CHAPTER 11

11-1.

a) Maximum frequency = 1/pipe stage delay = 1/0.8 ns = 1.25 GHz.

b) The latency time = 0.8 ns x 3 = 2.4ns.

c) The maximum throughput is 1 instruction per cycle or 1.25 billion instructions per second.

11-2.*

a) The latency time = 0.5 ns x 8 = 4.0 ns.

b) The maximum throughput is 1 instruction per cycle or 2 billion instructions per second.

c) The time required to execute is 10 instruction + 8 pipe stages -1 = 17 cycles *0.5ns = 8.5ns

11-3.

PC	IR	Data A	Data B	Data F	Reg
N	X	X	X	X	X
N+1	LDI R1, 1	X	X	X	X
N+2	LDI R2, 2	-1	1	X	X
N+3	LDI R3, 3	-1	2	1	X
N+4	LDI R4, 4	-1	3	2	R1 = 1
N+5	LDI R5, 5	-1	4	3	R2 = 2
N+6	LDI R6, 6	-1	5	4	R3 = 3
N+7	LDI R7, 7	-1	6	5	R4 = 4
N+8	X	-1	7	6	R5 = 5
N+9	X	X	X	7	R6 = 6
N+10	X	X	X	X	R7 = 7

Data I is not used and thus, not specified.

11-4.

Register Indirect:	Load, Store, JMR
Register, Immediate:	ADI, SBI, ANI, ORI, XRI, AIU, SIU
Relative:	BZ, BNZ, JMP, JML
None:	NOP
Register:	All instructions not listed above

11-5.

a) Right, SH = 0F = 15 = 0 + 12 + 3

47 lines = 0000 3DF3 CB4A, 35 lines = 0 0003 DF3C, 32 lines = 0000 7BE7

b) Left, SH = 1D = 29 Rt. Rotate = 64 − 29 = 35 = 32 + 0 + 3

47 lines = 4B4A 0000 0000, 35 lines = 2000 0000, 32 lines = 4000 0000

11-6.*

Cycle 1: PC = 10F

Cycle 2: PC_{-1} = 110, IR = 4418 $2F01_{16}$

Cycle 3: PC_{-2} = 110, RW = 1, DA = 01, MD = 0, BS = 0, PS = X, MW = 0, FS = 2, SH = 01, MA = 0, MB = 1
BUS A = 0000 001F, BUS B = 0000 2F01

Cycle 4: RW = 1, DA = 01, MD = 0, D0 = 0000 2F20, D1 = XXXX XXXX, D2 = 0000 00000

Cycle 5: R1 = 0000 2F20

11-7.(Errata: R6 = 01ABCDEF)

Cycle 1: IF PC = 10F

Cycle 2: DOF PC_{-1} = 110,IR = 1A61 001D

Cycle 3: EX PC_{-2} = 110,RW = 1, DA = 06, MD = 2, BS = 0, PS = X, MW = 0, FS =D, SH = 1D, MA = 0, MB = 0
BUS A = 01AB CDEF, BUS B = XXXX XXXX

Cycle 4: RW = 1, DA = 06, MD = 0, D0 = 0000 0000, D1 = XXXX XXXX, D2 = 0000 0000

Cycle 5: R6 = 0000 0000

11-8.

Cycle 1: PC = 10F

Cycle 2: PC_{-1} = 110, IR = CA71 9400

Cycle 3: PC_{-2} = 110, RW = 1, DA = 07, MD = 2, BS = 0, PS = 0, MW = 0, FS = 5, MA = 0, MB = 0, CS = 0
BUS A = 0000 F001, BUS B = 0000 000F

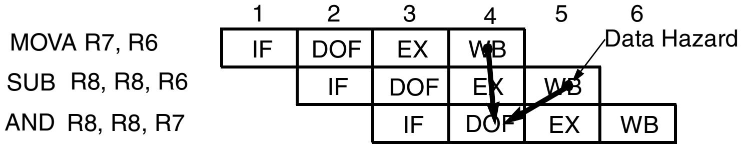
Cycle 4: RW = 1, DA = 07 MD = 0, D0 = 0000 EFF2, D1 = XXXX XXXX, D2 = 0000 0000

Cycle 5: R7 = 0000 0000

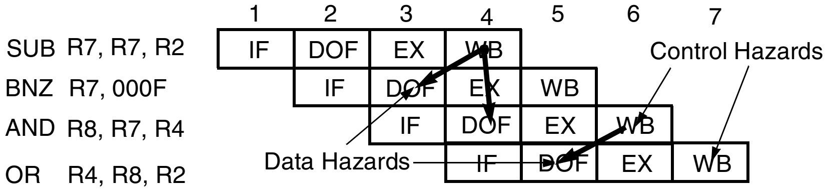
11-9.+(12-9C)

Answer not given; varies depending on synthesis software used.

11-10.*

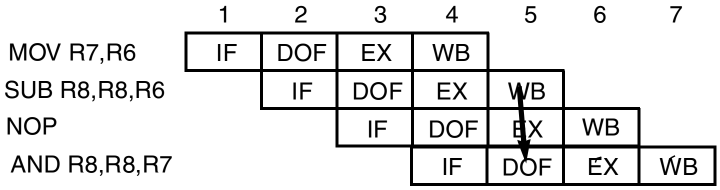


11-11.



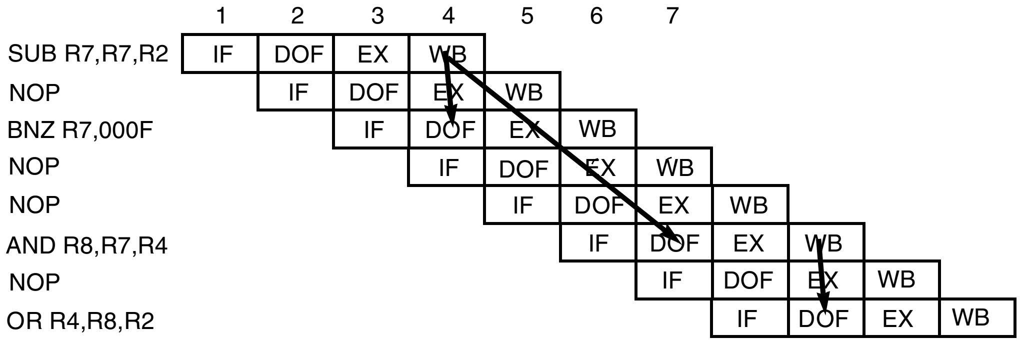
11-12.*

a)



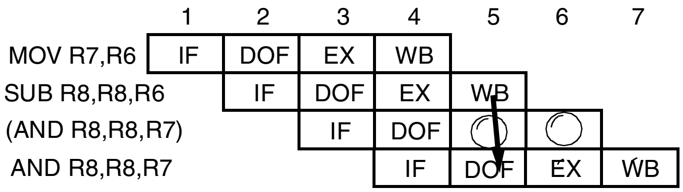
	1	2	3	4	5	6	7
MOV R7,R6	IF	DOF	EX	WB			
SUB R8,R8,R6		IF	DOF	EX	WB		
NOP			IF	DOF	EX	WB	
AND R8,R8,R7				IF	DOF	EX	WB

b)

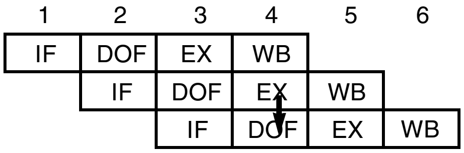


	1	2	3	4	5	6	7				
SUB R7,R7,R2	IF	DOF	EX	WB							
NOP		IF	DOF	EX	WB						
BNZ R7,000F			IF	DOF	EX	WB					
NOP				IF	DOF	EX	WB				
NOP					IF	DOF	EX	WB			
AND R8,R7,R4						IF	DOF	EX	WB		
NOP							IF	DOF	EX	WB	
OR R4,R8,R2								IF	DOF	EX	WB

11-13.



	1	2	3	4	5	6	7
MOV R7,R6	IF	DOF	EX	WB			
SUB R8,R8,R6		IF	DOF	EX	WB		
(AND R8,R8,R7)			IF	DOF	○	○	
AND R8,R8,R7				IF	DOF	EX	WB



1	2	3	4	5	6
IF	DOF	EX	WB		
	IF	DOF	EX	WB	
		IF	DOF	EX	WB

11-14.

```
                                        Time Cycle 1
IF   PC:   0000 0001
DOF  $PC_{-1}$: XXXXXXXX   IR:  XXXXXXXX
EX   $PC_{-2}$: XXXXXXXX   A:  XXXXXXXX   B:  XXXXXXXX   RW: X  DA: XX  MD: X  BS: X  PS: X  MW: X  FS: X  MB: X  CS: X
WB   D0:  XXXXXXXX   D1: XXXXXXXX   D2: XXXXXXXX   RW: X  DA: XX  MD: X

                                        Time Cycle 2
IF   PC:   0000 0002
DOF  $PC_{-1}$: 0000 0002      IR:  0A73 8800
EX   $PC_{-2}$: XXXXXXXX   A:  XXXXXXXX   B:  XXXXXXXX   RW: X  DA: XX  MD: X  BS: X  PS: X  MW: X  FS: X  MB: X  CS: X
WB   D0: : XXXXXXXX   D1: XXXXXXXX   D2: XXXXXXXX   RW: X  DA: XX  MD: X

                                        Time Cycle 3
IF   PC:   0000 0003
DOF  $PC_{-1}$: 0000 0003      IR:  9003800F
EX   $PC_{-2}$: 0000 0002      A:  0000 0030      B:  0000 0010      RW: 1   DA: 07   MD: 0   BS: 0  PS: X  MW: 0  FS: 5  MB: 0  CS:X
WB   D0:  XXXXXXXX   D1: XXXXXXXX   D2: XXXXXXXX   RW: X   DA: XX  MD: X

                                        Time Cycle 4
IF   PC:   0000 0003
DOF  $PC_{-1}$: 0000 0003    IR: 9003 800F
EX   $PC_{-2}$: 0000 0002    A:  0000 0030      B:  XXXXXXXX   RW: 0  DA: 00  MD: X   BS: 0  PS:X  MW: 0  FS: 0  MB: 1  CS: X
WB   D0:   0000 0020    D1: XXXXXXXX    D2: 0000 0000      RW: 1  DA: 07  MD: X

                                        Time Cycle 5
IF   PC:   0000 0004                                                                          R7: 0000 0020
DOF  $PC_{-1}$: 0000 0004    IR: 1083 9000
EX   $PC_{-2}$: 0000 0003    A:  0000 0020      B:  XXXXXXXX   RW: 0  DA: 00  MD: X  BS: 1  PS:1  MW: 0  FS: 0  MB: 1  CS: X
WB   D0:  0000 0030    D1: XXXXXXXX    D2: 0000 0000      RW: 0  DA: 00  MD: X          PC: 0000 0012

                                        Time Cycle 6
IF   PC:   0000 0012
DOF  $PC_{-1}$: 0000 0004     IR: 12440800
EX   $PC_{-2}$: 0000 0003     A:  0000 0020      B:  0000 0020    RW: 1  DA: 08  MD: 0   BS: 0  PS:X  MW: 0  FS: 8  MB: 0  CS: X
WB   D0:  0000 0020     D1:XXXXXXXX      D2: 0000 0000    RW: 0  DA: 00  MD: X

                                        Time Cycle 7
IF   PC:   0000 0012
DOF  $PC_{-1}$: 0000 0004     IR: 12440800
EX   $PC_{-2}$: 0000 0004     A:  XXXXXXXX     B:  0000 0010    RW: 0  DA: 00  MD: 0  BS: 0  PS:X  MW: 0  FS: 8  MB: 0  CS: X
WB   D0:  0000 0020     D1: XXXXXXXX     D2: 0000 0000    RW: 1  DA: 08  MD: 0

                                        Time Cycle 8
IF   PC:   0000 0013                                                                          R8: 0000 0020
DOF  $PC_{-1}$: 0000 0013      IR:  XXXXXXXX
EX   $PC_{-2}$: 0000 0004      A:  0000 0020       B:   0000 0010      RW: 1  DA: 04  MD: 0  BS: 0  PS: X  MW: 0  FS: 9  MB: 0  CS: X
WB   D0:  XXXXXXXX    D1: XXXXXXXX     D2: XXXXXXXX    RW: 0  DA: 00  MD: 0

                                        Time Cycle 9
IF   PC:   0000 0014
DOF  $PC_{-1}$: 0000 0014   IR:  XXXXXXXX
EX   $PC_{-2}$: 0000 0013   A:  XXXXXXXX    B:  XXXXXXXX   RW: X  DA: XX  MD: X  BS: X  PS: X  MW: X  FS: X  MB: 0  CS: X
WB   D0:  0000 0020   D1: XXXXXXXX    D2: 0000 0000       RW: 1  DA: 04  MD: 0

                                        Time Cycle 10
IF                                                                                         R4:  0000 0020
```

Fields not specified above have fixed values throughout or are unused: MA = 0, D', and SH. Based on the register contents, the branch is taken. The data hazards are avoided, but due to the control hazard, the last two instructions are erroneously executed.

11-15.*

Time Cycle 1

IF PC: 0000 0001
DOF PC_{-1}:XXXXXXXX IR: XXXXXXXX
EX PC_{-2}:XXXXXXXX A: XXXXXXXX B: XXXXXXXX RW:X DA:XX MD:X BS:X PS:X MW:X FS:X MB:X MA:X CS:X D':X
WB D0: XXXXXXXX D1:XXXXXXXX D2:XXXXXXXX RW:X DA:XX MD:X

Time Cycle 2

IF PC: 0000 0002
DOF PC_{-1}:0000 0002 IR: 0A73 8800
EX PC_{-2}:XXXXXXXX A: XXXXXXXX B: XXXXXXXX RW:X DA:XX MD:X BS:X PS:X MW:X FS:X MB:X MA:X CS:X D':X
WB D0: XXXXXXXX D1:XXXXXXXX D2:XXXXXXXX RW:X DA:XX MD:X

Time Cycle 3

IF PC: 0000 0003
DOF PC_{-1}:0000 0003 IR: 9003 800F
EX PC_{-2}:0000 0002 A: 0000 0030 B: 0000 0010 RW:1 DA:07 MD:0 BS:0 PS:X MW:0 FS:5 MB:0 MA:0 CS:X D':X
WB D0: XXXXXXXX D1:XXXXXXXX D2:XXXXXXXX RW:X DA:XX MD:X

Time Cycle 4

IF PC: 0000 0004
DOF PC_{-1}:0000 0004 IR:1083 9000
EX PC_{-2}:0000 0003 A: 0000 0020 B: XXXXXXXX RW:0 DA:XX MD:X BS:1 PS:1 MW:0 FS:0 MB:1 MA:2 CS:1 D':1
WB D0: 0000 0020 D1:XXXX XXXX D2:0000 0000 RW:1 DA:07 MD: 0 PC: 0000 0012

Time Cycle 5

IF PC: 0000 0013 R7: 0000 0020
DOF PC_{-1}: 0000 00013 IR: 1244 0800
EX PC_{-2}: 0000 0004 A: 0000 0020 B: 0000 0020 RW: 1 DA: 08 MD: 0 BS: 0 PS: X MW: 0 FS: 8 MB:0 MA:0 CS:X D':X
WB D0: 0000 0020 D1: XXXXXXXX D2: 0000 0000 RW: 0 DA: 00 MD: 0

Time Cycle 6

IF PC: 0000 0014
DOF PC_{-1}: 0000 0014 IR: XXXX XXXX
EX PC_{-2}: 0000 0013 A: 0000 0020 B: 0000 0010 RW: 1 DA: 04 MD: 0 BS: 0 PS: X MW: 0 FS: 9 MB:0 MA:0 CS:X D':X
WB D0: 0000 0010 D1: XXXX XXXX D2: 0000 0000 RW: 1 DA: 08 MD: 0

Time Cycle 7

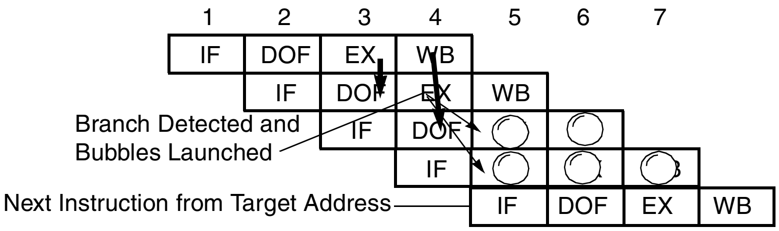
IF PC: 0000 0014 R7: 0000 0020
DOF PC_{-1}: 0000 0014 IR: XXXX XXXX
EX PC_{-2}: 0000 0013 A:XXXX XXXX B: XXXX XXXX RW: X DA: XX MD:X BS: X PS:X MW:X FS:X MB:X CS:X D’:X
WB D0: 0000 0010 D1: XXXX XXXX D2: 0000 0000 RW: 1 DA: 04 MD:0

Time Cycle 8

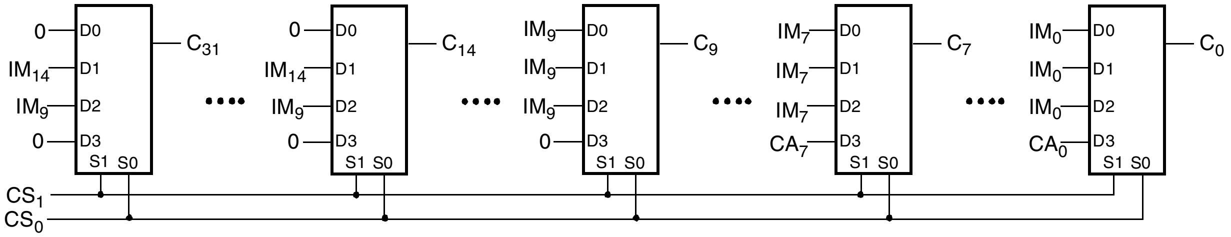
R4: 0000 0010

Fields not specified above have fixed values throughout or are unused: SH. Based on the register contents, the branch is taken. The data hazards are avoided, but due to the control hazard, the last two instructions are erroneously executed.

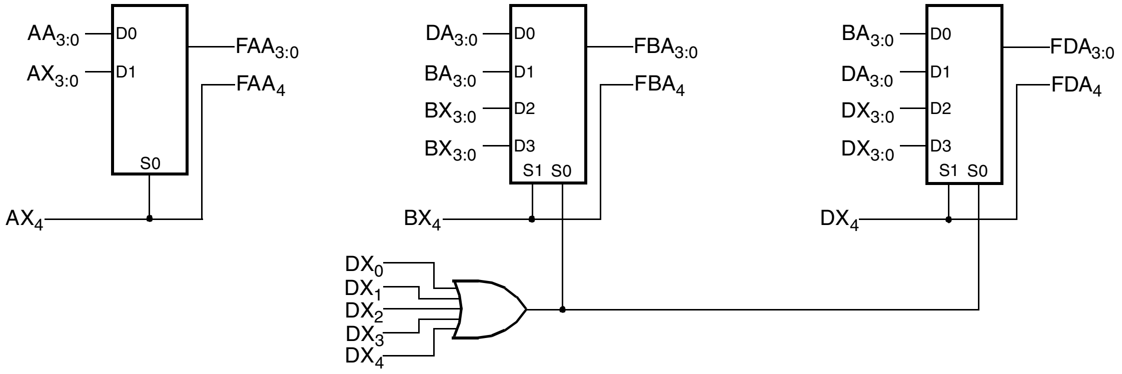
11-16.



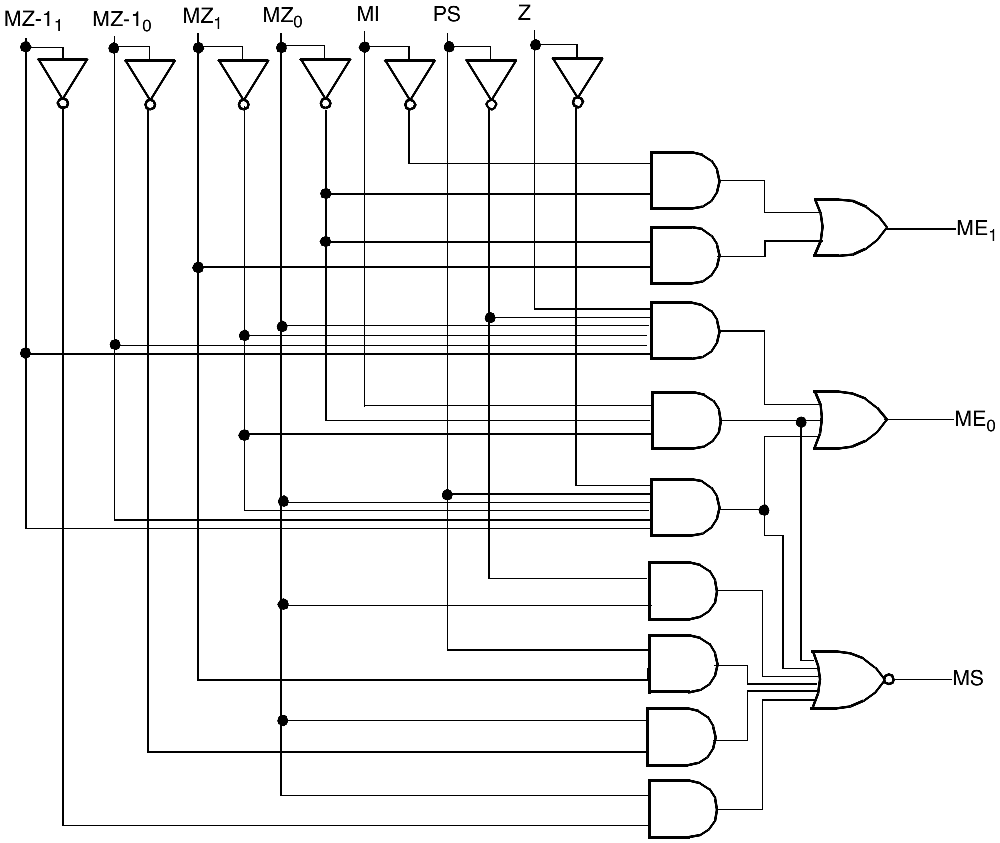
11-17.



11-18.*



11-19.



11-20. (Errata: Change (c), (d), (a) to (a), (b), (c))

(a) Branch if overflow

				R		M		P	M		L		M			
Action	Address	MZ	CA	W	DX	D	BS	S	W	FS	C	MA	B	AX	BX	CS
$R_{31} \leftarrow CC \wedge 00001$	BOV0	01	01	1	1F	0	00	0	0	8	0	10	1	00	00	11
$MC \leftarrow MC + 1$ (NOP)	BOV1	01	00	0	00	0	00	0	0	0	0	00	0	00	00	00
if (R_{31}= 0)$MC \leftarrow BOV5$ else $MC \leftarrow MC + 1$	BOV2	11	BOV5	0	00	0	00	0	0	0	0	00	0	1F	00	00
$MC \leftarrow MC + 1$ (NOP)	BOV3	01	00	0	00	0	00	0	0	0	0	00	0	00	00	00
$PC \leftarrow PC_{-1}$ + se IM	BOV4	01	00	0	00	0	11	0	0	0	0	01	1	00	00	01
$MC \leftarrow IDLE$	BOV5	00	IDLE	0	00	0	00	0	0	0	0	00	0	00	00	00

(b) Branch if greater than zero

				R		M		P	M		L		M			
Action	Address	MZ	CA	W	DX	D	BS	S	W	FS	C	MA	B	AX	BX	CS
$R_{31} \leftarrow CC \wedge 11000$	BLZ0	01	04	1	1F	0	00	0	0	8	0	10	1	00	00	11
$MC \leftarrow MC + 1$ (NOP)	BLZ1	01	00	0	00	0	00	0	0	0	0	00	0	00	00	00
if (R_{31}=0) $MC \leftarrow BLZ5$ else $MC \leftarrow MC + 1$	BLZ2	11	BLZ5	0	00	0	00	0	0	0	0	00	0	1F	00	00
$MC \leftarrow MC + 1$ (NOP)	BLZ3	01	00	0	00	0	00	0	0	0	0	00	0	00	00	00
$PC \leftarrow PC_{-1}$ + se IM	BLZ4	01	00	0	00	0	11	0	0	0	0	01	1	00	00	01
$MC \leftarrow IDLE$	BLZ5	00	IDLE	0	00	0	00	0	0	0	0	00	0	00	00	00

(c) Compare Less Than

				R		M		P	M		L		M			
Action	Address	MZ	CA	W	DX	D	BS	S	W	FS	C	MA	B	AX	BX	CS
$R[SA] - R[SB]$, $CC \leftarrow L \Vert Z \Vert N \Vert C \Vert V$	CGT0	01	00	0	00	0	00	0	0	5	1	00	0	00	00	00
$MC \leftarrow MC + 1$ (NOP)	CGT1	01	00	0	00	0	00	0	0	0	0	00	0	00	00	00
$R_{31} \leftarrow CC \wedge 10000$	CGT2	01	18	1	1F	0	00	0	0	8	0	10	1	00	00	11
$MC \leftarrow MC + 1$ (NOP)	CGT3	01	00	0	00	0	00	0	0	0	0	00	0	00	00	00
if (R_{31}≠0) $MC \leftarrow CGT7$ else $MC \leftarrow MC + 1$	CGT4	11	CGT7	0	00	0	00	1	0	0	0	00	0	1F	00	00
$MC \leftarrow MC + 1$ (NOP)	CGT5	01	00	0	00	0	00	0	0	0	0	00	0	00	00	00
$PC \leftarrow PC_{-1}$ + se IM_S	CGT6	01	00	0	00	0	11	0	0	0	0	01	1	00	00	10
$MC \leftarrow IDLE$	CGT7	10	IDLE	0	00	0	00	0	0	0	0	00	0	00	00	00

11-21. (Errata: (a) Assume DR = SA. (b)Assume SB = SA.CHECK)

(a) Push

				R		M		P	M		L		M			
Action	Address	MZ	CA	W	DX	D	BS	S	W	FS	C	MA	B	AX	BX	CS
$R[DR] \leftarrow R[SA] + 1$	PUSH0	01	01	1	01	0	00	0	0	2	0	00	1	00	00	11
$MC \leftarrow MC + 1$ (NOP)	PUSH1	01	00	0	00	0	00	0	0	0	0	00	0	00	00	00
$M[R[SA]] \leftarrow R[SB]$	PUSH2	01	00	0	01	0	00	0	1	0	0	00	0	00	00	00
$MC \leftarrow IDLE$	PUSH3	00	IDLE	0	00	0	00	0	0	0	0	00	0	00	00	00

(b) Pop

				R		M		P	M		L		M			
Action	Address	MZ	CA	W	DX	D	BS	S	W	FS	C	MA	B	AX	BX	CS
$R[DR] \leftarrow M[R[SA]]$	POP0	01	00	1	01	1	00	0	0	0	0	00	0	00	00	00
$R[SB] \leftarrow R[SA] - 1$	POP1	01	01	1	00	0	00	0	0	5	0	00	1	00	00	11
$MC \leftarrow IDLE$	POP2	00	IDLE	0	00	0	00	0	0	0	0	00	0	00	00	00

11-22.*

(a) Add with carry

				R		M		P	M		L		M			
Action	Address	MZ	CA	W	DX	D	BS	S	W	FS	C	MA	B	AX	BX	CS
$R_{31} \leftarrow CC \wedge 00010$	AWC0	01	02	1	1F	0	00	0	0	8	0	10	1	00	00	11
$R_{16} \leftarrow R[SA] + R[SB]$	AWC1	01	00	1	10	0	00	0	0	2	0	00	0	00	00	00
if (R_{31}=0) $MC \leftarrow AWC5$ else $MC \leftarrow MC + 1$	AWC2	11	AWC5	0	00	0	00	0	0	0	0	00	0	1F	00	00
$MC \leftarrow MC + 1$ (NOP)	AWC3	01	00	0	00	0	00	0	0	0	0	00	0	00	00	00
$R[DR] \leftarrow R_{16} + 1$	AWC4	01	01	1	01	0	00	0	0	2	0	00	1	10	00	11
$MC \leftarrow IDLE$	AWC5	00	IDLE	0	00	0	00	0	0	0	0	00	0	00	00	00

(a) Subtract with borrow

				R		M		P	M		L		M			
Action	Address	MZ	CA	W	DX	D	BS	S	W	FS	C	MA	B	AX	BX	CS
$R_{31} \leftarrow CC \wedge 00010$	SWB0	01	02	1	1F	0	00	0	0	8	0	10	1	00	00	11
$R_{16} \leftarrow R[SA] – R[SB]$	SWB1	01	00	1	10	0	00	0	0	5	0	00	0	00	00	00
if (R_{31}≠0) $MC \leftarrow SWB5$ else $MC \leftarrow MC + 1$	SWB2	11	SWB5	0	00	0	00	1	0	0	0	00	0	1F	00	00
$MC \leftarrow MC + 1$ (NOP)	SWB3	01	00	0	00	0	00	0	0	0	0	00	0	00	00	00
$R[DR] \leftarrow R_{16} - 1$	SWB4	01	01	1	01	0	00	0	0	5	0	00	1	10	00	11
$MC \leftarrow IDLE$	SWB5	00	IDLE	0	00	0	00	0	0	0	0	00	0	00	00	00

11-23.

(a) Add Memory Indirect

				R		M		P	M		L		M			
Action	Address	MZ	CA	W	DX	D	BS	S	W	FS	C	MA	B	AX	BX	CS
$R_{16} \leftarrow M[R[SB]]$	AMI0	01	00	1	10	1	00	0	0	0	0	00	0	00	00	00
$MC \leftarrow MC + 1$ (NOP)	AMI1	01	00	0	00	0	00	0	0	0	0	00	0	00	00	00
$R_{16} \leftarrow M[R_{16}]$	AMI2	01	00	1	10	1	00	0	0	0	0	00	0	10	00	00
$MC \leftarrow MC + 1$ (NOP)	AMI3	01	00	0	00	0	00	0	0	0	0	00	0	00	00	00
$R[DR] \leftarrow R[SA] + R_{16}$	AMI4	01	00	1	01	0	00	0	0	2	0	00	0	00	10	00
$MC \leftarrow IDLE$	AMI5	00	IDLE	0	00	0	00	0	0	0	0	00	0	00	00	00

(b) Add to memory

				R		M		P	M		L		M			
Action	Address	MZ	CA	W	DX	D	BS	S	W	FS	C	MA	B	AX	BX	CS
$R_{16} \leftarrow M[R[SA]]$	ATM0	01	00	1	10	1	00	0	0	0	0	00	0	00	00	00
$MC \leftarrow MC + 1$ (NOP)	ATM1	01	00	0	00	0	00	0	0	0	0	00	0	00	00	00
$R_{16} \leftarrow R_{16} + R[SB]$	ATM2	01	00	1	10	0	00	0	0	2	0	00	0	10	00	00
$MC \leftarrow MC + 1$ (NOP)	ATM3	01	00	0	00	0	00	0	0	0	0	00	0	00	00	00
$M[R[DR]] \leftarrow R_{16}$	ATM4	01	00	0	01	0	00	0	1	0	0	00	0	10	00	00
$MC \leftarrow IDLE$	ATM5	00	IDLE	0	00	0	00	0	0	0	0	00	0	00	00	00

11-24.*

Memory Scalar Add (Assume R[SB] > 0 to simplify coding)

				R		M		P	M		L		M			
Action	Address	MZ	CA	W	DX	D	BS	S	W	FS	C	MA	B	AX	BX	CS
$R_{16} \leftarrow R[SB]$	MSA0	01	00	1	10	0	00	0	0	0	0	00	0	00	00	00
$R_{18} \leftarrow R_0$	MSA1	01	00	1	12	0	00	0	0	0	0	00	0	00	00	00
$R_{16} \leftarrow R_{16} - 1$	MSA2	01	01	1	10	0	00	0	0	5	0	00	1	10	00	11
$MC \leftarrow MC + 1$ (NOP)	MSA3	01	00	0	00	0	00	0	0	0	0	00	0	00	00	00
$R_{17} \leftarrow R[SA] + R_{16}$	MSA4	01	00	1	11	0	00	0	0	2	0	00	0	00	10	00
$MC \leftarrow MC + 1$ (NOP)	MSA5	01	00	0	00	0	00	0	0	0	0	00	0	00	00	00
if ($R_{16} \neq 0$) $MC \leftarrow MSA2$ else $MC \leftarrow MC + 1$	MSA6	11	MSA2	0	00	0	00	0	0	0	0	00	0	10	00	00
$R_{18} \leftarrow M[R_{17}] + R_{18}$	MSA7	01	00	1	12	1	00	0	0	0	0	00	0	11	12	00
$R[DR] \leftarrow R_{17}$	MSA8	01	00	1	01	0	00	0	0	0	0	00	0	11	00	00
$MC \leftarrow IDLE$	MSA9	00	IDLE	0	00	0	00	0	0	0	0	00	0	00	00	00

11-25.

Memory Vector Add (Assume R[SB] > 0 to simplify coding)

				R		M		P	M		L		M			
Action	Address	MZ	CA	W	DX	D	BS	S	W	FS	C	MA	B	AX	BX	CS
$R_{16} \leftarrow R[SB]$	MVA0	01	00	1	10	0	00	0	0	0	0	00	0	00	00	00
$MC \leftarrow MC + 1$ (NOP)	MVA1	01	00	0	00	0	00	0	0	0	0	00	0	00	00	00
$R_{16} \leftarrow R_{16} - 1$	MVA2	01	01	1	10	0	00	0	0	5	0	00	1	10	00	11
$MC \leftarrow MC + 1$ (NOP)	MVA3	01	00	0	00	0	00	0	0	0	0	00	0	00	00	00
$R_{17} \leftarrow R[SA] + R_{16}$	MVA4	01	00	1	11	0	00	0	0	2	0	00	0	00	10	00
$R_{18} \leftarrow R_{16} + R[DR]$	MVA5	01	00	1	12	0	00	0	0	2	0	00	0	10	00	00
$R_{19} \leftarrow M[R_{17}]$	MVA6	01	00	1	13	1	00	0	0	0	0	00	0	11	00	00
$R_{20} \leftarrow M[R_{18}]$	MVA7	01	00	1	14	1	00	0	0	0	0	00	0	12	00	00
$MC \leftarrow MC + 1$ (NOP)	MVA8	01	00	0	00	0	00	0	0	0	0	00	0	00	00	00
$R_{21} \leftarrow R_{19} + R_{20}$	MVA9	01	00	1	15	0	00	0	0	2	0	00	0	13	14	00
if (R_{16}≠0) $MC \leftarrow MVA2$ else $MC \leftarrow MC + 1$	MVA10	11	MVA2	0	00	0	00	0	0	0	0	00	0	10	00	00
$M[R_{18}] \leftarrow R_{21}$	MVA11	01	00	0	01	0	00	0	1	0	0	00	0	12	15	00
$MC \leftarrow IDLE$	MSA12	00	IDLE	0	00	0	00	0	0	0	0	00	0	00	00	00

11-26.

(a)

$$\begin{aligned}R[DR] \leftarrow (&R[SA][31:24] + R[SB][31:24],\\ &R[SA][23:16] + R[SB][23:16],\\ &R[SA][15:8] + R[SB][15:8],\\ &R[SA][7:0] + R[SB][7:0])\end{aligned}$$

(b) The function unit requires an additional Add operation in which the carries entering bits 0, 8, 16, and 24 are set to 0.

All potential condition codes produced by the operation, including carries from bits 7, 15, 23 and 31 are ignored.

11-27.

(a) For 16-bit words, the operation can produce a 128-bit result containing 128/16 = 8 minimum words.

(b) For each SPE, there are 128/8 = 16 average bytes produced. Using the eight SPEs, 8 x 16 = 1286 average bytes can be produced.