

Microprocessor and Computer Architecture (MPCA) Laboratory
UE20CS252 4th Semester,
Academic Year 2021-22

Week 6

Vanshika Goel	PES1UG20CS484	H Section
----------------------	----------------------	------------------

1. Display hexadecimal digits [0–9, A–F] on the 7 segment display.

.DATA

ZERO: .BYTE 0b11101101
ONE: .BYTE 0b01100000
TWO: .BYTE 0b11001110
THREE: .BYTE 0b11101010
FOUR: .BYTE 0b01100011
FIVE: .BYTE 0b10101011
SIX: .BYTE 0b10101111
SEVEN: .BYTE 0b11100000
EIGHT: .BYTE 0b11101111
NINE: .BYTE 0b11101011
A: .BYTE 0b11100111
B: .BYTE 0b11101111
C: .BYTE 0b10001101
D: .BYTE 0b11101101
E: .BYTE 0b10001111
F: .BYTE 0b10000111

.TEXT

MOV R0,#0

ALWAYS:

SWI 0x202
CMP R0,#1
BEQ FORWARD
CMP R0,#2
BEQ BACKWARD
B ALWAYS

FORWARD:

MOV R3,#16
MOV R2,#1
LDR R1,=ZERO
LOOP1:
LDRB R0,[R1]

```

SWI 0x200
BL DELAY
ADD R1,R1,R2
SUB R3,R3,#1
CMP R3,#0
BNE LOOP1
B ALWAYS

```

BACKWARD:

```

MOV R3,#16
MOV R2,#-1
LDR R1,=F
LOOP2:
    LDRB R0,[R1]
    SWI 0x200
    BL DELAY
    ADD R1,R1,R2
    SUB R3,R3,#1
    CMP R3,#0
    BNE LOOP2
B ALWAYS

```

DELAY:

```

MOV R4,#8000
DELAYCOUNT:
SUB R4,R4,#1
CMP r4,#0
BGE DELAYCOUNT
MOV PC,LR

```

The screenshot displays the cpulab01xz.net emulator interface. The top menu bar includes options like 'Step Into', 'Step Over', 'Step Out', 'Continue', 'Stop', 'Restart', and 'Reload'. The main window is divided into several sections:

- Registers:** A list of registers (r0-r15, sp, lr, cpsr, spsr) with their current values. The 'pc' register is highlighted at 00000000.
- Disassembly (Ctrl-D):** A table showing the assembly code being executed. The table has columns for Address, Opcode, and Disassembly. The current instruction at address 00000000 is 'mov r0, #0 ; 0x0'.
- Messages:** A log at the bottom showing compilation and linking messages. It indicates that the code and data were loaded into memory and that the compilation was successful.

The disassembly table shows the following instructions:

Address	Opcode	Disassembly
ffffff00	aaaaaaa	bge 0xfeaaaa0
ffffff04	aaaaaaa	bge 0xfeaaaa4
ffffff08	aaaaaaa	bge 0xfeaaaa8
ffffff0c	aaaaaaa	bge 0xfeaaaac
00000000	e3a00000	21: mov r0, #0 ; 0x0
00000004	ef000202	22: ALWAYS: svc #514 ; 0x202
00000008	e3500001	23: cmp r0, #1 ; 0x1
0000000c	0a000002	24: BEQ FORWARD beq 0x1c (0x1c: FORWARD)
00000010	e3500002	25: cmp r0, #2 ; 0x2
00000014	0a00000b	26: BEQ BACKWARD beq 0x48 (0x48: BACKWARD)
00000018	ea000000	27: B ALWAYS b 0x4 (0x4: ALWAYS)
0000001c	e3a03010	28: FORWARD: mov r3, #16 ; 0x10
00000020	e3a02001	29: mov r2, #1 ; 0x1
00000024	e59f105c	30: LDR R1,=ZERO ldr r1, [pc, #92] ; 0x88
00000028	e5100000	31: LOOP1: ldr r1, [pc, #16]

