# objective:

* Execute instructions to create delays using subroutines.
* Perform communication with shift registers.

# references:

* Lab manual chapter 1-2.

# EXPERIMENT 1:

1. Use the following program:

| .include "m324PAdef.inc"  .org 00  ldi r16,0x01  out DDRA, r16  start:  sbi PORTA,PINA0  cbi PORTA, PINA0  rjmp start |
| --- |

Connect PA0 to a measurement channel on the TEST STATION and measure pulse forms using an oscilloscope.

# EXPERIMENT 2:

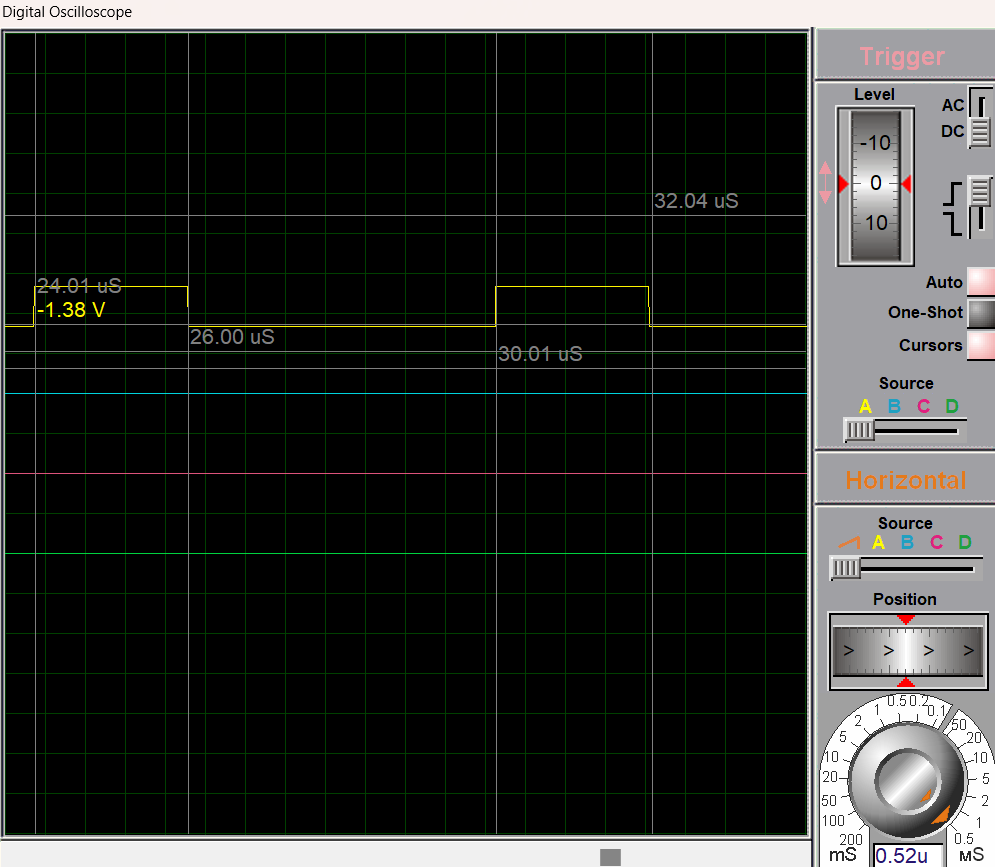
1. Write a subroutine Delay1ms and use it to write a program to generate a 1KHz square wave on PA0.
2. Use this subroutine to write subroutines Delay10ms, Delay100ms, and Delay1s.
3. Use the Delay1s subroutine to write a program to blink/turn off an LED connected to PA0.

# EXPERIMENT 3:

1. Connect the necessary signals from an AVR port to the control signals of the shift register on header J13. Connect the output of the shift register to a LED bar.
2. Using the sample programs from the experiment guide, write a program to create a gradually lit LED effect from left to right, then gradually turn them off from left to right after every 500ms.

# EXPERIMENT 1:

1. Answer the following questions:
   1. Capture a pulse waveform on PA0.



* 1. What is the frequency, duration of the high signal, and duration of the low signal?

By using a simulated oscilloscope, we can determine the clock period (T) is 6 microseconds (6 μs), we can calculate the frequency and durations more accurately.

f= 1/T = 166.67 kHz

Duration of the high signal: 26μs -24μs=2μs

Duration of the low signal: 30μs -26μs=4μs

* 1. Explain the measured results.

The high and low signals are set by the sbi instruction, and it is cleared by the cbi instruction. These results can be attributed to the execution time of the instructions in the microcontroller and the time it takes to change the state of the output pin. The specific timings can vary depending on the microcontroller's architecture and the compiler used to generate the code.

# EXPERIMENT 2:

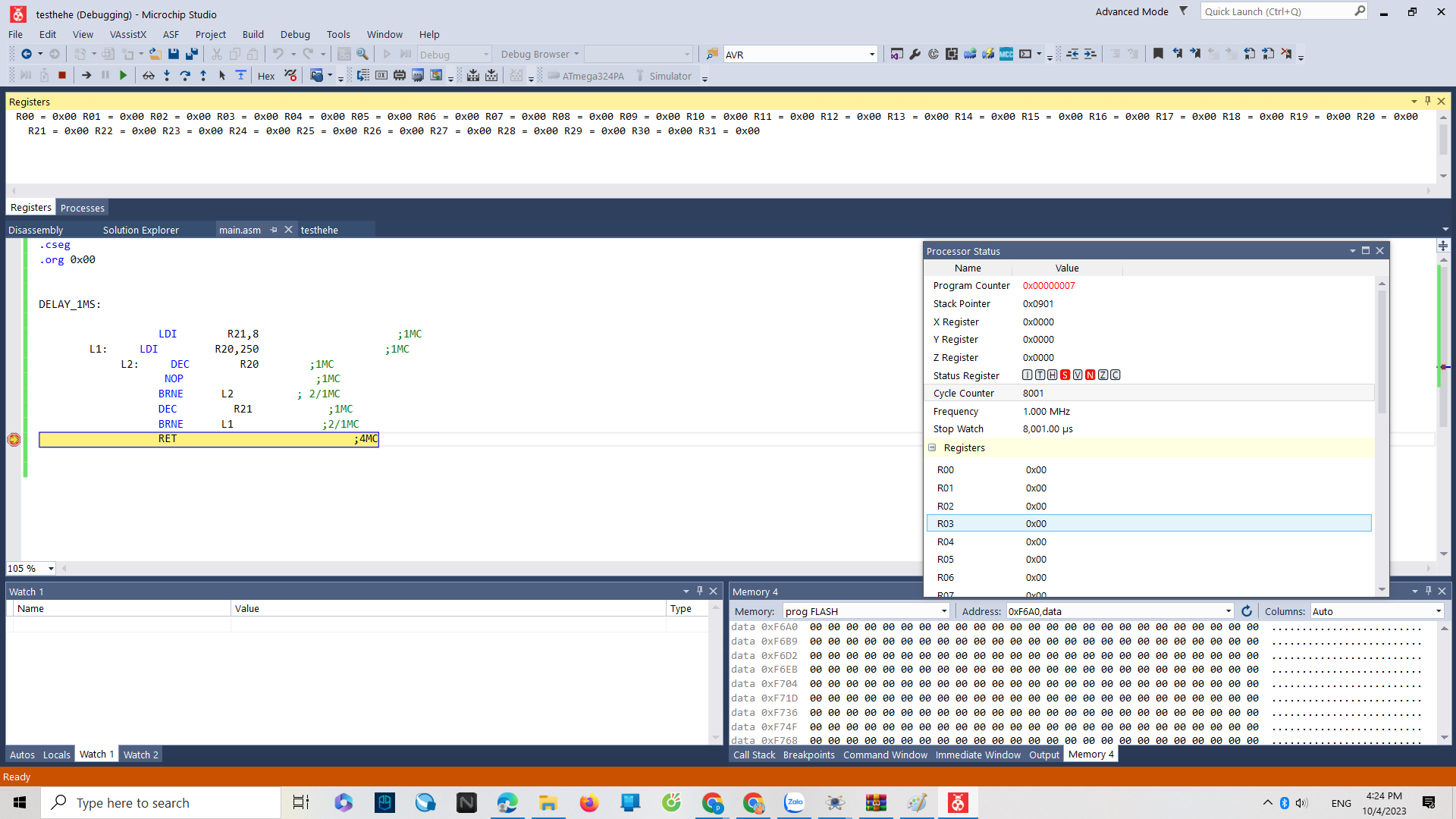
1. Answer the following questions:
   1. How to calculate the number of machine cycles needed to execute the Delay1ms subroutine. Present a simulation image.

| .cseg  .org 0x00  START:  SBI DDRA, 0  MAIN:  CBI PORTA,0  CALL DELAY\_1MS  SBI PORTA,0  CALL DELAY\_1MS  JMP MAIN  DELAY\_1MS:  LDI R21,8 ;1MC  L1: LDI R20,250 ;1MC  L2: DEC R20 ;1MC  NOP ;1MC  BRNE L2 ; 2/1MC  DEC R21 ;1MC  BRNE L1 ;2/1MC  RET ;4MC |
| --- |

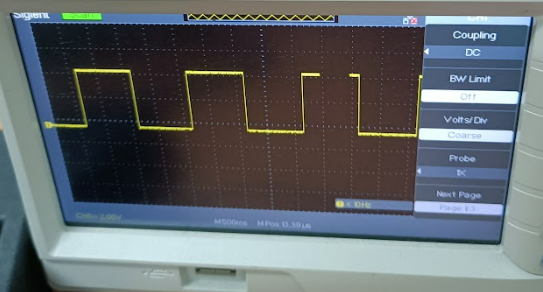
The above is an example of creating a 50Hz frequency signal on pin PA0. The subprogram

DELAY\_1MS generates a 1ms delay with an 8MHz clock, where 1MC (Machine Cycle)

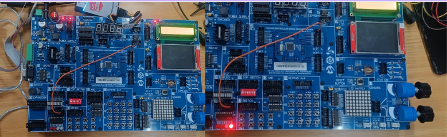
equals 0.125μs, corresponding to 8000MC x 0.125 = 1ms.



* 1. Image of a 1KHz square wave on PA0.



1. Source code for 2.c with comments.



| .ORG 0x00  LOOP:  SBI PORTA, 0  RCALL DL\_1SEC ;Call the delay function for 1s  CBI PORTA, 0  RCALL DL\_1SEC  RJMP LOOP  DL\_1SEC:  LDI R18, 10  C: RCALL DL\_100MSEC ;Call the delay function for 100ms  DEC R18  BRNE C  RET  DL\_100MSEC:  LDI R17, 10  B: RCALL DL\_10MSEC ;;Call the delay function for 10ms  DEC R17  BRNE B  RET  DL\_10MSEC:  LDI R16, 10  A: RCALL DL\_1MSEC  DEC R16  BRNE A  RET  DL\_1MSEC:  LDI R21, 80  LP2: LDI R20, 250  LP1: NOP  DEC R20  BRNE LP1  DEC R21  BRNE LP2  RET |
| --- |

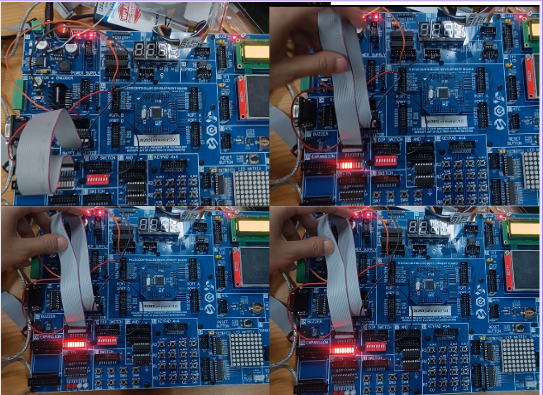
# EXPERIMENT 3:

1. Answer the following questions:
   1. Describe the connections on the experimental kit.

| Choosing JTAG, using 8x8 BUS to connect port J36 from Shift Register directly to **BARLED**  Connect PB0 to DSI  Connect PB1 to Latch  Connect PB2 to CLK  Connect PB3 to nCLR |
| --- |
|  |

* 1. According to the datasheet of 74HC595, what is the highest clock frequency it can operate at?
  2. How do you expand the display to 16 LEDs?

1. Source code with comments.



| .include "m324padef.inc" ; Include Atmega324pa definitions  .def shiftData = r20 ; Define the shift data register  .equ clearSignalPort = PORTB ; Set clear signal port to PORTB  .equ clearSignalPin = 3 ; Set clear signal pin to pin 0 of PORTB  .equ shiftClockPort = PORTB ; Set shift clock port to PORTB  .equ shiftClockPin = 2 ; Set shift clock pin to pin 1 of PORTB  .equ latchPort = PORTB ; Set latch port to PORTB  .equ latchPin = 1 ; Set latch pin to pin 0 of PORTB  .equ shiftDataPort = PORTB ; Set shift data port to PORTB  .equ shiftDataPin = 0 ; Set shift data pin to pin 3 of PORTB  main:  call initport  ldi shiftData,0x55  call cleardata  call shiftoutdata  stop:  jmp stop  ; Initialize ports as outputs  initport:  ldi r24,(1<<clearSignalPin)|(1<<shiftClockPin)|(1<<latchPin)|(1<<shiftDataPin)  out DDRB, r24 ; Set DDRB to output  ret  ldi shiftData,0x55  cleardata:  cbi clearSignalPort, clearSignalPin ; Set clear signal pin to low  ; Wait for a short time  sbi clearSignalPort, clearSignalPin ; Set clear signal pin to high  ret  ; Shift out data  shiftoutdata:  cbi shiftClockPort, shiftClockPin ;  ldi r18, 8 ; Shift 8 bits  shiftloop:  sbrc shiftData, 7 ; Check if the MSB of shiftData is 1  sbi shiftDataPort, shiftDataPin ; Set shift data pin to high  sbi shiftClockPort, shiftClockPin ; Set shift clock pin to high  lsl shiftData ; Shift left  cbi shiftClockPort, shiftClockPin ; Set shift clock pin to low  cbi shiftDataPort, shiftDataPin ; Set shift data pin to low  dec r18  brne shiftloop  ; Latch data  sbi latchPort, latchPin ; Set latch pin to high  CALL DELAY\_1S  cbi latchPort, latchPin ; Set latch pin to low  DEC R16  BRNE shiftoutdata  RET  DELAY\_500mS:  LDI R16,250  L2: LDI R17,250  L1: DEC R17  NOP  NOP  NOP  NOP  NOP  BRNE L1  DEC R16  BRNE L2  RET |
| --- |