CYBER 205 Timing

Specification

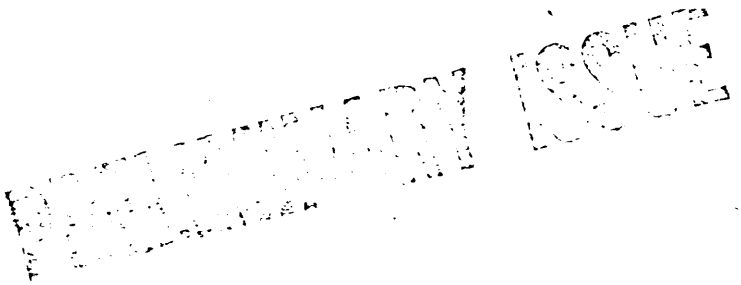


TABLE OF CONTENTS

SUPER COMPUTER OPERATIONS

Index To Timing Tables By Op Code

OP Code	Page #	Type	Description
00	41	MN	Idle
01	41		Illegal
02	41		Illegal
03	31		No op
04	31	NT	Breakpoint - Maintenance
05	31		Void Instruction Stack
06	31	NT	Fault Test - Maintenance
07	41		Illegal
08	41	MN	Input/Output Per R
09	41	BR	Exit Force
0A	41	MN	Transmit (R) to Monitor Interval Timer
0B	41		Illegal
0C	31	MN	Store Associative Registers
0D	31	MN	Load Associative Registers
0E	31	MN	Translate External Interrupt
0F	31	MN	Load Keys, Translate Address
10	31	RG	Convert BCD to Binary, Fixed Length
11	31	RG	Convert Binary to BCD, Fixed Length
12	31	NT	Load Byte
13	31	NT	Store Byte
14	41	NT	Bit Compress
15	41	NT	Bit Merge
16	41	NT	Bit Mask
17	41		Illegal
18	41		Illegal
19	41		Illegal
1A	41		Illegal
1B	41		Illegal
1C	41	NT	Form Repeated Bit Mask, Leading Zeros
1D	41	NT	Form Repeated Bit Mask, Leading Ones
1E	41	NT	Count Leading Equals
1F	41	NT	Count Ones in Field R

SUPER COMPUTER OPERATIONS

Index To Timing Tables By Op Code (Cont.)

OP Code	Page #	Type	Description
20	31	BR	Branch if R EQ S (32-Bit FP)
21	32	BR	Branch if R NE S (32-Bit FP)
22	32	BR	Branch if R GE S (32-Bit FP)
23	32	BR	Branch if R LT S (32-Bit FP)
24	32	BR	Branch if R EQ S (64-Bit FP)
25	32	BR	Branch if R NE S (64-Bit FP)
26	32	BR	Branch if R GE S (64-Bit FP)
27	32	BR	Branch if R LT S (64-Bit FP)
28	41	NT	Scan Equal
29	41		Illegal
2A	32	RG	Enter Length of R with I(16)
2B	32	RG	Add to Length Field
2C	32	RG	Logical Exclusive OR
2D	32	RG	AND
2E	32	RG	Logical Inclusive OR
2F	32	BR	Register Bit Branch and Alter
30	32	RG	Shift R Per S
31	32	BR	Increase R and Branch if R ≠ 0
32	33	BR	Bit Branch and Alter
33	33	BR	Data Flag Register Bit Branch and Alter
34	33	RG	Shift R Per (S)
35	33	BR	Decrease R and Branch if R ≠ 0
36	33	BR	Branch and Set R to Next Instruction
37	41	NT	Transmit Job Interval Timer
38	33	IN	Transmit R (00-15) to T (00-15)
39	41	NT	Transmit Real-Time Clock
3A	41	NT	Transmit R to Job Interval Timer
3B	41	BR	Data Flag Register Load/Store
3C	33	NT	Half Word Index Multiply
3D	33	NT	Index Multiply
3E	34	IN	Enter R with I (16)
3F	34	IN	Increase R by I (16)

Index To Timing Tables By Op Code (Cont.)

OP Code	Page #	Type	Description
40	34	RG	Add U
41	34	RG	Add L
42	34	RG	Add N
43	42		Illegal
44	34	PG	Subtract U
45	34	RG	Subtract L
46	34	RG	Subtract N
47	42		Illegal
48	34	RG	Multiply U
49	37	RG	Multiply L
4A	42		Illegal
4B	34	RG	Multiply S
4C	34	RG	Divide U
4D	34	IN	Half Word Enter R
4E	34	IN	Half Word Increase R
4F	34	RG	Divide S
50	34	RG	Truncate
51	34	RG	Floor
52	34	RG	Ceiling
53	34	RG	Significant Square Root
54	35	RG	Adjust Significance
55	35	RG	Adjust Exponent
56	42	NT	Select Link
57	42		Illegal
58	35	RG	Transmit
59	35	RG	Absolute
5A	35	RG	Exponent
5B	35	RG	Pack
5C	35	RG	Extend
5D	35	RG	Index Extend
5E	35	NT	Load T
5F	35	NT	Store T

Index To Timing Tables By Op Code (Cont.)

OP Code	Page #	Type	Description
60	35	RG	Add U
61	35	RG	Add L
62	35	RG	Add N
63	35	RG	Add Address
64	35	RG	Subtract U
65	35	RG	Subtract L
66	35	RG	Subtract N
67	35	RG	Subtract Address
68	36	RG	Multiply U
69	36	RG	Multiply L
6A	42		Illegal
6B	36	RG	Multiply S
6C	36	RG	Divide U
6D	36	RG	Insert
6E	36	RG	Extract
6F	36	RG	Divide S
70	36	RG	Truncate
71	36	RG	Floor
72	36	RG	Ceiling
73	36	RG	Significant Square Root
74	36	RG	Adjust Significant
75	36	RG	Adjust Exponent
76	36	RG	Contract
77	36	RG	Rounded Contract
78	36	RG	Transmit
79	36	RG	Absolute
7A	36	RG	Exponent
7B	37	RG	Pack
7C	37	RG	Length
7D	42	NT	Swap
7E	37	NT	Load T
7F	37	NT	Store T

Index To Timing Tables By Op Code (Cont.)

OP Code	Page #	Type	Description
80	42	VT	Add U
81	42	VT	Add L
82	42	VT	Add N
83	42	VT	Add A
84	42	VT	Subtract U
85	42	VT	Subtract L
86	42	VT	Subtract N
87	42	VT	Subtract A
88	42	VT	Multiply U
89	42	VT	Multiply L
8A	42	VT	Shift
8B	42	VT	Multiply S
8C	42	VT	Divide U
8D	42		Illegal
8E	42		Illegal
8F	42	VT	Divide S
90	42	VT	Truncate
91	42	VT	Floor
92	42	VT	Ceiling
93	42	VT	Significant Square Root
94	42	VT	Adjust Significance
95	43	VT	Adjust Exponent
96	43	VT	Contract
97	43	VT	Rounded Contract
98	43	VT	Transmit
99	43	VT	Absolute
9A	43	VT	Exponent
9B	43	VT	Pack
9C	43	VT	Extend
9D	43	VT	Logical
9E	43		Illegal
9F	43		Illegal

SUPER COMPUTER OPERATIONS

Index To Timing Tables By Op Code (Cont.)

OP Code	Page #	Type	Description
A0	43	SV	Add U
A1	43	SV	Add L
A2	43	SV	Add N
A3	43		Illegal
A4	43	SV	Subtract U
A5	43	SV	Subtract L
A6	43	SV	Subtract N
A7	43		Illegal
A8	43	SV	Multiply U
A9	43	SV	Multiply L
AA	43		Illegal
AB	43	SV	Multiply S
AC	43	SV	Divide U
AD	43		Illegal
AE	43		Illegal
AF	43	SV	Divide S
B0	37	BR	Index Branch
B1	37	BR	Index Branch
B2	37	BR	Index Branch
B3	38	BR	Index Branch
B4	38	BR	Index Branch
B5	38	BR	Index Branch
B6	38	BR	Branch to Immediate Address
B7	43	VM*	Transmit List Indexed
B8	44	VM	Transmit Reverse
B9	44		Illegal
BA	44	VM*	Transmit Indexed List
BB	44	NT	Mask
BC	44	NT	Compress
BD	44	NT	Merge
BE	38	IN	Enter R
BF	38	IN	Increase R

* = Multi-pass

SUPER COMPUTER OPERATIONS

Index To Timing Tables By Op Code (Cont.)

OP Code	Page #	Type	Description
C0	44	VM	Select EQ A EQ B, Item Count to C
C1	44	VM	Select NE A NE B, Item Count to C
C2	44	VM	Select GE A GE B, Item Count to C
C3	44	VM	Select LT A LT B, Item Count to C
C4	44	NT	Compare EQ A EQ B, Order Vector
C5	44	NT	Compare NE A NE B, Order Vector
C6	44	NT	Compare GE A GE B, Order Vector
C7	44	NT	Compare LT A LT B, Order Vector
C8	44	NT*	Search EQ, Index List
C9	44	NT*	Search NE, Index List
CA	44	NT*	Search GE, Index List
CB	44	NT*	Search LT, Index List
CC	44	NT	Masked Binary Compare
CD	44	IN	Half Word Enter R
CE	44	IN	Half Word Increase P
CF	44	NT	Arithmetic Compress
D0	44	VM	Average
D1	44	VM	Adjust Mean
D2	45		Illegal
C3	45		Illegal
D4	45	VM	Average Differential
D5	45	VM	Delta
D6	45		Illegal
D7	45		Illegal
D8	45	NT	Maximum of A to C, Item Count to B
D9	45	NT	Minimum of A to C, Item Count to B
DA	45	VM	Sum
DB	45	VM	Product
DC	45	VM	Vector Dot Product
DD	45		Illegal
CE	45		Illegal
CF	45	VM	Interval

*=Multi-pass

SUPER COMPUTER OPERATIONS

Index To Timing Tables By Op Code (Cont.)

OP Code	Page #	Type	Description
E0	45		Illegal
E1	45		Illegal
E2	45		Illegal
E3	45		Illegal
E4	45		Illegal
E5	45		Illegal
E6	45		Illegal
E7	45		Illegal
E8	45		Illegal
E9	45		Illegal
EA	45		Illegal
EB	45		Illegal
EC	45		Illegal
ED	45		Illegal
EE	45		Illegal
EF	45		Illegal
F0	45	LS	Logical Exclusive OR
F1	45	LS	Logical AND
F2	45	LS	Logical Inclusive OR
F3	45	LS	Logical Stroke
F4	46	LS	Logical Pierce
F5	46	LS	Logical Implication
F6	46	LS	Logical Inhibit
F7	46	LS	Logical Equivalence
F8	46	ST	Move Bytes Left
F9	46		Illegal
FA	46		Illegal
FB	46		Illegal
FC	46		Illegal
FD	46		Illegal
FE	46		Illegal
FF	46		Illegal

SUPER COMPUTER OPERATIONS

1.0 SCOPE

This timing specification is intended for persons already having familiarity with CYBER 200 concepts, terminology and general description as contained in the applicable specifications. In particular, this timing specification depends on much of the material in the CYBER 205 Functional Computer Specification 10358025. The times contained in this specification for the CYBER 205 are preliminary estimates with little verification being complete at this time.

In general, those instructions executed will be verified via simulation.

2.0 APPLICABLE DOCUMENTS

(To be added)

3.0 REQUIREMENTS

3.1 General Description

The tables in Section 3.2 are designed to provide timing data for the instruction sequences in the CYBER 205. All timing data is expressed in 20 nsec. minor cycles.

3.1 (Continued)

Scalar instructions are expressed as described below.

The ISSUE column of the tables contains the minimum number of minor cycles between the issue of the specific instruction listed in the left column and the issue of the next instruction in the sequence. Various operand or memory conflicts (as discussed later) can cause additional delay beyond this minimum time.

The Issue column of the tables is sub-divided when appropriate into three categories as defined below:

NB -- No Branch
ISB -- In Stack Branch
OSB -- Out of Stack Branch to first quarter sword. This time must be increased by 1, 2 or 3 if the Branch address is in the 2nd, 3rd or 4th quarter sword respectively.

The non-branch instructions use the entry under NB or No Branch. Example 1 illustrates a simple, no conflict, Branch sequence.

For this example and all following examples, the calculations are based on values from tables 3.1-1, 3.2-1 and 3.2-2. The times under each column are based on Issue or Busy (L/S or D/C) time plus SS, RF and MEM.

The R, S, T, A, X, Y, B, C, and Z under instruction represents the appropriate descriptors from the instruction. Non pertinent data is represented by a -.

Example 1

INSTRUCTION	F	R/G	S/X	T/A		F.P.	RESULTS			BUSY		REGISTER
	Y	B	Z	C	ISSUE	STACKED	SS	RF	MEM	L/S	D/C	RELEASE
A)	60	-	-	-	0	-	5	8	-	-	-	-
B)	25	-	-	-	1 (NB)	-	-	-	-	-	-	-
					//----------							
					11							
C)	60	-	-	-	12	-	17	20	-	-	-	-
D)	25	-	-	-	13 (OSB)	-	-	-	-	-	-	-
					//----------							
					41							
E)	-	-	-	-	42							

Instruction (B) is a branch with condition not met; therefore instruction (C) issues at time 12.

Instruction (D) is a branch with condition met to instruction (E) whose address is out of stack and falls in the third quarter sword therefore; (E)'s

Issue = Time 13 + OSB (D) + 3rd Quarter Sword
= 13 + 27 + 2
= 42.

The RESULT AVAILABLE column of the tables contains information necessary to time instruction sequences with operand dependencies. The first column, SS or shortstop, contains entries for those instructions which use the Floating Point Unit. These are the instructions which may use the shortstop feature to provide an input operand. This entry is the number of minor cycles after issue that the result operand will be available at the shortstop for use with a following instruction.

If instruction A issues at minor cycle X, any following instruction, B, needing the result of A must issue no later than minor cycle X+SS to utilize the shortstop. A floating point instruction needing the result of A, can be issued before X+SS and wait at the input of floating point for the shortstopped result of A. This allows other non-floating point instructions

SUPER COMPUTER OPERATIONS

(12,13,2F,31,32,35,36,5E,5F,7E,7F, (B0-B5)(.XX0X-X) and B6) to issue. The resulting time of an instruction that issues and waits for shortstop will be as if it had issued at the ideal time to match shortstop. A subsequent instruction requiring access to floating point will not issue any earlier than (X+SS) + 1.

If instruction B issues later than cycle X+SS, thus missing the shortstop; instruction B must wait until at least X+RF. At this time the desired operand will be available from the Register File. Example 2 illustrates operations using shortstop.

NOTE: Instructions that require 64-bit operands cannot shortstop the result of a 32-bit operation and vice versa.

Example 2

	INSTRUCTION				TIME								
	F	R/G	S/X	T/A		F.P.	RESULTS				BUSY		REGISTER
	Y	B	Z	C	ISSUE	STACKED	SS	RF	MEM		L/S	D/C	RELEASE
A)	60	-	-	12	0	-	5	8	-		-	-	-
B)	60	-	-	-	1	-	6	9	-		-	-	-
C)	60	-	-	-	2	-	7	10	-		-	-	-
D)	60	-	-	-	3	-	8	11	-		-	-	-
E)	60	-	-	-	4	-	9	12	-		-	-	-
F)	60	12	-	14	5	-	10	13	-				
G)	60	14	14	-	6	-	-	-	-				
H)	7F	-	-	-	7	(G)	-	-	17	(H)	9	-	14
					8	(G)							
I)	60	-	-	15	9	(G)							
					10	(G)	15	18	-	(G)	-	-	-
					11	-	16	19	-	(I)	-	-	-
J)	60	14	-	16	12								
					13	-	18	21	-		-	-	-
K)	60	16	15	-	14								
L)	60	-	-	-	15	(K)							
					16	(K)							
					17	(K)							
					18	(K)	23	26	-		-	-	-
					19	-	24	27	-		-	-	-
M)	34	-	-	-	20								
					21	(M)							
					22	-	25	28	-		-	-	-

Instruction (F) has a source operand conflict with instruction (A) whose SS time 5 matches instructions (F)'s earliest issue time 5, therefore no delay in issue.

Instruction (G) has a source operand conflict with instruction (F) and has to wait until instruction (F)'s SS time 10 before it can proceed but because one instruction can be stacked in front of the F.P. unit instruction (H) can issue and execute.

Instruction (J) has a source operand conflict with instruction (F). The earliest it can issue is time 12 and misses instruction (F)'s SS time 10, therefore it has to wait until RF time 13 of instruction (F).

Instruction (K) has a source conflict with instructions (I) and (J). It tries to issue as early as time 14, but has to wait until time 18 which is the latest SS time of instruction (I) and J). It will stack in front of the F.P. unit but does no good in this case because the following instruction needs the same unit.

Instruction (M) has a register file write conflict and must wait until the next available RF opening before it can issue.

The last column under RESULT AVAILABLE (MEM) contains entries for those scalar instructions (13, 32, 5F, 7F) which store a result into Central Memory. The time listed is the minimum time until the operand is in memory and available for use. The time may also be increased by 4 minor cycles if the desired memory bank is busy.

The UNIT BUSY column of Table 3.2-1 concerns instructions issued to either the Divide/Convert (D/C) Unit or the Load/Store Unit (L/S). The Divide/ Convert Unit executes the 10, 11, 4C, 4F, 53, 6C, 6F and 73 instructions.

This unit is the only portion of Floating Point which is not a Pipe; thus the appropriate unit busy time listed in Table 3.2-1 must elapse before another instruction can be issued to the Divide/Convert Unit. Floating Point instructions other than these eight may be issued to Floating Point while the Divide/Convert Unit is busy.

A second instruction from the set of eight can be issued, but will be held in front of the Floating Point Unit and issuing of non-floating point instructions will continue.

The Load/Store Unit executes the 12, 13, 32, 5E, 5F, 7E and 7F instructions. There are six address registers in the Load/Store Unit which enable requests to be stacked and executed in the proper order. The 12, 5E and 7E instructions each require one register and can be executed (in the absence of memory conflicts) at the rate of one load per minor cycle. The 5F and 7F instructions each require two address registers and can be executed at one store per two minor cycles. The 13 and 32 (not if G bits 2 & 3 = 0) instructions each require two address registers and can be executed at one per 14 and 15 minor cycles, respectively. The rate is determined from the unit busy (see table 3.2-1).

The LOAD/STORE Unit is then capable of streaming Load/Store instructions (other than the 13 and 32) at one minor cycle per load and two minor cycles per store assuming no Memory or Register File conflicts. For example, a stream of N loads will execute in N + 14 minor cycles from the issue of the first load until the operand from the last load is available in the Register File. The N + 14 comes from table 3.2-1 where the 14 is the RF time minus 1.

The method of determining the time of issue for a load/store instruction that is waiting on address registers to be released from previous load/store operation is as follows:

A + B + 5 M.C. (minor cycles)
(for register release)

Where A is the unit busy time from Table 3.2-1 for the instructions that release the register.

Where B is the sum of unit busies stacked up prior to A.

The method for determining result calculations for an instruction that waits on a unit busy and is already issued, is to substitute the previous unit busy accumulated time and add the instructions SS, RF and/or busy time. See Example 3.

SUPER COMPUTER OPERATIONS

A sequence of loads and store do not reference memory out of order. Therefore, a store followed by a load to the same location results in a delayed load of one memory busy (4 M.C.) minimum.

Example 3

	INSTRUCTION				TIME							
	F	R/G	S/X	T/A		F.P.	RESULTS			BUSY		REGISTER
	Y	B	Z	C	ISSUE	STACKED	SS	RF	MEM	L/S	D/C	RELEASE
A)	60	-	-	-	0	-	5	8	-	-	-	-
B)	7E	-	-	-	1	-	-	16	-	2	-	7
C)	13	-	-	-	2	-						
					3	-	-	-	25	16	-	21
D)	13	-	-	-	4							
					5	-	-	-	39	30	-	35
E)	60	-	-	-	6	-	11	14	-	-	-	-
F)	7E	-	-	-	7	-	-	45	-	31	-	36
G)	7E	-	-	-	8	-	-	46	-	32	-	37
H)	7E	-	-	-	9							
					--------//--------------							
					21	-	-	47	-	33	-	38
I)	13	-	-	-	22							
					--------//--------------							
					36	-	-	-	56	47	-	52
J)	6F	-	-	-	37	-	91	94	-	-	87	-
K)	6F	-	-	-	38							
L)	-	-	-	-	39	(K)						
					--------//--------------							
					87	(K)	141	144	-	-	137	-

Instruction (B) is a load instruction and its calculations are:

L/S Busy = Issue + L/S = 1 + 1 = 2
Register Release = Busy + 5 = 2 + 5 = 7
Result RF = Issue + RF = 1 + 15 = 16

Instruction (C) is a byte store and calculations are based on issue time 2 or prior L/S busy time 2, whichever is largest, in this case either can be used.

L/S Busy = L/S busy + L/S = 2 + 14 = 16
Register Release = L/S busy + 5 = 16 + 5 = 21
Mem = L/S busy + Min = 2 + 23 = 25

Instruction (D) is a byte store and the busy time is greater than issue time therefore:

L/S Busy = L/S busy + L/S = 16 + 14 = 30
Register Release = L/S busy + 5 = 30 + 5 = 35
Mem = L/S busy + Mem = 16 + 23 = 39

Load instruction (F's) times will be based on busy.

L/S Busy = L/S busy + L/S = 30 + 1 = 31
Register Release = L/S busy + 15 = 31 + 5 = 36
RF = L/S busy + RF = 30 + 15 = 45

Instruction (G) is the same as instruction (F) therefore all calculations will be 1 greater.

Instruction (H) is the same as instruction (G) but its issue will be held up because the 6 registers in Load/Store are busy by instruction (C) uses 2, (D) uses 2, (F) and (G) uses 1 apiece and will not issue until register release of instruction (C) time 21.

L/S Busy = L/S busy + L/S = 32 + 1 = 33
Register Release = L/S busy + 5 = 33 + 5 = 38
RF = L/S busy + RF = 32 + 15 = 47

Instruction (I) cannot issue because 5 of 6 registers are busy and must wait until instruction (D) register release time 35, therefore:

L/S busy = L/S busy + L/S = 33 + 14 = 47
Register Release = L/S busy + 5 = 47 + 5 = 52
Mem = L/S busy + Mem = 33 + 23 = 56

Instructions (J) and (K) use the D/C busy which is not busy therefore (J) issues at time 37 and its calculations are based on time 37.

SS = ISSUE + SS = 37 + 54 = 91
RF = ISSUE + RF = 37 + 57 = 94
D/C Busy = ISSUE + D/C = 37 + 50 = 87

Instruction (K) calculations will be based on D/C busy which is greater than issue time.

SS = D/C busy + SS = 87 + 54 = 141
RF = D/C busy + RF = 87 + 57 = 144
D/C busy = D/C busy + D/C = 87 + 50 = 137

Instruction (L) can issue at time 39 and execute if it is a non F.P. instruction, otherwise it must wait until time 88.

There are three Operand Dependencies which delay issue that must be considered.

1. Source operand conflict -- an instruction requiring the result of a previous instruction as an input operand waits until the operand becomes available.

2. Output operand conflict -- an instruction output to the same Register File location as a previously issued, but slower instruction, waits until the previous instruction stores its result in the Register File, unless it also has a source operand conflict then it will go at the shortstop time.

3. Register File Write conflict -- an instruction cannot issue if its result arrives at the Register File at the same minor cycle as the result of a previously issued but slower instruction.

Table 3.1-1 pertains to instructions having greater than 1 minor cycle issue time.

The first column lists the appropriate instructions. The second column indicates the minor cycle of issue that a specific operand is required. The third column indicates the availability of shortstop for that specific operand.

Example 4

	INSTRUCTION				TIME								
	F	R/G	S/A	T/A		F.P.	Results			BUSY		REGISTER	
	Y	B	Z	C	ISSUE	STACKED	SS	RF	MEM	L/S	D/C	RELEASE	
A)	60	-	-	12	0	-	5	8	-	-	-	-	
B)	31	12	-	-	1								
					7(OSB)								
C)	60	-	-	35	31	-	36	39	-	-	-	-	
D)	20	35	35	-	32								
					33								
					34								
					35								
					36(ISB)								
E)	60	-	-	-	47								

Instruction (B) branches out of stack to instruction (C) (in second quarter sword). Its issue occurs at time 7 as opposed to time the RF time of instruction that its depended on. The R descriptor is in the second cycle (from table 3.1) therefore:

Issue = Issue + OSB+1 (second quarter sword)
= 7 + 23 + 1 = 31

Instruction (D) branches in stack to instruction (E) and is dependent on instruction (D)'s SS time 36. Because (D) R & S descriptors are read in the second cycle, issue starts at time 35. Calculations are:

Issue = Issue + ISB = 35 + 12 = 47

Timing descriptions for those instructions in the vector/string units are as follows.

Each instruction has two values in its timing equations. One value is issue time (INB) which is the amount of time the issue unit spends on this instruction. The other value is a busy time (VBA). Overlap does occur in most cases between vector/string instructions (table 3.2-2) and scalar instructions (table 3.2-1). The exceptions are those instructions in table (3.2-2) that do not have a (VBA) time.

The busy time (VBA) represents the number of cycles of parallel work that can be accomplished by the scalar instructions before the next vector/string unit instruction will be executed. Overlap does not occur if the scalar instruction requires the load/store unit instructions (04, 0C, 0D, 0F, 12, 13, 32, 5E, 5F, 7E and 7F). In this case, busy time (VBA) is used to predict the execution start time for these instructions the same way it is used to predict execution start time for a vector/string instruction. Their results and busy times will be increased from (15-23) minor cycles because of common hardware between table 3.2-2 instructions and load/store unit instructions. The start time of a vector string instruction is influenced by a preceding load/store unit instruction. The vector/string operation will start when the load/store unit busy has 10 minor cycles remaining.

Table 3.2-2 lists the instruction OP-CODE in the first column, then the issue time in minor cycle, the busy times in minor cycles and in the last four

3.1 (Continued)

columns are the rate factors or pipe line size (PLS). The PLS is given for both 2 and 4 pipe size CPU's. PLS is the number of elements per minor cycle and is constant or has different values for elements of 32 bit (32M) and 64 bit (64M) moves.

Example 5 - C=40 and PLS=2 for vectors

	INSTRUCTION F	S/G B	R/X Z	T/A C	TIME Issue	F.P. Stacked	Results SS	RF	MEM
	Y								
A)	60	-	-	-	0	-	5	8	-
B)	60	-	-	-	1	-	6	9	-
C)	60	-	-	-	2	-	7	10	-
D)	63	-	-	12	3	-	4	7	-
E)	80	00	-	-	4+(INB)				
	-	12	-	-					
			----------//----------						
F)	60	-	-	-	21--	-	26	29	-
G)	60	-	-	-	22 |	-	27	30	-
H)	60	-	-	-	23 |	-	28	31	-
I)	60	-	-	-	24 +VBA(E)	-	29	32	-
J)	60	-	-	-	25 +INB(O)	-	30	33	-
K)	60	-	-	-	26 |	-	31	34	-
L)	60	-	-	-	27 |	-	32	35	-
M)	60	-	-	-	28 |	-	33	36	-
N)	60	-	-	-	29 |	-	34	37	-
O)	8F	00	-	-	30 |				
	-	-	-	-	|				
			----------//-|--------						
P)	60	-	-	-	93<-	-	98	111	-
Q)	60	-	-	-	94-|	-	99	112	-
R)	60	-	-	-	95 |	-	100	113	-
S)	60	-	-	-	96 +VBA(O)	-	101	114	-
T)	60	-	-	-	97 |	-	102	115	-
U)	7E	-	-	-	280<-	-			
						-	-	310	-
V)	60	-	-	-	281	-	286	289	-

Instruction (E) will start issue at time 4 even though designator B's contents are not available until time 6 (instructions (C) RF time) because table 3.1-1 shows it is not needed until the fourth cycle of the instruction.

Instruction (F) will issue at time 21 using table 3.2-2.

3.1 (Continued)

$$Issue = Issue + INB(E) = 4 + 17 = 21$$

Instruction (P) will issue at time 93 because the VBA (=34 + |40| of E), is greater than issue time spent by
1 21
instructions (F) through (N) therefore:

$$Issue = \underset{|40|}{Issue\ (F)} + VBA(E) + INB(O) = 21 + 34 \underset{1\ 21}{} + 18 = 93$$

Instruction (U) will issue at time 314 because instruction (U) is a load and cannot start until instruction (O)'s busy is expended therefore:

$$Issue = Issue\ (P) + VBA(O) = 93 + 62 + \overline{|\underset{1.32}{40}|} = 280$$

Instruction (U)'s result to register file RF will be delayed by an additional 15 minor cycles minimum because of the previous vector instruction.

Timing of link operation

The total INB time for a link operation is the sum of INB's for the 56 instruction F1 and F2. The VBA time is the longer of F1 or F2.

All times are approximate due to small variations listed under assumptions for vectors.

Assumptions

Calculation using tables does assume an ideal situation and time must be added for the following:

1. Alignment of slot time for memory reference, up to 7 minor cycles.
2. Space table searches in job mode: 34 + $\frac{N}{2}$ where N is number of element to search.
3. Interrupts.
4. Vector conflict with I/O 0-32 minor cycles only if there is a control vector. Not a high probability.

3.1 (Continued)

5. Starting bank address of input streams and output stream could add 4, 8 or 12 cycles.

6. Register file conflicts are calculated the same as scalar instructions by using Table 3.1.1. The exact cycle the register is needed for the vector instruction can be obtained from the table.

Table 3.1-1

ORDER THAT DESIGNATORS ARE READ FOR MULTIPLE ISSUE INSTRUCTION AND IF THEY CAN USE SHORTSTOP OF A PREVIOUS INSTRUCTION

INSTRUCTION	CYCLE	DESCRIPTORS	SHORTSTOP
13	1	P&S&T	No
	2	T	No
20-27	1	T	No
	2	P&S	Yes
2F	1	S	No
	2	T	No
	3	T	No
31&35	1	S&T	No
	2	R	No
	3	P	No
32	1	S	No
	2	T	No
36	1	S&T	No
	2	R	No
	3	R	No
5F	1	P&S&T	No
	2	T	No
6D	1	R&S&T	Yes (R&S)
	2	T	No
7F	1	P&S&T	No
	2	T	No
80-85.X00X-X	1	B&Y&Z	No
	2	X&A	No
	3	C&Z	No
	4	X&A&C	No
80-85.X01X-X	1	B&Y&Z	No
	2	X&A&C	Yes (X+A)
	3	C&Z	No
80-85.X10X-X	1	B&Y	No
	2	X&A	Yes
80-85.X11X-X	1	B&Y	No
	2	X&A+Y	Yes (X+A)
86	1	R	No

Table 3.1-1

ORDER THAT DESIGNATORS ARE READ FOR MULTIPLE ISSUE VECTOR UNIT INSTRUCTIONS OF WHICH CANNOT USE SHORTSTOP OF A PREVIOUS INSTRUCTION

INSTRUCTION	CYCLE	DESCRIPTORS	SHORTSTOP
09	1	S & T	No
	2	Store Trace Reg.	No
14->16, 1C->1F	1	S & T	No
	2	R	No
28	1	T & S	No
33	1	T	No
	2	Relative Branch	No
70	1	Store Trace Reg.	No
	2	S & T	No
	3	R	No
80->8C, 8F 90->98	1	C	No
		C+1 IF GBIT 2=1	
	2	Z	No
	3	A IF GBIT3=0	No
		X	
	4	B IF GBIT4=0	No
		Y	
	5	A IF GBIT3=1	No
		B IF GBIT4=1	
9C	1	C	No
		C+1 IF GBIT2=1	
	2	Z	No
	3	A IF GBIT3=0	No
		X	
	4	--	No
	5	A IF GBIT3=1	No
9D	1	C	No
		C+1 IF GBIT2=1	
	2	Z	No
	3	A IF GBIT3=0	No
		X	
	4	B IF GBIT4=0	No
		Y	
	5	A IF GBIT3=1	No
		B IF GBIT4=1	

Table 3.1-1

ORDER THAT DESIGNATORS ARE READ FOR MULTIPLE ISSUE VECTOR UNIT INSTRUCTIONS OF WHICH CANNOT USE SHORTSTOP OF A PREVIOUS INSTRUCTION

INSTRUCTION	CYCLE	DESCRIPTORS	SHORTSTOP
A0-A2, A4-A6	1	C & Z	No
A8, A9, AB,	2	Z	No
AC, AF	3	A IF GBIT3=0	No
		X	
	4	B IF GBIT4=0	No
		Y	
	5	A IF GBIT3=1	No
		B IF GBIT4=1	
B7	1	STORE TRACE REG.	No
	2	C & A	No
	3	Z	No
	4	A & X	No
	5	B IF GBIT4=0	No
		Y	
	6	B IF GBIT4=1	No
B8	1	C	No
		C+1 IF GBIT2=1	
	2	Z	No
	3	A IF GBIT3=0	No
		X	
	4	B IF GBIT4=0	No
		Y	
	5	A IF GBIT3=1	No
		B IF GBIT4=1	
BA	1	C & A	No
	2	Z	No
	3	A & X	No
	4	B & Y	No
	5	--	No

Table 3.1-1

ORDER THAT DESIGNATORS ARE READ FOR MULTIPLE ISSUE VECTOR UNIT INSTRUCTIONS OF WHICH CANNOT USE SHORTSTOP OF A PREVIOUS INSTRUCTION

INSTRUCTION	CYCLE	DESCRIPTORS	SHORTSTOP
8B, BC, BD, C0-C3	1	C & Z	No
	2	Z	No
	3	A IF GBIT3=0	No
		X	
	4	B IF GBIT4=0	No
		Y	
	5	A IF GBIT3=1	No
		B IF GBIT4=1	
C4-C7	1	Z	No
	2	Z	No
	3	A IF GBIT3=0	No
		X	
	4	B IF GBIT4=0	No
		Y	
	5	A IF GBIT3=1	No
		B IF GBIT4=1	
C8-CB CF	1	C & A	No
	2	Z	No
	3	A IF GBIT3=0	No
		X	
	4	B IF GBIT4=0	No
		Y	
	5	A IF GBIT3=1	No
		B IF GBIT4=1	
CC	1	C & B	No
	2	A & X	No
D0, D1, D4, D5	1	C	No
		C+1 IF GBIT2=1	
	2	Z	No
	3	A IF GBIT3=0	No
		X	
	4	B IF GBIT4=0	No
		Y	
	5	A IF GBIT3=1	No
		B IF GBIT4=1	

Table 3.1-1

ORDER THAT DESIGNATORS ARE READ FOR MULTIPLE ISSUE VECTOR UNIT INSTRUCTIONS OF WHICH CANNOT USE SHORTSTOP OF A PREVIOUS INSTRUCTION

INSTRUCTION	CYCLE	DESCRIPTORS	SHORTSTOP
08, 09	1	C	No
	2	Z	No
	3	A IF GBIT3=0	No
		X	
	4	B IF GBIT4=0	No
		Y	
	5	A IF GBIT3=1	No
		B IF GBIT 4=1	
0A	1	C & C+1	No
	2	Z	No
	3	A IF GBIT3=0	No
		X	
	4	B IF GBIT4=0	No
		Y	
	5	A IF GBIT3=1	No
		B IF GBIT4=1	
0B	1	C	No
	2	Z	No
	3	A IF GBIT3=0	No
		X	
	4	B IF GBIT4=0	No
		Y	
	5	A IF GBIT3=1	No
		B IF GBIT4=1	

Table 3.1-1

ORDER THAT DESIGNATORS ARE READ FOR MULTIPLE ISSUE VECTOR UNIT INSTRUCTIONS OF WHICH CANNOT USE SHORTSTOP OF A PREVIOUS INSTRUCTION

INSTRUCTION	CYCLE	DESCRIPTORS	SHORTSTOP
DC	1	C & C+1	No
	2	Z	No
	3	A IF GBIT3=0	No
		X	
	4	B IF GBIT4=0	No
		Y	
	5	A IF GBIT3=1	No
		B IF GBIT4=1	
DF	1	C	No
		C+1 IF GBIT 2=1	
	2	Z	No
	3	A & B	No
F0-F7	1	C & Z	No
	2	C & Z	No
	3	A & X	No
	4	B & Y	No
	5	--	
F8	1	C & Z	No
	2	Z	No
	3	A & X	No

SUPER COMPUTER OPERATIONS

3.2 Basic Instruction Timing

Table 3.2-1 Scalar Instruction Times

Instructions	Issue NB	Issue ISB	Issue OSB	Result Avail. S.S.	Result Avail. R.F.	Result Avail. MEM	Unit Busy L/S	Unit Busy D/C
03	1							
04	20							
05	-	-	32					
06	4							
OC	34							
OD	48							
**OE	20	31	46					
OF-IN ARS	9	--	--	--	22			
OF-OUT ARS	89 + N				92+N			
		2	--	--	2			
10	1	--	--	21	24			17
11	1	--	--	54	57			50
12	1	--	--	--	15		1*	
13	2	--	--	--	--	23	14*	
20	11	12	27	--	--	--		

* MUST ADD 5 MC FOR REGISTER RELEASE
* MUST BE TREATED LIKE A VECTOR INSTRUCTION-N=NUMBER WORDS SEARCHED

Table 3.2-1 Scalar Instruction Times (cont'd.)

	Issue			Result Avail.			Unit Busy	
Instructions	NB	ISB	OSB	S.S.	R.F.	MEM	L/S	D/C
21	11	12	27	--	--	--		
22	11	12	27	--	--	--		
23	11	12	27	--	--	--		
24	11	12	27	--	--	--		
25	11	12	27	--	--	--		
26	11	12	27	--	--	--		
27	11	12	27	--	--	--		
2A	1	--	--	3	6	--		
2B	1	--	--	3	6	--		
2C	1	--	--	3	6	--		
2D	1	--	--	3	6	--		
2E	1	--	--	3	6	--		
2F	7	8	23	--	7	--		
30	1	--	--	3	6	--		
31	7	8	23	--	7	--		

* MUST ADD 5 MC FOR REGISTER RELEASE

Table 3.2-1 Scalar Instruction Times (cont'd.)

Instructions	Issue NB	Issue ISB	Issue OSB	Result Avail. S.S.	Result Avail. P.F.	Result Avail. MEM	Unit Busy L/S	Unit Busy D/C
32.0X	2	--	--	--	--	--	--	
32.1X-3X	2	--	--	--	--	24	15*	
32.4X	--	9	24	--	--	--	--	
32.5X-7X	--	9	24	--	--	24	15*	
32.8X	20	21	36	--	--	--	--	
32.9X-BX	20	21	36	--	--	24	15*	
32.CX	20	21	36	--	--	--	--	
32.DX-FX	20	21	36	--	--	24	15*	
**33	20	31	46					
34	1	--	--	3	6	--		
35	7	8	23	--	7	--		
36,R=T,S=0	4	--	--	--	6	--		
36,R=T,S≠0		9	24	--	8			
36,R≠T		8	23	--	7	--		
38	1	--	--	1	4	--		
3C	1	--	--	5	8	--		
3D	1	--	--	5	8	--		

* MUST ADD 5 MC FOR REGISTER RELEASE
** MUST BE TREATED LIKE A VECTOR INSTRUCTION

Table 3.2-1 Scalar Instruction Times [cont'd.]

	Issue			Result Avail.			Unit Busy	
Instructions	NB	ISB	OSB	S.S.	P.F.	MEM	L/S	D/C
3E	1	--	--	1	4	--		
3F	1	--	--	1	4	--		
40	1	--	--	5	8	--		
41	1	--	--	5	8	--		
42	1	--	--	5	8	--		
44	1	--	--	5	8	--		
45	1	--	--	5	8	--		
46	1	--	--	5	8	--		
48	1	--	--	5	8	--		
49	1	--	--	5	8	--		
4B	1	--	--	5	8	--		
4C	1	--	--	30	33	--		26
4D	1	--	--	1	4	--		
4E	1	--	--	1	4	--		
4F	1	--	--	30	33	--		26
50	1	--	--	5	8	--		
51	1	--	--	5	8	--		
52	1	--	--	5	8	--		
53	1	--	--	29	32	--		25

Table 3.2-1 Scalar Instruction Times (cont'd.)

	Issue			Result Avail.			Unit Busy	
Instructions	NB	ISB	OSB	S.S.	P.F.	MEM	L/S	D/C
54	1	--	--	5	8	--		
55	1	--	--	5	8	--		
58	1	--	--	1	4	--		
59	1	--	--	5	8	--		
5A	1	--	--	3	6	--		
5B	1	--	--	3	6	--		
5C	1	--	--	5	8	--		
5D	1	--	--	5	8	--		
5E	1	--	--	--	15	--	1*	
5F	2	--	--	--	--	10	2*	
60	1	--	--	5	8	--		
61	1	--	--	5	8	--		
62	1	--	--	5	8	--		
63	1	--	--	1	4	--		
64	1	--	--	5	8	--		
65	1	--	--	5	8	--		
66	1	--	--	5	8	--		
67	1	--	--	1	4	--		

* MUST ADD 5 MC FOR REGISTER RELEASE

Table 3.2-1 Scalar Instruction Times [cont'd.]

	Issue			Result Avail.			Unit Busy	
Instructions	NB	ISB	OSB	S.S.	P.F.	MEM	L/S	D/C
68	1	--	--	5	8	--		
69	1	--	--	5	8	--		
6B	1	--	--	5	8	--		
6C	1	--	--	54	57	--		50
6D	2	--	--	4	7	--		
6E	1	--	--	3	6	--		
6F	1	--	--	54	57	--		50
70	1	--	--	5	8	--		
71	1	--	--	5	8	--		
72	1	--	--	5	8	--		
73	1	--	--	53	56	--		49
74	1	--	--	5	8	--		
75	1	--	--	5	8	--		
76	1	--	--	5	8	--		
77	1	--	--	5	8	--		
78	1	--	--	1	4	--		
79	1	--	--	5	8	--		
7A	1	--	--	3	6	--		

Table 3.2-1 Scalar Instruction Times [cont'd.]

	Issue			Result Avail.			Unit Busy	
Instructions	NB	ISB	OSB	S.S.	R.F.	MEM	L/S	D/C
7B	1	--	--	3	6	--		
7C	1	--	--	3	6	--		
7E	1	--	--	--	15	--	1*	
7F	2	--	--	--	---	10	2*	
B0.X00X-X	8	9	24	--	8	--		
B0.X01X-X	3	--	--	5	5+8**	--		
B0.X10X-X	11	12	27	--	--	--		
B0.X11X-X	2	--	--	6	9	--		
B1.X00X-X	8	9	24	--	8	--		
B1.X01X-X	3	--	--	5	5+8**	--		
B1.X10 X-X	11	12	27	--	--	--		
B1.X11 X-X	2	--	--	6	9	--		
B2.X00 X-X	8	9	24	--	8	--		
B2.X01 X-X	3	--	--	5	5+8**	--		
B2.X10 X-X	11	12	27	--	--	--		
B2.X11 X-X	2	--	--	6	9	--		

* MUST ADD 5 MC FOR REGISTER RELEASE.

**Output to be stored in Register C is available at 5 cycles and Y at 8 cycles. Y may be used from the Shortstop at time 5. C can not be shortstopped.

Table 3.2-1 Scalar Instruction Times [cont'd.]

	Issue			Result Avail.			Unit Busy	
Instructions	NB	ISB	OSB	S.S.	R.F.	MEM	L/S	D/C
B3.X00 X-X	8	9	24	--	8	--		
B3.X01 X-X	3	--	--	5	5+8**	--		
B3.X10 X-X	11	12	27	--	--	--		
B3.X11 X-X	2	--	--	6	9	--		
B4.X00 X-X	8	9	24	--	8	--		
B4.X01 X-X	3	--	--	5	5+8**	--		
B4.X10 X-X	11	12	27	--	--	--		
B4.X11 X-X	2	--	--	6	9	--		
B5.X00 X-X	8	9	24	--	8	--		
B5.X01 X-X	3	--	--	5	5+8**	--		
B5.X10 X-X	11	12	27	--	--	--		
B5.X11 X-X	2	--	--	6	9	--		
B6		8	23	--	--			
BE	1	--	--	1	4			
BF	1	--	--	1	4			
CD	1	--	--	1	4			
CE	1	--	--	1	4			

**Output to be stored in Register C is available at 5 cycles and Y at 8 cycles. Y may be used from the Shortstop at time 5. C can not be shortstopped.

Table 3.2-2

Definitions for Table 3.2-2

32M = 32 Mode Operation
64M = 64 Mode Operation
PLS = Pipe Line Size (Number of elements per minor cycle)

$\lceil \quad \rceil$ = The ceiling defined as the nearest integer greater than or equal

A, B, C, X, Y, Z, R, S, T > These letters stand for usable length of their respective fields, that is: B's field length of a field etc. Usable field length means given field or sometimes requires the subtraction of its offset.

$$CR = \left\lceil \frac{C}{PLS} \right\rceil$$

$$ZR = \left\lceil \frac{Z}{PLS} \right\rceil$$

$$BR = \left\lceil \frac{B}{PLS} \right\rceil$$

$$AR = \left\lceil \frac{A}{PLS} \right\rceil$$

$$ABR = \left\lceil \frac{A}{PLS} \right\rceil \text{ OR } \left\lceil \frac{B}{PLS} \right\rceil \text{ Use smaller of the two}$$

$$TRA = \left(\frac{I}{IP} \right) \left(\left\lceil \frac{R}{16} \right\rceil + \left\lceil \frac{S}{16} \right\rceil \right)$$

Table 3.2-2

Definitions for Table 3.2-2 (Continued)

$$TRB = \left(\frac{\left(\frac{I}{R+S} \right)}{\left(\frac{P}{16} \right)} + \frac{S}{16} \right)$$

$$TRC = \left(\frac{\left(\frac{I}{S} \right)}{\left(\frac{R}{16} \right)} + \frac{S-R}{16} \right)$$

$$TR = \frac{I}{2} \quad \text{or} \quad \frac{R}{2} \quad \text{Use larger of the two results.}$$

$$RR = \frac{R}{PLS}$$

$$TRD = \frac{I}{PLS}$$

Table 3.2-2
VECTOR/STRING INSTRUCTION TIMES

Instruction	Issue INB	Busy VBA	Notes	PLS 2 Pipe 32M	PLS 2 Pipe 64M	PLS 4 Pipe 32M	PLS 4 Pipe 64M
00							
01	23						
02	23						
07	23						
08	14						
09	MONITOR	TO JOB					
	187	TWO PIPE					
	123	FOUR PIPE					
	JOB TO	MONITOR					
	190	TWO PIPE					
	126	FOUR PIPE					
0A	14						
0B	23						
14	14	54+TRA	------	16	16	16	16
15	14	35+TRB	------	16	16	16	16
16	14	35+TRB	------	16	16	16	16
17	23						
18	23						
19	23						
1A	23						
1B	23						
1C	14	31+TRC	------	16	16	16	16
1D	14	31+TRC	------	16	16	16	16
1E	16	33+RR	-----	16	16	16	16
1F	16	37+RR	-----	16	16	16	16
28	16	37+TRD	------	2	2	2	2
29	23						
37	15						
39	15						
3A	15						
3B	20						

Table 3.2-2
VECTOR/STRING INSTRUCTION TIMES

Instruction	Issue	Busy	Notes	PLS			
				2 Pipe		4 Pipe	
	INB	VBA		32M	64M	32M	64M
43	23						
47	23						
4A	23						
56	15						
57	23						
6A	23						
70 R=0	28+TR						
70 R≠0	56+TR						
80	17	34+CR	-----	4	2	8	4
81	17	34+CR	-----	4	2	8	4
82	17	34+CR	-----	4	2	8	4
83	17	34+CR	-----	-	2	-	4
84	17	34+CR	-----	4	2	8	4
85	17	34+CR	-----	4	2	8	4
86	17	34+CR	-----	4	2	8	4
87	17	34+CR	-----	-	2	-	4
88	18	34+CR	-----	4	2	8	4
89	18	34+CR	-----	4	2	8	4
8A	17	34+CR	-----	-	2	-	4
8B	18	34+CR	-----	4	2	8	4
8C GO=1	18	50+CR	-----	.61	-	1.22	-
8C GO=0	18	62+CR	-----	-	.32	-	.64
8D	23						
8E	23						
8F GO=1	18	50+CR	-----	.61	-	1.22	-
8F GO=0	18	62+CR	-----	-	.32	-	.64
90	17	34+CR	-----	4	2	8	4
91	17	34+CR	-----	4	2	8	4
92	17	34+CR	-----	4	2	8	4
93 GO=1	17	50+CR	-----	.61	-	1.22	-
93 GO=0	17	62+CR	-----	-	.32	-	.64
94	17	34+CR	-----	4	2	8	4

Table 3.2-2
VECTOR/STRING INSTRUCTION TIMES

Instruction	Issue INB	Busy VBA	Notes	PLS 2 Pipe 32M	PLS 2 Pipe 64M	PLS 4 Pipe 32M	PLS 4 Pipe 64M
95	17	34+CR	-----	4	2	8	4
96	17	34+CR	-----	4	-	8	-
97	17	34+CR	-----	4	-	8	-
98	17	34+CR	-----	4	2	8	4
99	17	34+CR	-----	4	2	8	4
9A	17	34+CR	-----	4	2	8	4
9B	17	34+CR	-----	4	2	8	4
9C	17	34+CR	-----	-	2	-	4
9D	17	34+CR	-----	4	2	8	4
9E	23						
9F	23						
A0	19	69+ZR	-----	4	2	8	4
A1	19	69+ZR	-----	4	2	8	4
A2	19	69+ZR	-----	4	2	8	4
A3	23			-	-	-	-
A4	19	69+ZR	-----	4	2	8	4
A5	19	69+ZR	-----	4	2	8	4
A6	19	69+ZR	-----	4	2	8	4
A7	23			-	-	-	-
A8	20	69+ZR	-----	4	2	8	4
A9	20	69+ZR	-----	4	2	8	4
AA	23			-	-	-	-
AB	20	69+ZR	-----	4	2	8	4
AC GO=1	20	85+ZR	------	.61	-	1.22	-
AC GO=0	20	97+ZR	------	-	.32	-	.64
AD	23						
AE	23						
AF GO=1	20	85+ZR	------	.61	-	1.22	-
AF GO=0	20	97+ZR	------	-	.32	-	.64
B7 G567=0	18	65+AR	------	.8	.8	.8	.8
1	63+AR	--	------	.8	.8	.8	.8
2	18	40+[A (43+3R)]	------	4	2	8	4
4	18	53+AR	------	.8	.8	.8	.8
5	61+AR	--	------	.8	.8	.8	.8
6	18	8+[A (48+3R)]	------	4	2	8	4

Table 3.2-2
VECTOR/STRING INSTRUCTION TIMES

Instruction	Issue INB	Busy VBA	Notes	PLS 2 Pipe 32M	PLS 2 Pipe 64M	PLS 4 Pipe 32M	PLS 4 Pipe 64M
88	17	34+CR	-----	3.8	1.9	3.8	1.9
89	23						
8A G567=0	17	52+AR	-----	.8	.8	.8	.8
=1	63+AR			.8	.8	.8	.8
=2	18	32+[A	-----	4	2	8	4
		(48+BR)]	-----				
=4	17	22+AR	-----	.8	.8	.8	.8
=5	32+AR			.8	.8	.8	.8
=6	18	8+[A	-----	4	2	8	4
		(48+BR)]	-----				
8B	18	36+ZR	-----	4	2	8	4
8C	16	36+ZR	-----	4	2	8	4
8D	18	40+ZR	-----	4	2	8	4
C0	21	51+ABR	-----	4	2	8	4
C1	21	51+ABR	-----	4	2	8	4
C2	21	51+ABR	-----	4	2	8	4
C3	21	51+ABR	-----	4	2	8	4
C4	20	36+ZR	-----	4	2	8	4
C5	20	36+ZR	-----	4	2	8	4
C6	20	36+ZR	-----	4	2	8	4
C7	20	36+ZR	-----	4	2	8	4
C8	16	15+[A	-----	4	2	8	4
		(62+BR)]	-----				
C9	16	15+[A	-----	4	2	8	4
		(62+BR]	-----				
CA	16	15+[A	-----	4	2	8	4
		(62+BR)]	-----				
CB	16	15+[A	-----	4	2	8	4
		(62+BR)]	-----				
CC	26	48+AR	-----	-	2	-	4
CF	19	47+AR	-----	4	2	8	4
D0	17	34+CR	-----	4	2	8	4
D1	19	34+CR	-----	4	2	8	4

Table 3.2-2
VECTOR/STRING INSTRUCTION TIMES

Instruction	Issue	Busy	Notes	PLS			
				2 Pipe		4 Pipe	
	INB	VBA		32M	64M	32M	64M
D2	23						
D3	23						
D4	17	34+CR	-----	4	2	8	4
D5	19	34+CR	-----	4	2	8	4
D6	23						
D7	23						
D8	21	48+AR	1	1	1	1	1
D9	21	48+AR	1	1	1	1	1
DA	21	75+AR	2	1	1	1	1
DB	21	76+AR	2	1	1	1	1
DC	21	86+ABF	2	1	1	1	1
DD	23			-	-	-	-
DE	23			-	-	-	-
DF	17	29+CR	-----	1	1	1	1
E0	23						
E1	23						
E2	23						
E3	23						
E4	23						
E5	23						
E6	23						
E7	23						
E8	23						
E9	23						
EA	23						
EB	23						
EC	23						
ED	23						
EE	23						
EF	23						
F0	17	30+CR	-----	16	16	16	16
F1	17	30+CR	-----	16	16	16	16
F2	17	30+CR	-----	16	16	16	16
F3	17	30+CR	-----	16	16	16	16

Table 3.2-2
VECTOR/STRING INSTRUCTION TIMES

Instruction	Issue	Busy	Notes	PLS			
				2 Pipe		4 Pipe	
	INB	VBA		32M	64M	32M	64M
F4	17	30+CR	-----	16	16	16	16
F5	17	30+CR	-----	16	16	16	16
F6	17	30+CR	-----	16	16	16	16
F7	17	30+CR	-----	16	16	16	16
F8	17	35+CR	-----	2	2	2	2
F9	23						
FA	23						
FB	23						
FC	23						
FD	23						
FE	23						
FF	23						

SUPER COMPUTER OPERATIONS

NOTES

1. For each new Max/Min add 16 minor cycles.
2. Each discontinuity in data flow adds 8 minor cycles.

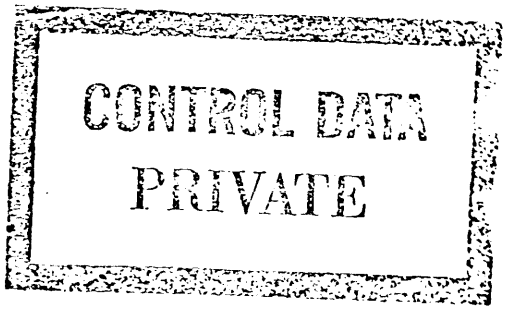
CYBER 205

NOMINAL PERFORMANCE

	2-PIPE	4-PIPE
VECTOR ADD 64-bit	1.02 + L/100	1.02 + L/200
32-bit	1.02 + L/200	1.02 + L/400
VECTOR MULTIPLY 64-bit	1.04 + L/100	1.04 + L/200
32-bit	1.04 + L/200	1.04 + L/400
LINKED ADD, MULTIPLY 64-bit	* 1.48 + L/100	1.48 + L/200
32-bit	* 1.48 + L/200	1.48 + L/400
TRANSMIT 64-bit	1.02 + L/100	1.02 + L/200
32-bit	1.02 + L/200	1.02 + L/400
VECTOR DIVIDE 64-bit	1.60 + L/16	1.60 + L/32
32-bit	1.36 + L/3.5	1.36 + L/61
SWAP single 64-bit	0.56 + R/100	
double 64-bit	1.12 + R/100	
SQUARE ROOT 64-bit	1.60 + L/16	1.60 + L/32
32-bit	1.36 + L/30.5	1.36 + L/61

ALL TIMES IN MICROSECONDS

* START-UP TIME NOT VERIFIED



CJP
8/20/80

CYBER 205

NOMINAL PERFORMANCE

		2-PIPE	4-PIPE
SPARSE ADD	64-bit	1·76 + Z/100	1·76 + Z/200
	32-bit	1·76 + Z/200	1·76 + Z/400
{all Boolean connectives}		{skips on runs of 16, 8, 4 or 2 zeros}	
{the only difference in timing Vector & Sparse is start-up;0-74}			
SPARSE MULTIPLY	64-bit	1·78 + Z/100	1·78 + Z/200
	32-bit	1·78 + Z/200	1·78 + Z/400
SPARSE DIVIDE	64-bit	2·34 + Z/14	2·34 + Z/28
	32-bit	2·10 + Z/25	2·10 + Z/50
GATHER	random	1·38 + L/40	
	periodic	0·78 + L/40	
SCATTER	random	1·66 + L/40	
	periodic	1·42 + L/40	
MASK	64-bit	1·08 + Z/100	1·08 + Z/200
	32-bit	1·08 + Z/200	1·08 + Z/400
COMPRESS	64-bit	1·00 + Z/100	1·00 + Z/200
	32-bit	1·00 + Z/200	1·00 + Z/400
MERGE	64-bit	1·16 + Z/100	1·16 + Z/200
	32-bit	1·16 + Z/200	1·16 + Z/400

ALL TIMES IN MICROSECONDS

CJP
8/20/80

CYBER 205

NOMINAL PERFORMANCE

	2-PIPE	4-PIPE
SELECT 64-bit 32-bit	1.72 + pipe rate	
COMPARE 64-bit 32-bit	1.12 + pipe rate	
SEARCH 64-bit 32-bit	0.32 + {0.30+A}*{1.24 + piperate}	
MASKED WORD SEARCH 64-bit	1.92 + L/100	1.92 + L/200
MAXIMUM	* 1.72 + L/50	
MINIMUM	* 1.72 + L/50	
SUM double precision	* 2.32 + L/50	
PRODUCT single precision	* 2.52 + L/50	
INNER PRODUCT double	* 2.32 + L/50	
BIT LOGICALS	0.94 + C/800	
BIT COMPRESS	1.36 + X/800	
BIT MERGE	0.98 + Z/800	
BIT MASK	0.98 + Z/800	

* START-UP TIME
NOT VERIFIED

CJP
8/20/80