LAB 4-C

7-SEGMENT

OBJECTIVES:

- To interface seven-segment to the AVR simulator.

MATERIAL:

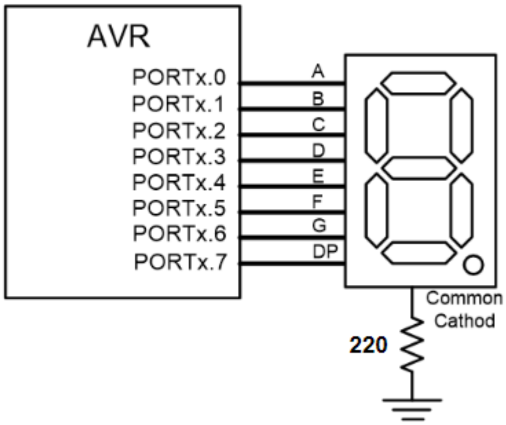
- Atmel Studio or the assembler of your choice.
- https://lcgamboa.github.io/js/picsimlab.html?../picsimlab_examples/ (Simulator)

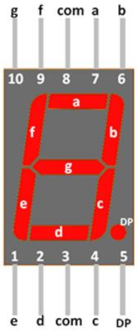
WEB SITES:

- www.microchip.com for Atmel Studio Software

ACTIVITY 1

a) Connect a common cathode 7-segment directly to PORTD.



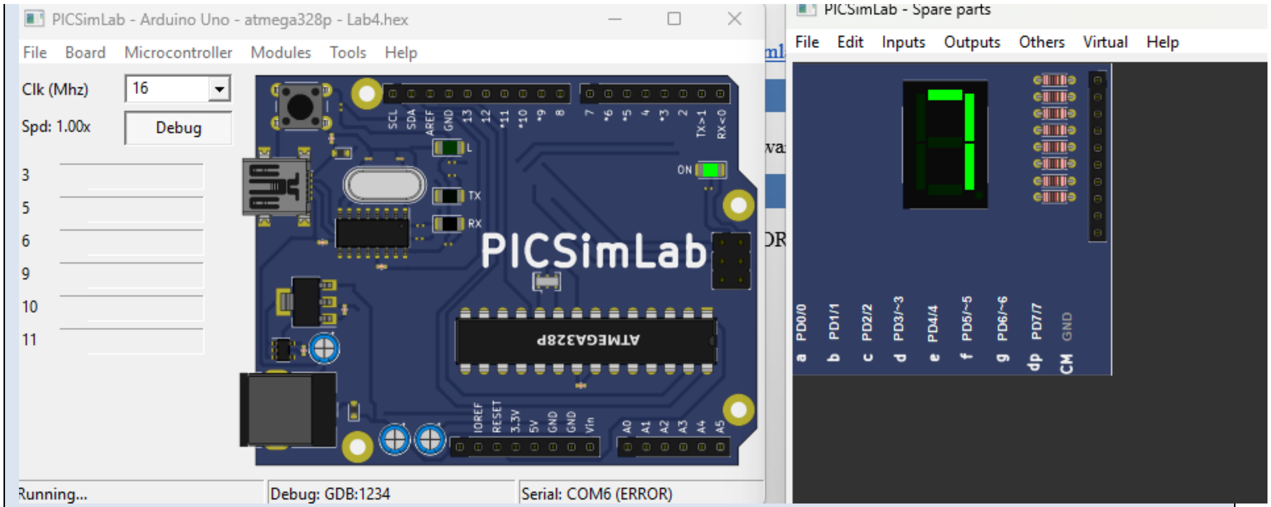


b) Write the following program in the AVR Studio, build and download to the picsimlab.

```
         LDI     R20, 0xFF
         OUT     DDRD, R20
         LDI     R20, 0b00000111
         OUT     PORTD, R20
HERE: RJMP      HERE
```

LAB 4-C

7-SEGMENT



PICSimLab - Arduino Uno - atmega328p - Lab4.hex

Clk (Mhz): 16

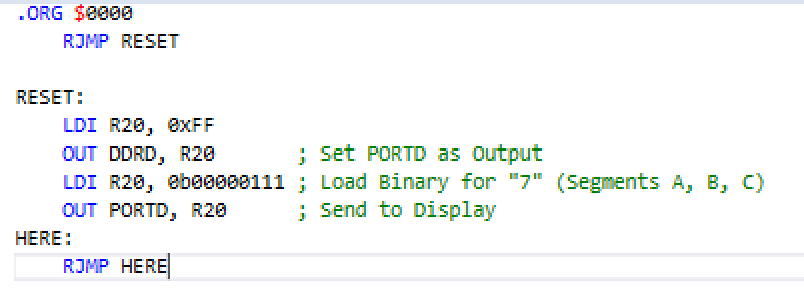
Spd: 1.00x

Running...

Debug: GDB:1234

Serial: COM6 (ERROR)

PICSimLab - Spare parts



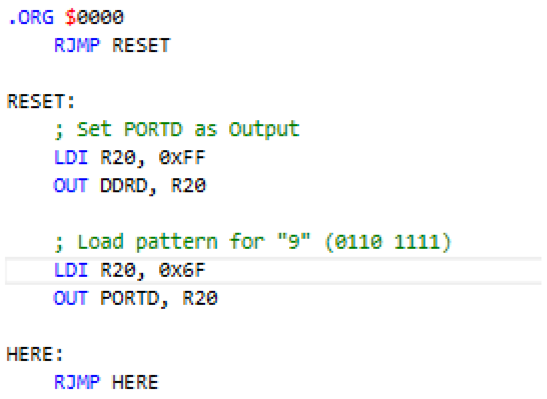
```
.ORG $0000
    RJMP RESET

RESET:
    LDI R20, 0xFF
    OUT DDRD, R20       ; Set PORTD as Output
    LDI R20, 0b00000111 ; Load Binary for "7" (Segments A, B, C)
    OUT PORTD, R20      ; Send to Display
HERE:
    RJMP HERE
```

Lab 4-C

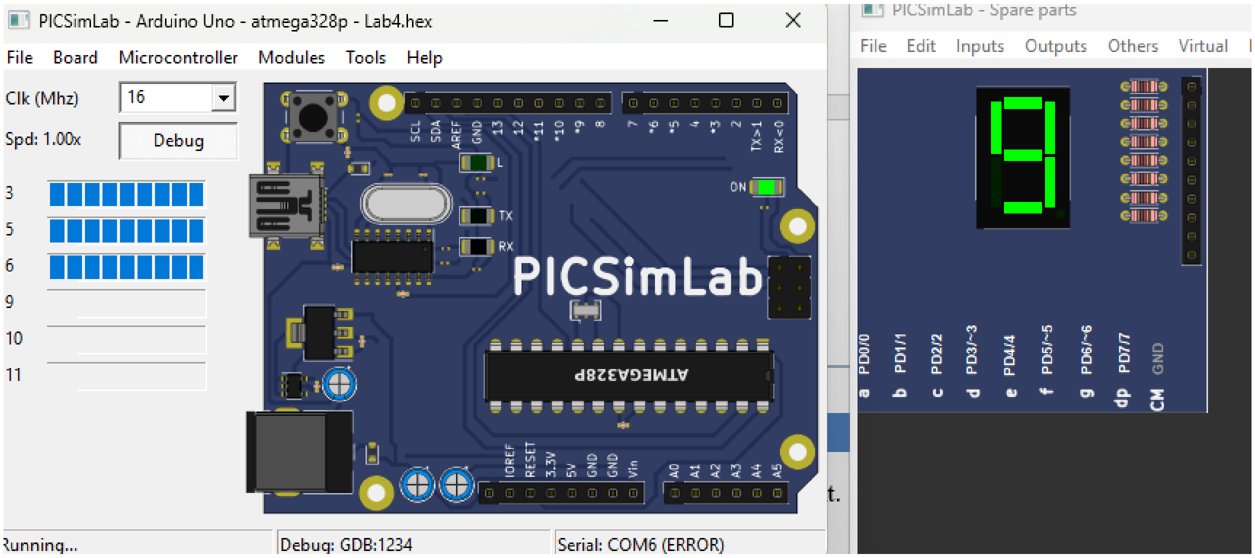
7-Segment

ACTIVITY 2

Modify the previous program to display 9 on the 7-segment.



```
.ORG $0000
    RJMP RESET

RESET:
    ; Set PORTD as Output
    LDI R20, 0xFF
    OUT DDRD, R20

    ; Load pattern for "9" (0110 1111)
    LDI R20, 0x6F
    OUT PORTD, R20

HERE:
    RJMP HERE
```



PICSimLab - Arduino Uno - atmega328p - Lab4.hex

Clk (Mhz) 16

Spd: 1.00x

Running...

Debug: GDB:1234

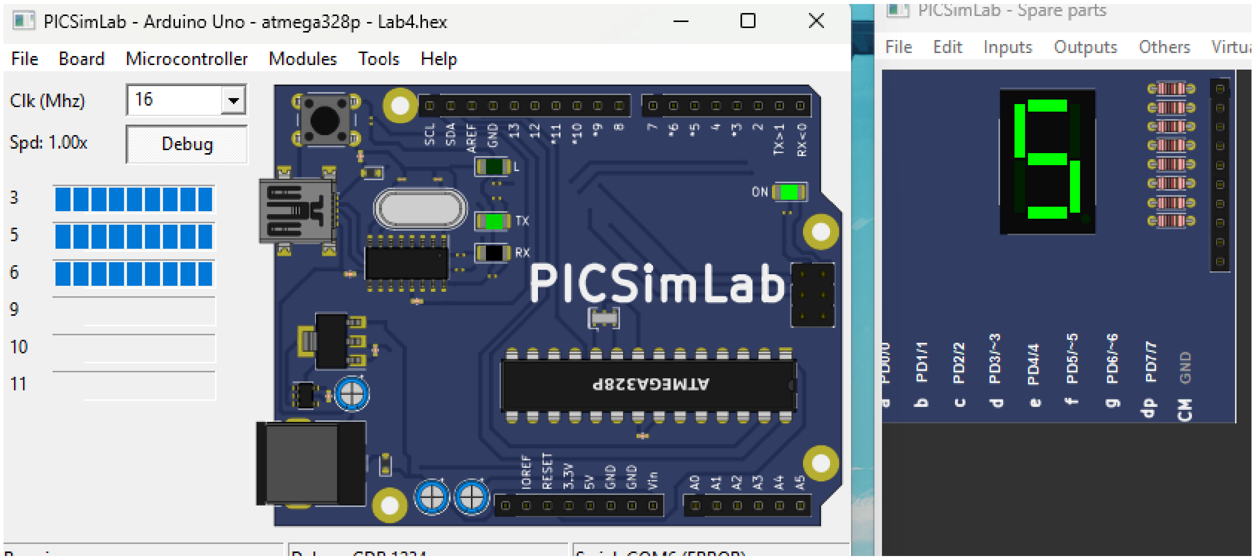
Serial: COM6 (ERROR)

LAB 4-C

7-SEGMENT

ACTIVITY 3

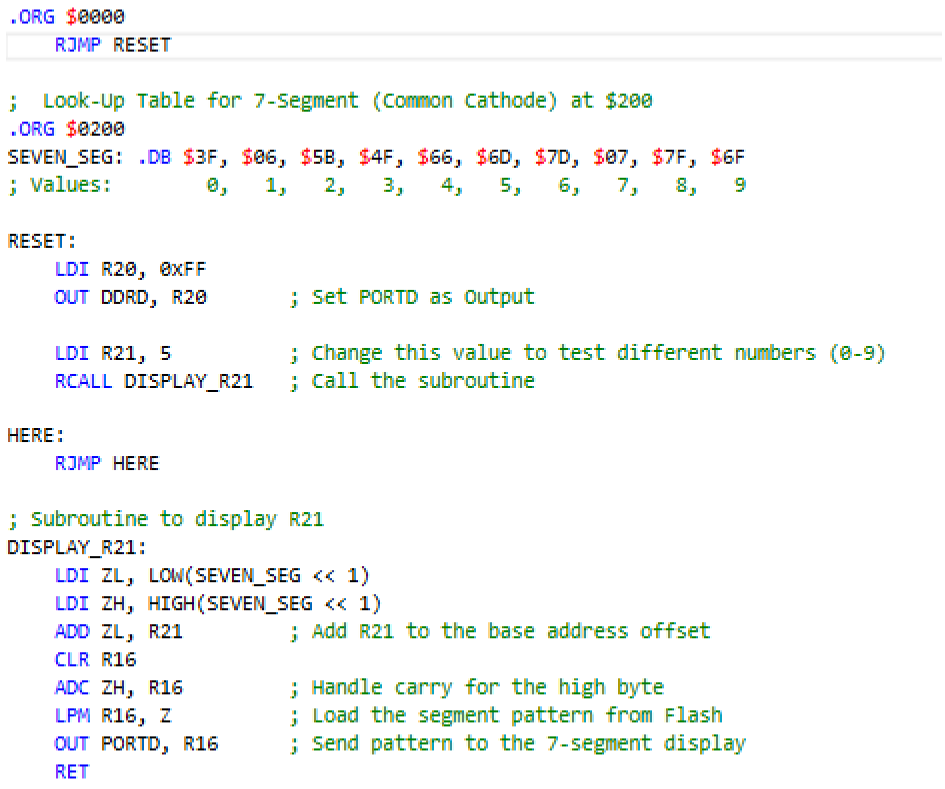
Use a **look-up table** to write a subroutine that displays the value stored in R21 on the 7-segment.



PICSimLab - Arduino Uno - atmega328p - Lab4.hex

Clk (Mhz): 16

Spd: 1.00x



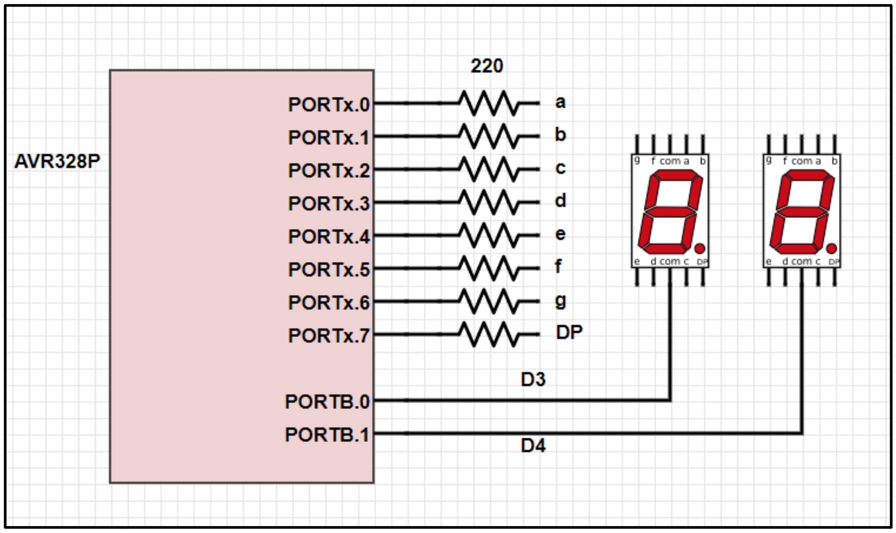
```
.ORG $0000
    RJMP RESET

;  Look-Up Table for 7-Segment (Common Cathode) at $200
.ORG $0200
SEVEN_SEG: .DB $3F, $06, $5B, $4F, $66, $6D, $7D, $07, $7F, $6F
; Values:        0,   1,   2,   3,   4,   5,   6,   7,   8,   9

RESET:
    LDI R20, 0xFF
    OUT DDRD, R20        ; Set PORTD as Output

    LDI R21, 5           ; Change this value to test different numbers (0-9)
    RCALL DISPLAY_R21    ; Call the subroutine

HERE:
    RJMP HERE

; Subroutine to display R21
DISPLAY_R21:
    LDI ZL, LOW(SEVEN_SEG << 1)
    LDI ZH, HIGH(SEVEN_SEG << 1)
    ADD ZL, R21          ; Add R21 to the base address offset
    CLR R16
    ADC ZH, R16          ; Handle carry for the high byte
    LPM R16, Z           ; Load the segment pattern from Flash
    OUT PORTD, R16       ; Send pattern to the 7-segment display
    RET
```

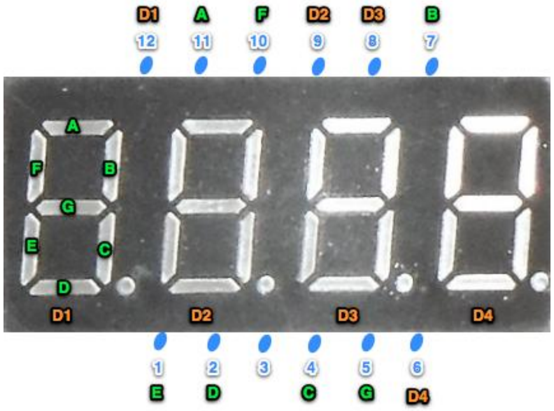
LAB 4-C

7-SEGMENT

ACTIVITY 4

a) Connect two 7-segments to the same port of AVR, as shown below.

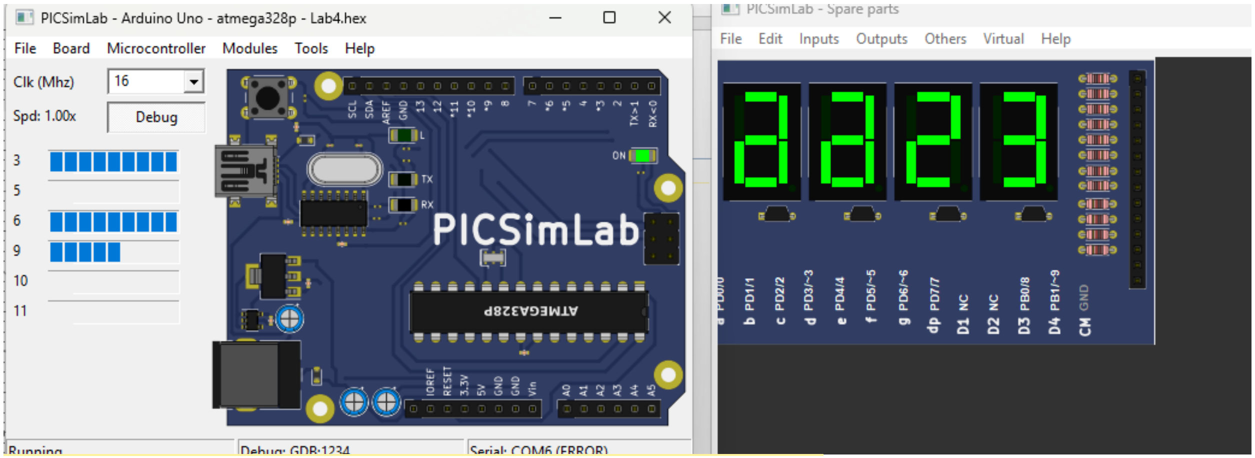


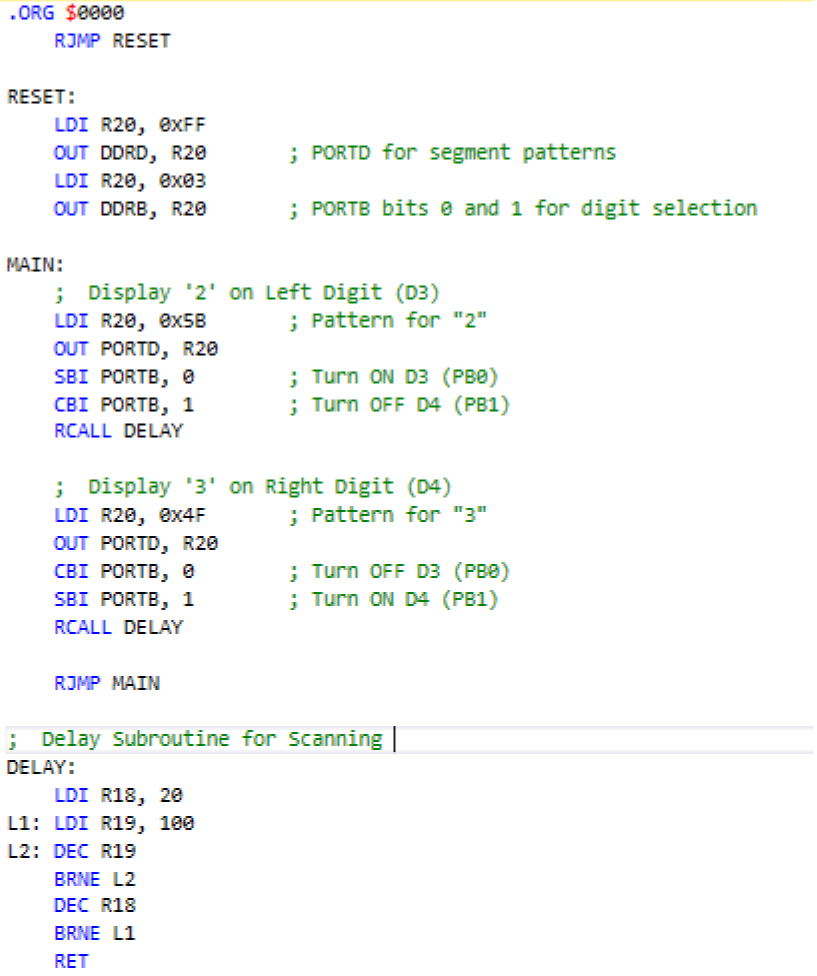


b) Display 23 by scanning on those two 7-segments.

Lab 4-C

7-Segment





```
.ORG $0000
    RJMP RESET

RESET:
    LDI R20, 0xFF
    OUT DDRD, R20       ; PORTD for segment patterns
    LDI R20, 0x03
    OUT DDRB, R20       ; PORTB bits 0 and 1 for digit selection

MAIN:
    ;  Display '2' on Left Digit (D3)
    LDI R20, 0x5B       ; Pattern for "2"
    OUT PORTD, R20
    SBI PORTB, 0        ; Turn ON D3 (PB0)
    CBI PORTB, 1        ; Turn OFF D4 (PB1)
    RCALL DELAY

    ;  Display '3' on Right Digit (D4)
    LDI R20, 0x4F       ; Pattern for "3"
    OUT PORTD, R20
    CBI PORTB, 0        ; Turn OFF D3 (PB0)
    SBI PORTB, 1        ; Turn ON D4 (PB1)
    RCALL DELAY

    RJMP MAIN

;  Delay Subroutine for Scanning
DELAY:
    LDI R18, 20
L1: LDI R19, 100
L2: DEC R19
    BRNE L2
    DEC R18
    BRNE L1
    RET
```