes by a minor amount these times, it is possible that FIX is marginally longer with larger memories.

action		mean	max.	min.	var.	stdev.	samples
FIX	1 page seg. in core	2.37	2.65	2.24	0.11	5.%	59
UNFIX	1 page seg.	0.52	0.85	0.45	0.10	19.%	60
FIX	2 pages seg. in core	2.44	4.27	2.26	0.26	11.%	59
UNFIX	2 pages seg.	0.50	0.77	0.46	0.07	15.%	60
FIX	4 pages seg. in core	2.48	4.32	2.30	0.27	11.%	59
UNFIX	4 pages seg.	0.54	2.30	0.46	0.25	46.%	60
FIX	8 pages seg. in core	2.55	2.79	2.38	0.10	4.%	59
UNFIX	8 pages seg.	0.54	2.43	0.45	0.26	48.%	60
FIX	32 pages seg. in core	3.16	4.76	3.00	0.24	8.%	59
UNFIX	32 pages seg.	0.51	0.88	0.43	0.09	18.%	60

FIX with swap in

In order to measure the time to bring the segments from disc, one first runs the SWAPPER, fixes the segments, unfixes them, and the process is repeated including running the SWAPPER. It is also necessary to remove all the short samples, indicating that the segments were still in core. The large segments take about 61 ms per page. The sample for 4 pages shown here look too high, previous experiments had a shorter time. As the variance is fairly high, and no grouping could be found, this is left to be explained.

For this measurement, it seems that some internal swap to SINTRAN occurs, as the FIX with swap is shorter when more memory is available, and gives then a more reasonable time related to the disc time alone. The figures for 128 kw and 196 kw should be compared, but it is also feasible that the swap out was not complete, and some pages were still in core !..

action	(80A / 128 kw)	mean	max.	min.	var.	stdev.	samples
FIX	1 page, with swap	74.	112.	26.	36.2	49.%	59
FIX	2 pages, with swap	94.	160.	61.	36.9	39.%	59
FIX	4 pages, with swap	437.	897.	228.	146.0	33.%	59
FIX	8 pages, with swap	495.	1005.	260.	170.3	34.%	59
FIX	32 pages, with swap	1955.	2195.	1551.	161.8	8.%	59

action	(80E)	128kw mean	max.	min.	stdev.	(196kw) mean	max.	min.	stdev.
FIX	1 page, with swap	71.	110.	27.	41.%	36.	50.	26.	20.%
FIX	2 pages, with swap	84.	136.	60.	34.%	61.	63.	60.	1.%
FIX	4 pages, with swap	454.	875.	227.	34.%	128.	129.	126.	0.%
FIX	8 pages, with swap	481.	982.	260.	37.%	260.	261.	258.	0.%
FIX	32 pages, with swap	1919.	2175.	1554.	8.%	1098.	1683.	922.	12.%

FIXC with swapout

The SWAPPER program is run to mark most of core pages, and the series of 5 segments is fixed contiguously. It is very surprising that the small segment of 1 page, fixed first, has a very high penalty: it seems as most of the pages were written back, just to find space for 1 page !? Subsequent fixes took from 40 to 200 ms per page. This behaviour is the same for all systems and memory sizes. This indicates that the algorithm deciding to write back pages operates differently in the case of a FIXC, it seems that the first action of FIXC is to rebuild the "dummy segment" holding all the free pages, when it is found that the required page is not contained in it. Rebuilding this segment is apparently done by writing back all demand segments.

We were also very surprised to find that the segments were brought in one page at a time, instead as a single disc operation, as is perfectly feasible as the space is contiguous on both memory and disc: this is a small improvement which could make FIXC much faster, when the "dummy segment" need not be rebuilt.

action	(80A / 128 kw)	mean	max.	min.	var.	stdev.	samples
FIXC 1 page, with swapout		2553.	2631.	2482.	49.01	2.%	9
FIXC 2 pages, few swapout		293.	312.	287.	10.25	4.%	9
FIXC 4 pages, few swapout		370.	387.	362.	11.73	3.%	9
FIXC 8 pages, few swapout		499.	535.	485.	17.01	3.%	9
FIXC 32 pages, few swapout		1306.	1328.	1303.	7.59	1.%	9

Most file system actions, as well as most XMSG functions have been timed and will be made available later.

References

- [1] Technical Manual for Toggling Scaler/Timer 1009
SESSION
- [2] NORISK DATA A.S. unpublished internal test program to measure
CPU execution speed on the production line, called EXECUTION.
- [3] NORISK DATA A.S. Cartridge disc system for NORD 10 ND-11.008.01.

Appendix 1 : basic monitor calls

MTIME, units in milliseconds,
SINTRAN 80 A

1981/ 6/29 15:28 1981/ 7/14 17:28
/ 128 kw / 125 pages for swapping / 196 kw / 225B pages for swapping

action	mean	max.	min.	var.	stdev.	mean	max.	min.	var.	stdev.
	ms	ms	ms	ms	%	ms	ms	ms	ms	%
dummy loop	1.77	2.69	1.74	0.11	6.%	1.80	2.75	1.77	0.11	6.%
timing call	1.76	1.97	1.73	0.06	4.%	1.80	2.81	1.76	0.12	7.%
OUTCH(intern.dev,byte)	0.87	1.16	0.82	0.07	8.%	0.87	1.74	0.82	0.11	12.%
INCH (intern.dev,byte)	0.24	1.15	0.21	0.11	44.%	0.24	0.57	0.20	0.08	34.%
M8OUT(intern.dev,bytes)	1.02	1.95	0.96	0.12	12.%	1.01	1.31	0.95	0.08	8.%
B8INB(intern.dev,bytes)	0.32	0.54	0.32	0.03	10.%	0.33	0.56	0.31	0.06	17.%
TIME function	0.30	0.60	0.26	0.12	41.%	0.31	0.64	0.27	0.14	43.%
ALTOF mon. call	0.31	0.51	0.28	0.12	40.%	0.32	0.61	0.28	0.13	41.%
ALTON/ALTOF mon.calls	0.63	0.90	0.59	0.13	21.%	0.66	0.98	0.62	0.14	21.%
RT (rtadrs)	1.35	1.51	0.99	0.07	5.%	0.97	1.20	0.96	0.02	2.%
ABORT(rtadrs)	1.49	1.68	1.35	0.07	5.%	1.06	1.08	1.06	0.00	0.%
SET (rtadrs,ticks,units)	0.81	1.91	0.70	0.21	26.%	0.53	1.30	0.43	0.15	29.%
ABSET(rtadrs,secs,mins,hours)	0.96	1.31	0.85	0.11	12.%	0.67	1.64	0.62	0.12	18.%
RTWT RT (repeat bit escape)	2.11	2.43	2.00	0.14	7.%	1.66	1.95	1.61	0.14	8.%
CONCT(rtadrs,logical unit)	0.66	0.95	0.59	0.09	14.%	0.40	0.61	0.36	0.07	17.%
DSCNT(rtadrs,logical unit)	4.71	4.98	4.47	0.14	3.%	6.42	6.67	6.32	0.11	2.%
PRIOR(rtadrs,priority)	1.22	1.61	1.08	0.15	13.%	0.88	1.19	0.83	0.14	16.%
RESRV(unit,1,1) owning it	0.70	1.11	0.62	0.15	21.%	0.44	0.73	0.41	0.13	30.%
RESRV(unit,1,1) not owning it	0.72	1.05	0.62	0.10	13.%	0.43	0.64	0.40	0.07	16.%
RELES(unit,1) owning it	0.73	1.02	0.63	0.10	14.%	0.45	0.78	0.42	0.08	19.%
WHDEV(unit,r/w) free unit	0.65	1.10	0.58	0.15	23.%	0.40	0.73	0.36	0.13	33.%
WHDEV(unit,r/w) resrwd. unit	0.65	1.10	0.59	0.14	22.%	0.40	0.69	0.36	0.13	33.%
RTOFF(rtadrs)	0.65	1.10	0.59	0.14	22.%	0.40	0.69	0.36	0.13	33.%
RTON (rtadrs)	0.59	0.84	0.52	0.13	22.%	0.34	0.67	0.30	0.13	38.%
LATSC(unit)	0.63	1.02	0.56	0.15	23.%	0.38	0.60	0.35	0.13	34.%
RTDSC(rtadrs,array)	7.08	7.33	6.84	0.16	2.%	6.32	6.62	6.23	0.16	2.%
GETRT(0)	0.55	0.84	0.40	0.14	24.%	0.31	0.53	0.28	0.13	41.%
GRIDA("DUMMY" 1st RT program"	2.91	3.30	2.77	0.16	6.%	2.39	2.68	2.34	0.15	6.%
GRIDA("EC031" 20th RT program"	6.17	6.52	5.92	0.18	3.%	5.14	5.41	5.05	0.16	3.%
GRIDA("MDUMMY" 36th RT program"	37.00	37.20	36.74	0.14	0.%	33.60	33.78	33.39	0.16	0.%
HOLD 0 clock tick	18.17	18.45	17.32	0.26	1.%	18.19	18.55	17.95	0.20	1.%
HOLD 1 clock tick (20 ms)	18.23	18.61	17.93	0.21	1.%	18.21	18.55	17.95	0.21	1.%
HOLD 2 clock ticks (40 ms)	38.24	38.53	37.99	0.18	0.%	38.21	38.60	38.05	0.18	0.%

MTIME, units in milliseconds,
SINTRAN 80 E

1981/ 7/16 17:48.07.15 1981/ 7/15 13: 2 .07.15
/ 128 kw / 123B pages for swapping /196 kw / 223B pages for swapping /

action	mean	max.	min.	var.	stdev.	mean	max.	min.	var.	stdev.
	ms	ms	ms	ms	%	ms	ms	ms	ms	%
dummy loop	1.80	2.69	1.76	0.14	8.%	1.80	2.62	1.77	0.10	6.%
timing call	1.78	2.00	1.76	0.06	3.%	1.79	2.74	1.76	0.11	6.%
OUTCH(intern.dev,byte)	0.88	1.77	0.83	0.12	13.%	0.88	1.73	0.82	0.11	13.%
INCH (intern.dev,byte)	0.86	1.70	0.82	0.11	13.%	0.23	1.19	0.21	0.11	47.%
M8OUT(intern.dev,bytes)	0.93	0.94	0.92	0.00	0.%	0.93	0.93	0.91	0.00	0.%
B8INB(intern.dev,bytes)	0.33	0.42	0.32	0.01	4.%	0.32	0.32	0.31	0.00	0.%
TIME function	0.30	0.59	0.27	0.15	50.%	0.31	0.64	0.27	0.12	39.%
ALTOF mon. call	0.31	0.60	0.27	0.15	50.%	0.32	0.65	0.29	0.13	39.%
ALTON/ALTOF mon.calls	0.64	0.96	0.60	0.15	24.%	0.65	0.94	0.61	0.13	20.%
RT (rtadrs)	0.98	0.99	0.97	0.00	0.%	0.96	0.98	0.95	0.01	1.%
ABORT(rtadrs)	1.07	1.09	1.07	0.01	1.%	1.06	1.07	1.05	0.01	1.%
SET (rtadrs,ticks,units)	0.52	1.43	0.41	0.18	34.%	0.54	1.44	0.42	0.18	33.%
ABSET(rtadrs,secs,mins,hours)	0.67	0.99	0.63	0.08	11.%	0.66	0.99	0.62	0.07	10.%
RTWT RT (repeat bit escape)	1.64	1.94	1.60	0.16	10.%	1.66	2.00	1.61	0.14	8.%
CONCT(rtadrs,logical unit)	0.41	0.74	0.37	0.07	17.%	0.40	0.73	0.36	0.07	18.%
DSCNT(rtadrs,logical unit)	6.41	6.66	6.31	0.11	2.%	6.42	6.68	6.33	0.11	2.%
PRIOR(rtadrs,priority)	0.87	1.19	0.82	0.16	18.%	0.88	1.20	0.84	0.13	15.%
RESRV(unit,1,1) owning it	0.43	0.75	0.39	0.16	37.%	0.44	0.77	0.40	0.12	28.%
RESRV(unit,1,1) not owning it	0.44	0.65	0.40	0.07	16.%	0.44	0.79	0.40	0.09	20.%
RELES(unit,1) owning it	0.45	0.75	0.43	0.07	16.%	0.45	0.66	0.42	0.07	16.%
WDEV(unit,r/w) free unit	0.39	0.75	0.35	0.15	40.%	0.40	0.73	0.36	0.13	32.%
WDEV(unit,r/w) resrwd. unit	0.39	0.72	0.35	0.16	40.%	0.40	0.72	0.36	0.13	32.%
RTOFF(rtadrs)	0.39	0.72	0.35	0.16	40.%	0.40	0.72	0.36	0.13	32.%
RTON (rtadrs)	0.33	0.62	0.29	0.15	46.%	0.34	0.63	0.30	0.13	37.%
LATSC(unit)	0.37	0.73	0.34	0.15	41.%	0.38	0.72	0.34	0.12	33.%
RTDSC(rtadrs,array)	6.29	6.55	6.19	0.17	3.%	6.32	6.57	6.22	0.15	2.%
GETRT(0)	0.30	0.58	0.28	0.15	50.%	0.31	0.65	0.27	0.12	40.%
GRIDA("DUMMY" 1st RT program"	2.37	2.68	2.32	0.16	7.%	2.39	2.70	2.33	0.14	6.%
GRIDA("BC031" 26th RT program"	5.12	5.41	5.03	0.18	4.%	5.14	5.41	5.05	0.16	3.%
GRIDA("MDUMMY" 42th RT program"	33.46	33.76	33.26	0.18	1.%	33.58	33.76	33.36	0.16	0.%
HOLD 0 clock tick	18.19	18.41	17.95	0.20	1.%	18.18	18.65	17.62	0.25	1.%
HOLD 1 clock tick (20 ms)	18.20	18.41	17.95	0.21	1.%	18.20	18.72	17.51	0.29	2.%
HOLD 2 clock ticks (40 ms)	38.21	38.34	37.93	0.21	1.%	38.20	38.49	37.83	0.22	1.%

MTIME, units in milliseconds, 81.04.23 1981/ 7/14 17:28
 196 kw memory / 225B pages for swapping / SINTRAN 80 A

1981/ 7/15 13: 2
 / SINTRAN 80 E /

action	mean	max.	min.	var.	stdev.	mean	max.	min.	var.	stdev.
	ms	ms	ms	ms	%	ms	ms	ms	ms	%
dummy loop	1.80	2.75	1.77	0.11	6.%	1.80	2.62	1.77	0.10	6.%
timing call	1.80	2.81	1.76	0.12	7.%	1.79	2.74	1.76	0.11	6.%
OUTCH(intern.dev,byte)	0.87	1.74	0.82	0.11	12.%	0.88	1.73	0.82	0.11	13.%
INCH (intern.dev,byte)	0.24	0.57	0.20	0.08	34.%	0.23	1.19	0.21	0.11	47.%
M8OUT(intern.dev,bytes)	1.01	1.31	0.95	0.08	8.%	0.93	0.93	0.91	0.00	0.%
B8INB(intern.dev,bytes)	0.33	0.56	0.31	0.06	17.%	0.32	0.32	0.31	0.00	0.%
TIME function	0.31	0.64	0.27	0.14	43.%	0.31	0.64	0.27	0.12	39.%
ALTOF mon. call	0.32	0.61	0.28	0.13	41.%	0.32	0.65	0.29	0.13	39.%
ALTON/ALTOF mon.calls	0.66	0.98	0.62	0.14	21.%	0.65	0.94	0.61	0.13	20.%
RT (rtadrs)	0.97	1.20	0.96	0.02	2.%	0.96	0.98	0.95	0.01	1.%
ABORT(rtadrs)	1.06	1.08	1.06	0.00	0.%	1.06	1.07	1.05	0.01	1.%
SET (rtadrs,ticks,units)	0.53	1.30	0.43	0.15	29.%	0.54	1.44	0.42	0.18	33.%
ABSET(rtadrs,secs,mins,hours)	0.67	1.64	0.62	0.12	18.%	0.66	0.99	0.62	0.07	10.%
RTWT RT (repeat bit escape)	1.66	1.95	1.61	0.14	8.%	1.66	2.00	1.61	0.14	8.%
CONCT(rtadrs,logical unit)	0.40	0.61	0.36	0.07	17.%	0.40	0.73	0.36	0.07	18.%
DSCNT(rtadrs,logical unit)	6.42	6.67	6.32	0.11	2.%	6.42	6.68	6.33	0.11	2.%
PRIOR(rtadrs,priority)	0.88	1.19	0.83	0.14	16.%	0.88	1.20	0.84	0.13	15.%
RESRV(unit,1,1) owning it	0.44	0.73	0.41	0.13	30.%	0.44	0.77	0.40	0.12	28.%
RESRV(unit,1,1) not owning it	0.43	0.64	0.40	0.07	16.%	0.44	0.79	0.40	0.09	20.%
RELES(unit,1) owning it	0.45	0.78	0.42	0.08	19.%	0.45	0.66	0.42	0.07	16.%
WHDEV(unit,r/w) free unit	0.40	0.73	0.36	0.13	33.%	0.40	0.73	0.36	0.13	32.%
WHDEV(unit,r/w) resrwd. unit	0.40	0.69	0.36	0.13	33.%	0.40	0.72	0.36	0.13	32.%
RTOFF(rtadrs)	0.40	0.69	0.36	0.13	33.%	0.40	0.72	0.36	0.13	32.%
RTON (rtadrs)	0.34	0.67	0.30	0.13	38.%	0.34	0.63	0.30	0.13	37.%
LATSC(unit)	0.38	0.60	0.35	0.13	34.%	0.38	0.72	0.34	0.12	33.%
RTDSC(rtadrs,array)	6.32	6.62	6.23	0.16	2.%	6.32	6.57	6.22	0.15	2.%
GETRT(0)	0.31	0.53	0.28	0.13	41.%	0.31	0.65	0.27	0.12	40.%
GRIDA("DUMMY" 1st RT program"	2.39	2.68	2.34	0.15	6.%	2.39	2.70	2.33	0.14	6.%
GRIDA("ECO31" 20th RT program"	5.14	5.41	5.05	0.16	3.%	5.14	5.41	5.05	0.16	3.%
GRIDA("MDUMMY" 36th RT program"	33.60	33.78	33.39	0.16	0.%	33.58	33.76	33.36	0.16	0.%
HOLD 0 clock tick	18.19	18.55	17.95	0.20	1.%	18.18	18.65	17.62	0.25	1.%
HOLD 1 clock tick (20 ms)	18.21	18.55	17.95	0.21	1.%	18.20	18.72	17.51	0.29	2.%
HOLD 2 clock ticks (40 ms)	38.21	38.60	38.05	0.18	0.%	38.20	38.49	37.83	0.22	1.%

Appendix 2 : segment handling

STIME, segment handling measurements, 81.04.23 1981/ 7/16 17:46 1981/ 7/16 8:53
 SINTRAN 80 E / 128 kw memory / 123B pages for swapping / AB5 196 kw / 223B pages for swapping

action	mean	max.	min.	var.	stdev.	mean	max.	min.	var.	stdev.
	ms	ms	ms	ms	%	ms	ms	ms	ms	%
dummy loop	2.05	2.89	2.02	0.12	6.%	2.08	2.99	2.03	0.14	7.%
timing call	2.06	2.98	2.02	0.14	7.%	2.09	2.94	2.02	0.14	7.%
MEXIT 32p ND seg. fixed	1.35	1.67	1.29	0.08	6.%	1.88	3.01	1.65	0.26	14.%
MEXIT 32p DM 1st seg.	1.44	3.07	1.36	0.23	16.%	1.89	2.41	1.70	0.16	8.%
MEXIT 32p DM 2nd seg.	1.05	1.99	1.00	0.14	13.%	1.47	2.40	1.33	0.17	11.%
REENT 32p seg. after SWAPP	128.07	139.51	114.34	12.28	10.%	1.43	2.18	1.29	0.24	17.%
REENT 32p seg. 1st repeat	1.23	1.30	1.11	0.05	4.%	1.48	2.44	1.29	0.33	23.%
REENT 32p seg. 2nd repeat	1.34	2.22	1.08	0.31	23.%	1.47	2.47	1.29	0.33	22.%
Page fault/page, with swap out	70.30	138.50	53.27	17.24	25.%	*****	*****	*****	*****	.*
Page fault/page, no swap out	30.34	38.15	26.20	4.60	15.%	*****	*****	*****	*****	.*
FIX 1 page seg. in core	2.54	4.32	2.34	0.26	10.%	2.65	4.52	2.45	0.27	10.%
UNFIX 1 page seg.	0.70	1.11	0.61	0.09	13.%	0.73	2.66	0.60	0.27	37.%
FIX 2 pages seg. in core	2.57	4.28	2.36	0.26	10.%	2.63	2.97	2.46	0.12	5.%
UNFIX 2 pages seg.	0.70	0.98	0.60	0.10	15.%	0.78	2.68	0.60	0.36	47.%
FIX 4 pages seg. in core	2.68	4.56	2.45	0.35	13.%	2.82	4.73	2.52	0.44	15.%
UNFIX 4 pages seg.	0.71	0.92	0.60	0.08	11.%	0.69	1.04	0.60	0.11	15.%
FIX 8 pages seg. in core	2.68	3.03	2.50	0.12	5.%	2.75	3.07	2.60	0.10	3.%
UNFIX 8 pages seg.	0.79	2.52	0.61	0.34	43.%	0.82	2.62	0.60	0.28	34.%
FIX 32 pages seg. in core	3.33	5.07	3.07	0.26	8.%	3.44	5.30	3.19	0.28	8.%
UNFIX 32 pages seg.	0.73	2.55	0.60	0.26	35.%	0.73	2.74	0.59	0.28	39.%
FIX 1 page, with swap	71.15	110.06	26.57	28.86	41.%	36.16	50.13	25.99	7.11	20.%
FIX 2 pages, with swap	83.91	136.00	60.42	28.56	34.%	61.17	62.73	59.95	0.51	1.%
FIX 4 pages, with swap	454.18	875.40	226.70	154.91	34.%	127.86	128.86	125.95	0.49	0.%
FIX 8 pages, with swap	480.49	981.98	260.04	178.57	37.%	260.11	260.93	258.39	0.44	0.%
FIX 32 pages, with swap	1919.02	2175.18	1554.27	145.14	8.%	1098.46	1683.06	922.05	128.98	12.%
FIXC 1 page, with swapout	2438.25	2568.33	2238.88	85.49	4.%	2553.80	2654.27	2420.24	75.82	3.%
FIXC 2 pages, few swapout	362.83	379.63	329.84	16.48	5.%	369.58	377.99	352.96	11.71	3.%
FIXC 4 pages, few swapout	426.52	454.27	404.08	14.14	3.%	421.94	452.43	402.30	15.67	4.%
FIXC 8 pages, few swapout	575.77	579.10	553.40	7.91	1.%	584.52	600.97	576.02	11.59	2.%
FIXC 32 pages, few swapout	1363.81	1375.41	1349.78	12.19	1.%	1377.61	1394.23	1368.96	11.73	1.%

STIME, segment handling measurements, 81.04.23

1981/ 7/16 17:24

1981/ 7/16 16:35

SINTRAN 80 A / 128 kw memory / 125B pages for swapping /

196 kw / 225B pages for swapping /

action	mean	max.	min.	var.	stdev.	mean	max.	min.	var.	stdev.
	ms	ms	ms	ms	%	ms	ms	ms	ms	%
dummy loop	2.24	2.55	2.10	0.08	3.%	2.26	3.31	2.09	0.16	7.%
timing call	2.23	3.34	2.08	0.17	7.%	2.23	2.59	2.08	0.09	4.%
MEXIT 32p ND seg. fixed	1.55	1.87	1.43	0.11	7.%	1.59	1.93	1.41	0.12	8.%
MEXIT 32p IM 1st seg.	1.64	3.31	1.49	0.25	15.%	1.69	3.46	1.50	0.30	18.%
MEXIT 32p IM 2nd seg.	1.21	2.28	1.04	0.17	14.%	1.18	1.48	1.06	0.09	7.%
REENT 32p seg. after SWAPP	148.54	164.50	139.39	11.93	8.%	1.11	1.19	0.96	0.05	5.%
REENT 32p seg. 1st repeat	73.03	88.88	63.74	11.93	16.%	1.16	1.28	1.13	0.06	5.%
REENT 32p seg. 2nd repeat	0.99	1.22	0.87	0.09	9.%	1.31	2.25	1.13	0.31	24.%
Page fault/page, with swap out	78.17	204.90	53.20	24.92	32.%	*****	*****	*****	*****	.%
Page fault/page, no swap out	31.64	51.04	26.85	7.95	25.%	*****	*****	*****	*****	.%
FIX 1 page seg. in core	2.26	2.95	2.12	0.15	7.%	2.37	2.83	2.22	0.13	6.%
UNFIX 1 page seg.	0.53	2.31	0.41	0.26	49.%	0.52	0.79	0.43	0.10	19.%
FIX 2 pages seg. in core	2.27	2.59	2.14	0.11	5.%	2.39	2.69	2.26	0.11	4.%
UNFIX 2 pages seg.	0.54	2.33	0.42	0.26	47.%	0.55	2.49	0.43	0.27	49.%
FIX 4 pages seg. in core	2.33	2.70	2.18	0.12	5.%	2.47	3.29	2.30	0.16	6.%
UNFIX 4 pages seg.	0.53	2.27	0.41	0.24	46.%	0.52	0.84	0.43	0.10	19.%
FIX 8 pages seg. in core	2.43	2.71	2.28	0.13	5.%	2.53	2.89	2.39	0.14	6.%
UNFIX 8 pages seg.	0.54	2.30	0.41	0.25	45.%	0.51	0.78	0.43	0.10	19.%
FIX 32 pages seg. in core	3.06	4.61	2.88	0.24	8.%	3.14	3.45	2.98	0.13	4.%
UNFIX 32 pages seg.	0.50	0.77	0.41	0.09	18.%	0.54	2.42	0.43	0.26	48.%
FIX 1 page, with swap	76.79	119.97	37.88	34.95	46.%	47.66	54.77	37.65	6.02	13.%
FIX 2 pages, with swap	97.33	143.99	69.00	35.70	37.%	69.14	71.12	67.26	0.47	1.%
FIX 4 pages, with swap	444.49	856.55	186.00	169.77	38.%	136.29	137.99	134.58	0.35	0.%
FIX 8 pages, with swap	493.35	881.50	260.52	157.72	32.%	260.40	262.36	258.34	0.51	0.%
FIX 32 pages, with swap	1921.78	2198.58	1651.58	136.21	7.%	847.12	1320.02	625.60	173.37	20.%
FIXC 1 page, with swapout	2555.43	2626.68	2453.65	55.35	2.%	2523.65	2601.30	2475.94	39.85	2.%
FIXC 2 pages, few swapout	298.42	313.23	287.16	12.37	4.%	292.71	311.93	287.13	10.27	4.%
FIXC 4 pages, few swapout	381.15	387.09	361.87	10.31	3.%	375.61	386.59	361.65	12.23	3.%
FIXC 8 pages, few swapout	516.47	536.11	510.67	10.38	2.%	529.89	535.62	510.47	10.35	2.%
FIXC 32 pages, few swapout	1300.04	1305.86	1280.73	10.28	1.%	1337.79	1354.69	1328.85	11.68	1.%

MAILBOX

MASS STORAGE UTILIZATION ON NORD COMPUTERS

The SINTRAN III File System is wasting mass storage space. Each user has a reserved area. This prevents deadlocks, but takes much space in case of several users. The 1024 words index blocks are relatively empty.

Simple techniques for compressing would save perhaps more than 40% of the mass storage areas used today, and also lead to large improvements in disc transfer time.

Is anyone working on these matters? Is the rate of hardware development too fast, causing utilization to fall behind? What about some intelligence built into disc and discette drives?

Helge Totland
A/S Computas

ANSWER FROM ND

These questions has been raised before, and the problem is therefore well known. Both suggestions for improvements is on the list of things that we would like to do with SINTRAN III. The restrictions with fixed area for each user is easiest to do something with, but is still not high enough on the priority list, and we have therefore not any immediate plans to change this.

A general comprimation of all files by means of more intelligent disk drivers, disk interfaces or disk drives has nor been evaluated. Various suggestions for reducing the space for index blocks have been evaluated, but we have so far found that they will lead to too many changes in the file system to be implemented.

We will also use the opportunity to inform that in the latest release of the BRF-editor (the S version of ND-10005 or Q version of ND-10044) is included a command for making library files that can be loaded by NRL. This may be used for saving file space for BRF-files.

Knut Nordbye
Manager, Product Development

--ooOoo--

"There is one quality more important than 'know-how'... This is 'know-what' by which we determine not only how to accomplish our purposes, but what our purposes are to be... Whether we entrust our decisions to machines of metal, or to those machines of flesh and blood which are bureaus and vast laboratories and armies and corporations, we shall never receive the right answer to our questions unless we ask the right questions."

Norbert Wiener

THE NAMING OF SINTRAN VERSIONS

If SINTRAN III/D was the Dangerous version, and SINTRAN III/E the Erroneous version, will then SINTRAN III/F be the Fixed or the Fatal version?

Experienced, 32 years.

PS-PASCAL ERROR CORRECTION NOTICE

Correction No. 14, November 26, 1980.

R. Calliau, M. Krueger, PS-Division, CERN, 1211 Geneva 23, Switzerland

Error corrected: Real number output to mass storage files not correct. This has been reported by many users.

Reason: The D-register gets corrupted by PTCI (but, indeed, only for mass-storage files).

Corrections to be made: In the routine ØWRR-ØWRF of the library (part NPL-INPOUT-LIB).

- 1) In the DISP-section following the line SUBR ØWRR,ØWRF; add INTEGER TEMPD at the very end of the section (before the PSID).
- 2) In the scientific notation part, between the point WRR: and *)Fill, change the last IF in the FOR-loop to:

```
FOR X:=0 TO 4 DO
  A:=0;
  T:=TENX(X);
  *RDIV ST;
  IF A><0 OR X=4 THEN MIN FLAG FI;
  IF T:=FLAG><0 THEN
    X:=TEMPX; X:=D:=TEMPD;
    A+##0; X:=FCBAD; A:=X.PASBF; CALL OPTCI;
    X:=TEMPD:=D; X:=TEMPX;
  FI;
OD;
GO FAR BACK;
```

- 3) Rebuild the library.

NEWS & RUMOURS

IS SOFTWARE SCIENCE?

One of our big brothers, the well known ELECTRONICS, sites in its August 11 issue the Bell Laboratories software guru Jack M. Scanlon: "We did a great disservice to computer science when we called it computer science. That label led software practitioners to spend a wealth of time and effort in search of an underlying science for their craft. And when they got tired of looking for a science, then they looked for a method.

While a great deal has been gained through this process..... we still haven't made anywhere near the kind of progress in software that you can see in other technologies. It's time to marry software to a faster moving technology, one based on science.

Not surprisingly, the technology Scanlon says looks most promising as a wedding partner for software is silicon. The union,, will one day lead to systems by which a programmer sitting at a terminal will produce, not an applications program, but an actual system based on the programmer's instructions.

The customer design process that we associate today with silicon will be almost indistinguishable with what we see now as a software process, he predicts."

Any comments from our readers?

--ooOoo--

"The fact that a scientific advance if useful does not make it unscientific."

Glenn T. Seaborg

WHAT WILL FOLLOW NEXT FROM ND?

We have seen the NORD-1 followed by the 10 and 100, and the 50 followed by the 500. What will come next? Yet another zero? Something faster and more expensive - a ND-1000? Or do we see a new direction of development perhaps indicated by the advent of the PIOC. Furthermore, from the Swedish magazine "Industriell Datateknik" we learn that Norsk Data believes in distributed computing power. They will, together with Swedish interests, develop a "terminal computer". It will have stand-alone capabilities, but can as well be a part of a network. It will contain a ND-100 CPU and 512 kb memory, floppy or a 20 Mb Winchester and a 1024x1024 pixel raster-scan display to look at. Cambridge Ring is a probable outcome in the choice of network. Programming languages will be C, Fortran, Lisp, and Pascal under SINTRAN.

The Swedish companies Upnod AB, Cosyl and Karolinska Institutet will be closely engaged in the development.

PROMISING HALF-YEAR RESULT FROM NORSK DATA

The half-year report from Norsk Data A/S for the first six months of this year shows that total operating revenues added up to NOK 187.1 mill., with profits before taxation and extraordinary items of NOK 5.1 mill. This means that the company has had an increase of 65% and 110% respectively, compared to the first half year of 1980.

Orders received increased from NOK 150 mill. in the first half of 1980 to NOK 198.8 mill. this year, which gives an increase of 33%. Export orders now make up more than 50% of these bookings, compared to 47% for 1980 as a whole.

The results for the first six months reflect the traditional uneven distribution of the company's revenue for the full year, while the costs are more evenly distributed. The second half of the year is expected to produce the large majority of operating revenues and particularly profits. The targets announced in the 1980 Annual Report of 40% sales increase, whilst maintaining profit margins, are still valid.

As we referred to in the Annual Report, Norsk Data expects to recover between 1980 and 1982 NOK 30 mill. of its cost of scheduled research and development programs as a result of two agreements concluded with Mobil Exploration Norway Inc. in 1979. During 1980 there were extraordinary credits of NOK 6.3 mill. under these agreements- and for the first half of 1981 a further NOK 4.5 mill. of extraordinary items has been credited.

This means that the half-year result for Norsk Data before year-end adjustments and taxes, adds up to NOK 9.6 mill.

This is the last issue of NOCUS NEWS you will receive - if you are not a NOCUS member.

Membership in NOCUS can be obtained by organizations, institutions, companies or the equivalent that are users of computer systems delivered by Norsk Data, through application to the Board of NOCUS.

ADVERTISEMENTS

FOR SALE

NORD-10/S No. 396

- 128 kb memory
- Memory Management System
- Floppy disc with controller/formatter
- 2x10 Mb Hawk disc with controller/formatter
- 1 synchronous modem interface
- 8 current-loop ports (2x1122)

Contact Johan Samuelson or Jan Larsen, Jonas Øglænd AS, Box 115, N-4301 Sandnes, Norway. Telephone (04) 605000.

Do you need a 66 Mb disc controller (N-10)?

It was bought in 1978 for NOK 130.000,-. What's your offer?

Please contact Hans Munthe-Kaas, Norwegian Institute for Water Research (NIVA), Box 333 Blindern, Oslo 3, Norway.

Telephone (02) 235280.

After upgrading our discs we want to sell 3 dispacks, 33 Mb. type CDC Model 9876.

Please contact Ove Høgtun, Tromsø Telemetristasjon, Box 387, 9001 Tromsø, Norway. Telephone (083) 84817.

WANTED

Tromsø Telemetristasjon wants to expand the Multiport Memory System on the N-10-S. Accordingly we want to buy:

- Multiport Memory Crate
- Memory Ports
- 8 kb MOS memory modules

Please contact Ove Høgtun, Tromsø Telemetristasjon, Box 387, 9001 Tromsø, Norway. Telephone (083) 84817.

Please contact Norsk Data regarding service agreement for second-hand equipment.

NOCUS, C/O NORSK DATA A.S, P.O. BOX 4 LINDEBERG GÅRD, OSLO 10, NORWAY

PHONE: (02) 29 54 00