Digital Electronics and Microprocessor Systems (ELEC211)

Dave McIntosh and **Valerio Selis**

dmc@liverpool.ac.uk
V.Selis@liverpool.ac.uk

UNIVERSITY OF LIVERPOOL

Week 4 – Lecture 10
Microprocessor Systems

UNIVERSITY OF
LIVERPOOL

Question

What values are held in r4, r5 and r6 after the execution of the following assuming that the value in r0 is 0xE0300A0B?

r0:=1110 0000 0011 0000 0000 1010 0000 1011_2

```
LSLS r4, r0, #8     ; 2 hex digits left
ASRS r5, r0, #16    ; 4 hex digits right
LSRS r6, r0, #12    ; 3 hex digits right
```

Go to

www.menti.com

and use the code

XX XX XX

UNIVERSITY OF LIVERPOOL

Answer

LSLS r4, r0, #8

r0 held:

0xE0300A0B =

$1110\ 0000\ 0011\ 0000\ 0000\ 1010\ 0000\ 1011_2$

after the instruction is executed r4 would hold:

$0011\ 0000\ 0000\ 1010\ 0000\ 1011\ \mathbf{0000\ 0000}_2$

So the value in r4 is 0x300A0B00.

Answer

ASRS r5, r0, #16

r0 held:

0xE0300A0B =

1110 0000 0011 0000 0000 1010 0000 1011_2

after the instruction is executed r5 would hold:

1111 1111 1111 1111 1110 0000 0011 0000_2

So the value in r5 is 0xFFFFE030.

Answer

LSRS r6, r0, #12

r0 held:

0xE0300A0B =

$1110\ 0000\ 0011\ 0000\ 0000\ 1010\ 0000\ 1011_2$

after the instruction is executed r6 would hold:

$0000\ 0000\ 0000\ 1110\ 0000\ 0011\ 0000\ 0000_2$

So the value in r6 is 0x000E0300.

Question

What values are held in r3 and r4 after the execution of the following assuming that the value in r0 is 0x00000005 (= 5_{10})?

```
LSLS r3, r0, #1
LSLS r4, r0, #6
```

Go to
www.menti.com
and use the code
xx xx xx

UNIVERSITY OF LIVERPOOL

Answer

LSLS r3, r0, #1

r0 held 0x00000005 which is:

$0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0101_2$

shift left by 1 bit so that the value in r3 is 0x0000000A

$0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 1010_2$

$= 10_{10} = (2 \times 5)$, where $2 = 2^1$

LSLS r4, r0, #6

shift left by 6 bits so that the value in r4 is 0x00000140

$0000\ 0000\ 0000\ 0000\ 0000\ 0001\ 0100\ 0000_2$

$= 320_{10} = (64 \times 5)$, where $64 = 2^6$

Question

Assuming that the values in r5 are:

(i) 0x0000001C (= 28_{10})

$0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0001\ 1100_2$

and (ii) 0xFFFFFFE4 (= -28_{10})

$1111\ 1111\ 1111\ 1111\ 1111\ 1111\ 1110\ 0100_2$

What values are held in r7 after the execution of the following instruction?

ASRS r7, r5, #2

Go to
www.menti.com
and use the code
xx xx xx

UNIVERSITY OF LIVERPOOL

Answer

ASRS r7, r5, #2

If value in r5 is 0x0000001C:

$0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0001\ 1100_2$

then value in r7 is

$0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0111_2$ =

0x00000007 = $(28_{10} \div 4)$, where $4 = 2^2$

If value in r5 is 0xFFFFFFE4:

$1111\ 1111\ 1111\ 1111\ 1111\ 1111\ 1110\ 0100_2$

then value in r7 is

$1111\ 1111\ 1111\ 1111\ 1111\ 1111\ 1111\ 1001_2$ =

0xFFFFFFF9 = $-7_{10} = (-28_{10} \div 4)$, where $4 = 2^2$

Week 4 – Lecture 11
Microprocessor Systems

UNIVERSITY OF
LIVERPOOL

Question

What value is stored in the link register, r14, when the ADDS instruction is executed?

Address	Instruction
0x00000FC2	MOV r1, #0
0x00000FC4	BL subX
0x00000FC8	MOV r0, #1
0x00000FCA	ADDS r1, r0, #1
0x00000FCC	BL subX
0x00000FD0	SUBS r3, r2, #2

Subroutine

Address	Instruction
subX	BICS r6, r1
0x00008108	ANDS r2, r1
0x0000810A	EORS r5, r6
0x0000810C	BX lr

Go to

www.menti.com

and use the code

XX XX XX

UNIVERSITY OF LIVERPOOL

Answer

Address	Instruction
0x00000FC2	MOV r1, #0
0x00000FC4	**BL subX**
0x00000FC8	MOV r0, #1
0x00000FCA	**ADDS** r1, r0, #1
0x00000FCC	**BL subX**
0x00000FD0	SUBS r3, r2, #2

Subroutine

Address	Instruction
subX	BICS r6, r1
0x00008108	ANDS r2, r1
0x0000810A	EORS r5, r6
0x0000810C	BX lr

- The ‘subX’ subroutine is called twice (**BL subX**)
- The **ADDS** instruction is after the first pass through the subroutine but before the second pass.
- So the **link register**, r14, will hold the return address for the first pass, which is **0x00000FC8**.

Question

What value is stored in the link register, r14, when the SUBS instruction is executed?

Address	Instruction
0x00000FC2	MOV r1, #0
0x00000FC4	BL subX
0x00000FC8	MOV r0, #1
0x00000FCA	ADDS r1, r0, #1
0x00000FCC	BL subX
0x00000FD0	SUBS r3, r2, #2

Subroutine

Address	Instruction
subX	BICS r6, r1
0x00008108	ANDS r2, r1
0x0000810A	EORS r5, r6
0x0000810C	BX lr

Go to

www.menti.com

and use the code

XX XX XX

UNIVERSITY OF LIVERPOOL

Answer

Address	Instruction
0x00000FC2	MOV r1, #0
0x00000FC4	**BL subX**
0x00000FC8	MOV r0, #1
0x00000FCA	ADDS r1, r0, #1
0x00000FCC	**BL subX**
0x00000FD0	**SUBS** r3, r2, #2

Subroutine

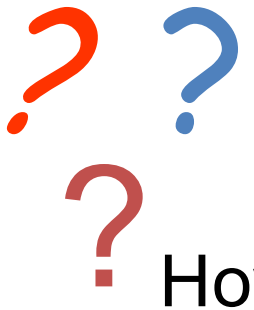
Address	Instruction
subX	BICS r6, r1
0x00008108	ANDS r2, r1
0x0000810A	EORS r5, r6
0x0000810C	BX lr

- The ‘subX’ subroutine is called twice (**BL subX**)
- The **SUBS** instruction is after the second pass through the subroutine.
- So the **link register**, r14, will hold the return address for the second pass, which is **0x00000FD0**.

UNIVERSITY OF LIVERPOOL

Week 4 – Lecture 12
Microprocessor Systems

UNIVERSITY OF
LIVERPOOL



Question

How many clock cycles do the following instructions take to execute if the value in r0 is (a) not equal to 0 and (b) equal to 0? What is the CPI?

```
        MOVS r0, r0
        BNE cont            ; branch if not zero
        ADDS r3, r4, r5
        EORS r2, r1, r4
cont    SUBS r6, r7, r2
```

Go to

www.menti.com

and use the code

XX XX XX

UNIVERSITY OF LIVERPOOL

Answer

(a) r0:= value not equal to 0

- The branch is taken so that needs 3 clock cycles.
- The MOVS and SUBS instructions take 1 each
- so 5 clock cycles are required in total.
- 5 clock cycles and 3 instructions execute = 1.66 CPI

(b) r0:= value equal to 0

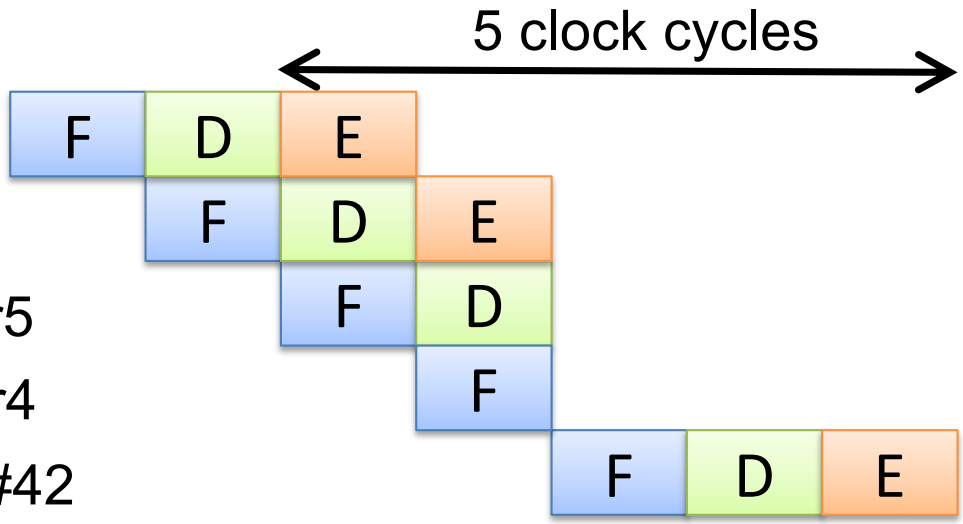
- The branch is not taken so that only needs one clock cycle. The other four instructions (MOVS, ADDS, EORS and SUBS) each take 1 so again 5 clock cycles are needed.
- 5 clock cycles and 5 instructions execute = 1 CPI

Answer

(a) r0:= value not equal to 0

The first instruction clears the zero flag (Z=0) so that the condition, NE, for the branch is met. The program counter is reloaded with the address of the SUBS instruction. The ADDS and EORS instructions are not executed.

Address	Instruction
start	MOVS r0, r0
start + 2	BNE cont
start + 4	ADDS r3, r4, r5
start + 6	EORS r2, r1, r4
cont	SUBS r6, r7, #42



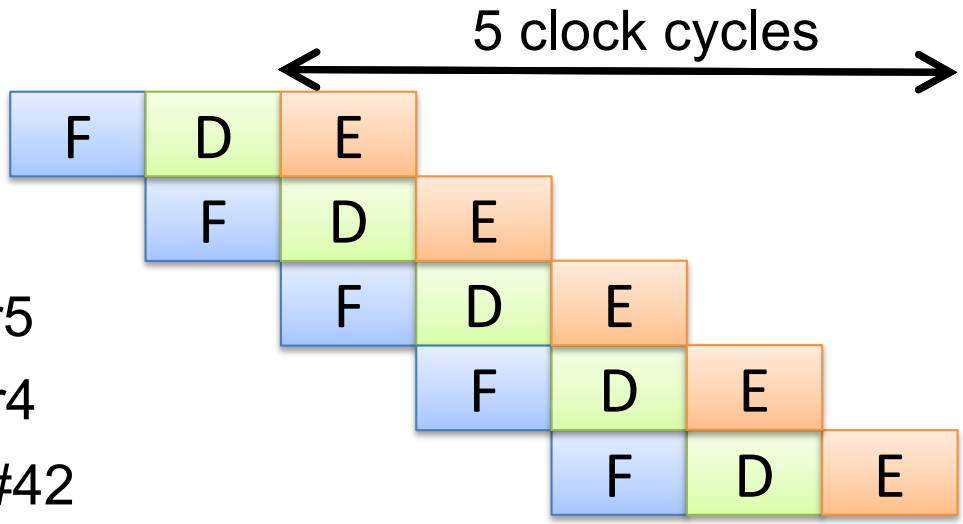
5/3 = 1.66 CPI

Answer

(b) r0:= value equal to 0

The first instruction sets the zero flag (Z=1) so that the condition, NE, for the branch is not met. The program counter continues to increment by 2 so that the ADDS, EORS and SUBS instructions are executed in that order.

Address	Instruction
start	MOVS r0, r0
start + 2	BNE cont
start + 4	ADDS r3, r4, r5
start + 6	EORS r2, r1, r4
cont	SUBS r6, r7, #42



5/5 = 1 CPI (Optimum)

UNIVERSITY OF LIVERPOOL