

**Namal University**

**Mianwali**

**Department of Electrical Engineering**

EE-252L: Introduction to Embedded Systems

**Lab Manual: 11**

**AVR Sensor Interfacing**

|  |  |
| --- | --- |
| **Students Name** |  |
| **Roll Number** |  |
| **Submission Date** |  |
| **Marks Obtained** |  |

**Instructors: Dr. Hamza Zad Gul**

# Objectives

# In this lab, the student will learn about sensor interfacing and motor control programming using atmega328p.

# Course Learning Outcomes

CLO1: Practice the correct use of programming constructs of assembly language

CLO2: Construct systems by interfacing AVR peripherals

CLO3: Perform the assigned task individually/as a team effectively

CLO4: Report the outcomes of task performed effectively in oral and written form

# Software

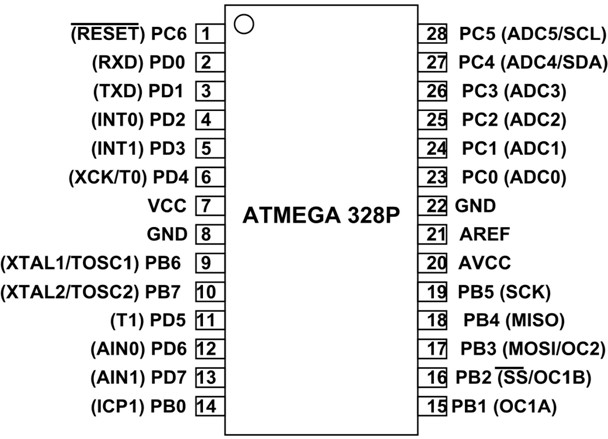
* Microchip studio

# Hardware

* Atmega 328p
* Atmega328p USBasp programmer Board
* Breadboard
* Connecting wires
* LEDs
* Resistors
* Capacitors
* Crystal oscillator
* Push buttons
* Oscilloscope
* IR Sensor
* Servo Motor (SG90)

# Instructions

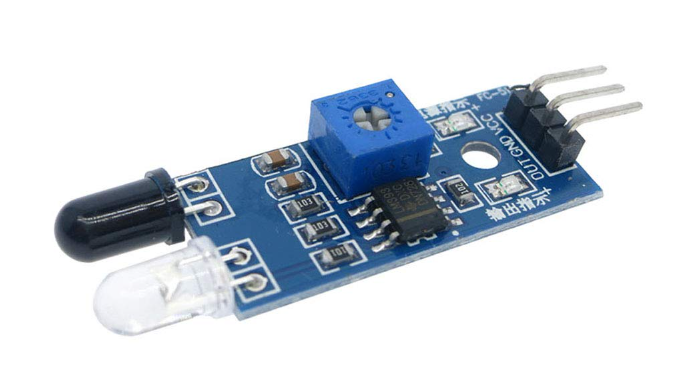
* You must submit the lab report complete within given deadline.
* Plagiarism or any hint thereof will be dealt with strictly. Any incident where plagiarism is caught, both (or all) students involved will be given zero marks, regardless of who copied whom.
* Multiple such incidents will result in disciplinary action being taken.



**Introduction:**

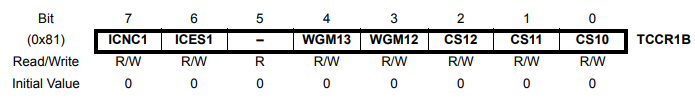
**Infrared Sesnor:**

* IR communication uses IR (Infrared) waves from the electromagnetic spectrum.
* IR waves are the waves in the frequency range of 300 GHz to 430 THz and having wavelengths in the range of around 700 nm to 1mm.
* Communication between remote and television is an example of IR communication.
* An IR LED is used to transmit data wirelessly in digital form (0 – LED OFF or 1 – LED ON).
* An IR photodiode or IR phototransistor receives this data. The IR photodiode or IR phototransistor gives different current values according to the intensity of light.
* As transmitted data is in digital form (LED ON or OFF), a microcontroller can be used to decode this data.
* It is possible to modulate the data transmitted and there are special decoder IR receivers available that can receive the modulated data.

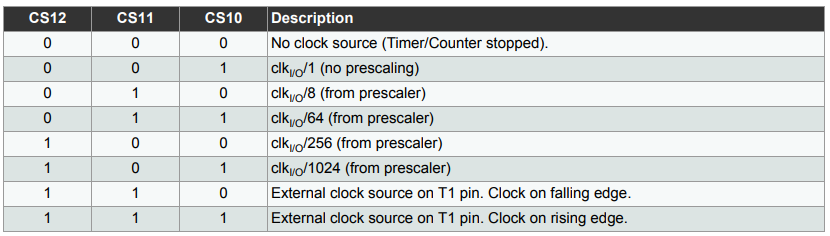


1. **TCCR1B**

Another decision one has to make before using input capture is whether to use noise canceller feature and whether to use rising edge or falling edge detection. This is configured using the ICNC1 and ICES1 bits located in the TCCR1B register as shown below.

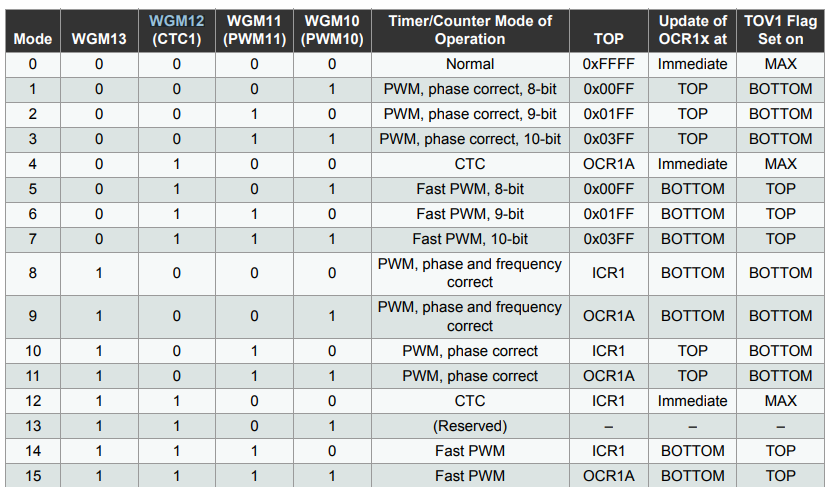


When ICNC1 bit is set High then the noise canceller is enabled. Setting ICES1 to High means capturing in rising edge and when ICES1 is Low it means capturing on falling edge.

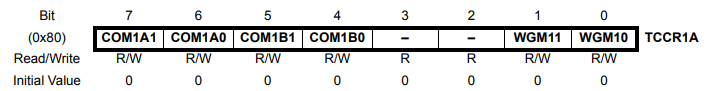


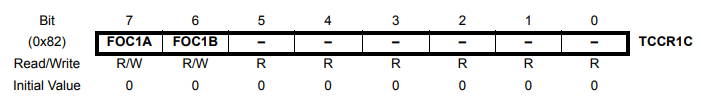
1. **Timer1 Modes:**

For using input capture we configure the Timer 1 is normal mode which is mode 0 where all the WGM bits are 0. The following is the table for timer 1.



Note that the WGM11 and WGM10 bits are located in the TCCR1A register.



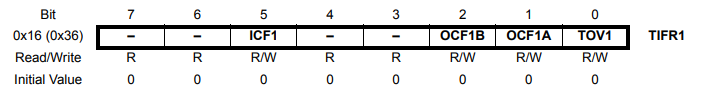


1. **About Clock Select bits**

The clock select bits (CS12, CS11, CS10) which are located in TCCR1B (provided above) register configures the pre-scalar value. This can be chosen according to user need. It effects the size of the byte of the measured result.

1. **Flags**

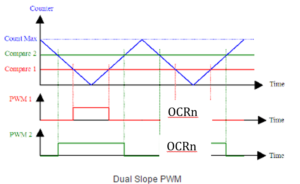
The last thing we need to know about while using the input capture is the flag associated with the input capture feature. Once logic level transition is detected, the ICF1 flag bit located in the Timer/Counter1 Interrupt Flag Register (TIFR1) is set. This flag can be monitored to know that capture event has occurred and take necessary action. The TIFR1 is shown below.



The other way to know that event capture is to enable the input capture event interrupt. This is done by enabling the ICIE1 bit in the TIMSK1 register(Timer Interrupt Mask Register 1) which is shown below.

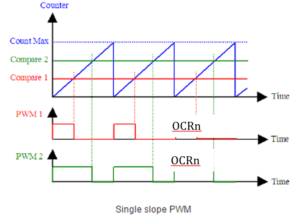
## 

Now, there are two main operating modes of PWM: phase-correct PWM and fast PWM. To briefly explain the difference between the two, in phase-correct PWM mode the timer counter counts up and down (generating a triangle wave). When the counter reaches the compare bit, the PWM signal will be high, and when the counter counts back down below the compare bit the PWM signal will be low. Characteristics of phase-correct PWM are that the pulses are wider and that the middle of OCRnA and OCRnB are in alignment (where OCRnA and OCRnB are two separate PWM pins connected to the same timer counter) as shown in Figure.

[](https://www.arxterra.com/wp-content/uploads/2018/05/arduino_119.png)

*Phase-correct PWM*

In fast PWM mode, the timer counter counts up and then drastically drops down to zero again (generating a sawtooth wave). When the counter reaches the compare bit, the PWM signal will be high, and when the counter drops back down, the PWM signal will be low. Characteristics of fast PWM are that it is twice as fast, the pulses are narrower, and that the rising edges of OCRnA and OCRnB are in alignment as shown in Figure below.

[](https://www.arxterra.com/wp-content/uploads/2018/05/arduino_120.png)

*Fast PWM*

## I: Write the following code in microchip studio and implement the circuit in proteus. Also implement this circuit on hardware

## Write comments in front of each line of code

#define *F\_CPU* 16000000UL

#include <avr/io.h>

#include <util/delay.h>

#define LCD\_Port PORTD

#define LCD\_DPin DDRD

#define RSPIN PD0

#define ENPIN PD1

void LCD\_Action(unsigned char cmnd)

{

LCD\_Port = (LCD\_Port & 0x0F) | (cmnd & 0xF0);

LCD\_Port &= ~ (1<<RSPIN);

LCD\_Port |= (1<<ENPIN);

*\_delay\_us*(1);

LCD\_Port &= ~ (1<<ENPIN);

*\_delay\_us*(200);

LCD\_Port = (LCD\_Port & 0x0F) | (cmnd << 4);

LCD\_Port |= (1<<ENPIN);

*\_delay\_us*(1);

LCD\_Port &= ~ (1<<ENPIN);

*\_delay\_ms*(2);

}

void LCD\_Init (void)

{

LCD\_DPin = 0xFF;

*\_delay\_ms*(15);

LCD\_Action(0x33);

LCD\_Action(0x32);

LCD\_Action(0x28);

LCD\_Action(0x02);

LCD\_Action(0x0c);

LCD\_Action(0x06);

LCD\_Action(0x01);

*\_delay\_ms*(2);

}

void LCD\_Clear()

{

LCD\_Action (0x01);

*\_delay\_ms*(2);

LCD\_Action (0x80);

}

void LCD\_Print (char \*str)

{

int i;

for(i=0; str[i]!=0; i++)

{

LCD\_Port = (LCD\_Port & 0x0F) | (str[i] & 0xF0);

LCD\_Port |= (1<<RSPIN);

LCD\_Port|= (1<<ENPIN);

*\_delay\_us*(1);

LCD\_Port &= ~ (1<<ENPIN);

*\_delay\_us*(200);

LCD\_Port = (LCD\_Port & 0x0F) | (str[i] << 4);

LCD\_Port |= (1<<ENPIN);

*\_delay\_us*(1);

LCD\_Port &= ~ (1<<ENPIN);

*\_delay\_ms*(2);

}

}

void LCD\_Printpos (char row, char pos, char \*str)

{

if (row == 0 && pos<16)

LCD\_Action((pos & 0x0F)|0x80);

else if (row == 1 && pos<16)

LCD\_Action((pos & 0x0F)|0xC0);

LCD\_Print(str);

}

void servo1 ()

{

TCCR1A |=(1<<WGM11)|(1<<COM1A1);

TCCR1B |= (1<<WGM12)|(1<<WGM13)|(1<<CS11);

ICR1 = 39999;

int offset = 800;

{

OCR1A = 1999+ offset;

*\_delay\_ms*(500);

}

}

void servo2 ()

{

TCCR1A |=(1<<WGM11)|(1<<COM1A1);

TCCR1B |= (1<<WGM12)|(1<<WGM13)|(1<<CS11);

ICR1 = 39999;

int offset = 800;

{

OCR1A = 999 - offset;

*\_delay\_ms*(500);

}

}

int main(void)

{

LCD\_Init();

unsigned char z;

DDRC = 0;

while (1)

{

z = PINC;

z = z & 0b00000011;

switch(z)

{

case(0):

{

LCD\_Printpos(0,0,"Welcome To");

LCD\_Printpos(1,0,"Namal Univesity");

*\_delay\_ms*(1000);

LCD\_Clear();

break;

}

case (1):

{

LCD\_Clear();

LCD\_Printpos(0,0,"To Pass Course");

LCD\_Printpos(1,0,"Minimum is 50%");

DDRB |= 1<<PINB1;

servo1();

*\_delay\_ms*(1000);

break;

}

case (2):

{

LCD\_Clear();

LCD\_Printpos(0,0,"Congratulation");

LCD\_Printpos(1,0,"You Passed");

DDRB |= 1<<PINB1;

servo2();

*\_delay\_ms*(1000);

break;

}

case (3):

{

LCD\_Clear();

LCD\_Printpos(0,0,"This is all");

LCD\_Printpos(1,0,"A DREAM");

*\_delay\_ms*(1000);

break;

}

}

}

return 0;

}

# Answer the following questions:

# 1. Implement the circuit on proteus and attach screenshot?

# 2. Attach a picture of the circuit you have implemented on breadboard?

# 3. We have used ICR1 in the code, what is ICR1 and what function does it provide?

# 4. The servo motor has 3 terminal, brown, orange and red. What is each wire used for?

# 5. In your words explain the working of IR sensor?

# 6. When you simulate this code in proteus is there anything that is not working properly?

**Introduction to Embedded System Lab Rubrics**

* **Method of Evaluation** Viva Conducted during lab and lab reports submitted by students

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Assessment tool/ weightage/**  **(CLO, PLO)** | **Excellent**  **(10 - 9)** | **Good**  **(8 – 7)** | **Satisfactory**  **(6 – 4)** | **Unsatisfactory**  **(3 – 1)** | **Poor**  **0** | **Marks Obtained** |
| **Programming**  **(CLO1, PLO5)** | Correct Code. Easy to understand with proper comments | Correct Code but without proper indentation or comments | Slightly incorrect code with proper comments | Incorrect code with improper format and no comments | Code not submitted |  |
| **Circuit Design**  **(CLO2: PLO3)** | Circuit is simulated/implemented correctly without any errors | Circuit is simulated but implemented with minor errors | Circuit is simulated & implementation both have errors | Circuit is simulated & implemented however some components are missing/incorrect value | Circuit is simulated/implemented does not work |  |
| **Individual/ Teamwork**  **(CLO3:PLO9)** | The student/s worked effectively throughout lab to perform the assigned tasks | The student/s performed all the assigned lab tasks however one member took lead | The student/s completed all tasks however failed to work effectively | The student/s attempted all the tasks however the one member did most of the work | The student/s did not work together/at all |  |
| **Lab Report**  **(CLO4:PLO10)** | The student was able to effectively answer all questions regarding performed tasks and report provides all information without mistakes | The student was able to effectively answer all questions regarding performed tasks however the report has minor mistakes | The student was able to answer most questions regarding performed tasks and information in report is not communicated effectively | The student was able to answer some questions regarding performed tasks and report is confusing and misleading | The student was not able to answer questions regarding performed tasks and report information is incorrect/irrelevant |  |
| Total | | | | | |  |