Jose Franco Baquera

April 1, 2018

CS – 273

Homework #8

1. Hand assemble the following program: You must indicate the low and high byte at each program memory location using the correct endianness:

YL = R28
YH = R29
ZL = R30
ZH = R31
XL = R26
XH = R27

```
.ORG 0
   LDI R19,0x5 ;R19 = 5 (R19 for counter)
   LDI R16,0x55 ;load R16 with value 0x55 (value to be copied)
   LDI YL,0x40 ;load the low byte of Y with value 0x40
   LDI YH,0x1 ;load the high byte of Y with value 0x1
L1: ST Y, R16 ;copy R16 to memory location 0x140
   INC YL ;increment the low byte of Y
   DEC R19 ;decrement the counter
   BRNE L1 ;loop until counter = zero
```

Answer:

Assembly Code	Address	Low Byte	High Byte	"Little Endian"
.ORG 0	N/A	N/A	N/A	N/A
LDI R19, 0x5	0x0000	0011-0101 (or 0x35)	1110-0000 (or 0xE0)	0x35E0

LDI R16, 0x55	0x0001	0000-0101 (or 0x05)	1110-0101 (or 0xE5)	0x05E5
LDI YL, 0x40	0x0002	1100-0000 (or 0xC0)	1110-0100 (or 0xE4)	0xC0E4
LDI YH,0x1	0x0003	1101-0001 (or 0xD1)	1110-0000 (or 0xE0)	0xD1E0
L1: ST Y, R16	0x0004	0000-1000 (or 0x08)	1000-0011 (or 0x83)	0x0883
INC YL	0x0005	1100-0011 (or 0xC3)	1001-0101 (0x95)	0xC395
DEC R19	0x0006	0011-1010 (or 0x3A)	1001-0101 (or 0x95)	0x3A95
BRNE L1	0x0007	1110-0001 (or 0xE1)	1111-0111 (or 0xF7)	0xE1F7

2. Hand assemble the following program: You must indicate the low and high byte at each program memory location using the correct endianess:

```
.ORG 0 ;burn into ROM starting at 0
    LDI ZL, lo8(MYDATA) ;R30 = 00 low-byte addr
    LDI ZH, hi8(MYDATA) ;R31 = 05, high-byte addr
    LDI XL, lo8(0x140) ;R26 = 40, low-byte RAM address
    LDI XH, hi8(0x140) ;R27 = 1, high-byte RAM address
AGAIN:
    LPM R16, Z+ ;read the table, then increment Z
    CPI R16, 0 ;compare R16 with 0
    BREQ END ;exit if end of string
    ST X+, R16 ;store R16 in RAM and inc X
```

```
        RJMP AGAIN
    END: RJMP END
         .ORG 0x500 ;data burned starting at 0x500
MYDATA: .asciz “The Promise of World Peace”
```

Answer:

Assembly Code	**Address**	**Low Byte**	**High Byte**	**“Little Endian”**
.ORG 0	N/A	N/A	N/A	N/A
LDI ZL, lo8(MYDATA)	0x0000	1110-0000 (or 0xE0)	1110-0000 (or 0xE0)	0xE0E0
LDI ZH, hi8(MYDATA)	0x0001	1111-0101 (or 0xF5)	1110-0000 (or 0xE0)	0xF5E0
LDI XL, lo8(0x140)	0x0002	1010-0000 (or 0xA0)	1110-0100 (or 0xE4)	0xA0E4
LDI XH, hi8(0x140)	0x0003	1011-0001 (or 0xB1)	1110-0000 (or 0xE0)	0xB1E0
AGAIN: (NOTE: *AGAIN* IS ASSOCIATED WITH THE NEXT INSTRUCTION)	N/A	N/A	N/A	N/A
LPM R16, Z+	0x0004	0000-0101 (or 0x05)	1001-0001 (or 0x91)	0x0591
CPI R16, 0	0x0005	0000-0000 (or 0x00)	0011-0000 (0x30)	0x0030
BREQ END	0x0006	0001-0001 (or 0x11)	1111-0000 (or 0xF0)	0x11F0

ST X+, R16	0x0007	0000-1101 (or 0x0D)	1001-0011 (or 0x93)	0x0D93
RJMP AGAIN	0x0008	1111-1011 (or 0xFB)	1100-1111 (or 0xCF)	0xFBCF
END: RJMP END	0x0009	1111-1111 (or 0xFF)	1100-1111 (or 0xCF)	0xFFCF
.ORG 0x500	N/A	N/A	N/A	N/A
MYDATA: .asciz “The Promise of World Peace”	**‘T’ = 0x54** (NOTE: This is stored in memory address 0x0500) **‘h’ = 0x68** (NOTE: This is stored in memory address 0x0501) **‘e’ = 0x65** (NOTE: This is stored in memory address 0x0502) **‘ ’ = 0x20** (NOTE: This is stored in memory address 0x0503) **‘P’ = 0x50** (NOTE: This is stored in memory address 0x0504) **‘r’ = 0x72** (NOTE: This is stored in memory address 0x0505) **‘o’ = 0x6F** (NOTE: This is stored in memory address 0x0506) **‘m’ = 0x6D** (NOTE: This is stored in memory address 0x0507) **‘i’ = 0x69** (NOTE: This is stored in memory address 0x0508) **‘s’ = 0x73** (NOTE: This is stored in memory address 0x0509) **‘e’ = 0x65** (NOTE: This is stored in memory address 0x050A) **‘ ’ = 0x20** (NOTE: This is stored in memory address 0x050B) **‘o’ = 0x6F** (NOTE: This is stored in memory address 0x050C) **‘f’ = 0x66** (NOTE: This is stored in memory address 0x050D) **‘ ’ = 0x20** (NOTE: This is stored in memory address 0x050E) **‘W’ = 0x57** (NOTE: This is stored in memory address 0x050F) **‘o’ = 0x6F** (NOTE: This is stored in memory address 0x0510) **‘r’ = 0x72** (NOTE: This is stored in memory address 0x0511) **‘l’ = 0x6C** (NOTE: This is stored in memory address 0x0512) **‘d’ = 0x64** (NOTE: This is stored in memory address 0x0513)			

	‘ ’ = 0x20 (NOTE: This is stored in memory address 0x0514) **‘P’ = 0x50** (NOTE: This is stored in memory address 0x0515) **‘e’ = 0x65** (NOTE: This is stored in memory address 0x0516) **‘a’ = 0x61** (NOTE: This is stored in memory address 0x0517) **‘c’ = 0x63** (NOTE: This is stored in memory address 0x0518) **‘e’ = 0x65** (NOTE: This is stored in memory address 0x0519) **NUL = 0x00** (NOTE: This is stored in memory address 0x051A) (The zero byte was added at the end of the string since the “.asciz is just like .ascii, but each string is followed by a zero byte”.)