Natthanan Bhukan 61070507206

King Mongkut's University of Technology Thonburi
Faculty of Engineering, Department of Computer Engineering

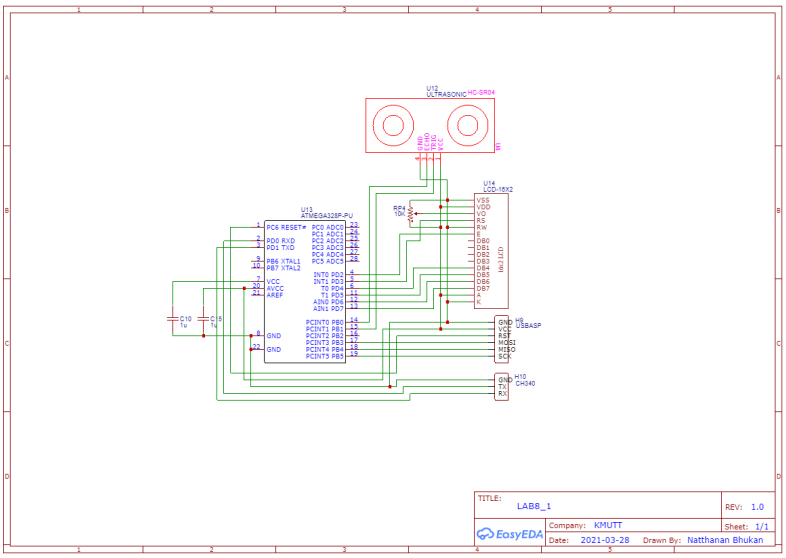
CPE 328 Embedded System, 2/2020

LAB Lecture 8: AVR Timer (Input Capture / PWM)

Assign Date: 24 Mar 2021 Due Date: 30 Mar 2021

On Board

Circuit



Natthanan Bhukan 61070507206

King Mongkut's University of Technology Thonburi

Faculty of Engineering, Department of Computer Engineering

CPE 328 Embedded System, 2/2020

LAB Lecture 8: AVR Timer (Input Capture / PWM)

Assign Date: 24 Mar 2021 Due Date: 30 Mar 2021

Code

```
#define F_CPU 8000000L

#include <avr/io.h>
#include <util/delay.h>
#include <stdio.h>
#include <string.h>
#include <stdlib.h>

#define LCD_Dir  DDRD      /* Define LCD data port direction */
#define LCD_Port PORTD      /* Define LCD data port */
#define RS PD3             /* Define Register Select pin */
#define EN PD2             /* Define Enable signal pin */
#define TRI_PIN PB1

void LCD_Command( unsigned char cmnd )
{
  LCD_Port = (LCD_Port & 0x0F) | (cmnd & 0xF0);  /* sending upper nibble */
  LCD_Port &= ~ (1<<RS);                         /* RS=0, command reg. */
  LCD_Port |= (1<<EN);                          /* Enable pulse */
  _delay_us(1);
  LCD_Port &= ~ (1<<EN);

  _delay_us(200);

  LCD_Port = (LCD_Port & 0x0F) | (cmnd << 4);    /* sending lower nibble */
  LCD_Port |= (1<<EN);
  _delay_us(1);
  LCD_Port &= ~ (1<<EN);
  _delay_ms(2);
}
void LCD_Char( unsigned char data )
{
  LCD_Port = (LCD_Port & 0x0F) | (data & 0xF0);    /* sending upper */
  LCD_Port |= (1<<RS);                           /* RS=1, data reg. */
  LCD_Port|= (1<<EN);
  _delay_us(1);
  LCD_Port &= ~ (1<<EN);

  _delay_us(200);

  LCD_Port = (LCD_Port & 0x0F) | (data << 4);      /* sending lower */
  LCD_Port |= (1<<EN);
  _delay_us(1);
  LCD_Port &= ~ (1<<EN);
  _delay_ms(2);
}

void LCD_Init (void)       /* LCD Initialize function */
{
  LCD_Dir = 0xFF;          /* Make LCD port direction as o/p */
  _delay_ms(20);           /* LCD Power ON delay always >15ms */

  LCD_Command(0x02);         /* send for 4 bit initialization of LCD  */
  LCD_Command(0x28);           /* 2 line, 5*7 matrix in 4-bit mode */
  LCD_Command(0x0c);           /* Display on cursor off*/
  LCD_Command(0x06);           /* Increment cursor (shift cursor to right)*/
  LCD_Command(0x01);           /* Clear display screen*/
  _delay_ms(2);
}
```

Natthanan Bhukan 61070507206

King Mongkut's University of Technology Thonburi

Faculty of Engineering, Department of Computer Engineering

CPE 328 Embedded System, 2/2020

LAB Lecture 8: AVR Timer (Input Capture / PWM)

Assign Date: 24 Mar 2021 Due Date: 30 Mar 2021

```
void LCD_String (char *str)   /* Send string to LCD function */
{
  int i;
  for(i=0; str[i]!=0 && str[i]!=0x000a; i++)    /* Send each char of string till the NULL An
  {
    LCD_Char(str[i]);
  }
}

void LCD_Clear()
{
  LCD_Command (0x01);   /* Clear display */
  _delay_ms(2);
  LCD_Command (0x80);   /* Cursor at home position */
}

void TIMER1_Init() {
    TCCR1B |= (1 << ICNC1) | (1 << CS11);
}
int main(void) {

    LCD_Init();             /* Initialization of LCD*/

    //set trig port to output, low
    DDRB |= (1 << TRI_PIN);
    PORTB &= ~(1 << TRI_PIN);

    TIMER1_Init();

    char output_string[20] = {};
    while (1) {
        //phase 1: start
        //trig high
        PORTB |= (1 << TRI_PIN);
        //delay 10 us
        _delay_us(10);

        //trig low
        PORTB &= ~(1 << TRI_PIN);

        //phase 2: capture wave rising
        //set to capture rising edge
        TCCR1B |= (1 << ICES1);
        //check ICF1 high for capture event
        while (!(TIFR1 & (1 << ICF1)));
        //set TCNT1 to 0
        TCNT1 = 0;
        //reset ICF1 flag
        TIFR1 |= (1 << ICF1);
```

King Mongkut's University of Technology Thonburi
Faculty of Engineering, Department of Computer Engineering

CPE 328 Embedded System, 2/2020

LAB Lecture 8: AVR Timer (Input Capture / PWM)

Assign Date: 24 Mar 2021 Due Date: 30 Mar 2021

```
        //phase 3: capture wave falling
        //set to capture falling edge
        TCCR1B &= ~(1 << ICES1);
        //check ICF1 high for capture event
        while (!(TIFR1 & (1 << ICF1)));
        TIFR1 |= (1 << ICF1);
        //copy 16-bit data out from ICR1
        uint16_t output_data = ICR1;
        //reset ICF1 flag


        //phase 4: convert and output
        //convert
        uint8_t distance = ((output_data / 1000000.0) * 340 * 100) / 2.0;
        sprintf(output_string, "Distance: %d cm\n", distance);
        //output
        LCD_Clear();
        LCD_String(output_string);
        _delay_ms(1000);
    }
}
```

1. ให้นักศึกษา ออกแบบวงจรและเขียนโปรแกรมเพื่ออ่านค่าระยะทางจากเซนเซอร์วัดระยะทางแบบ Ultrasonic รุ่น HC-SR04 และแสดงผลระยะทางในหน่วยซม.ออกทางหน้าจอ LCD

   Result

Natthanan Bhukan 61070507206

King Mongkut's University of Technology Thonburi

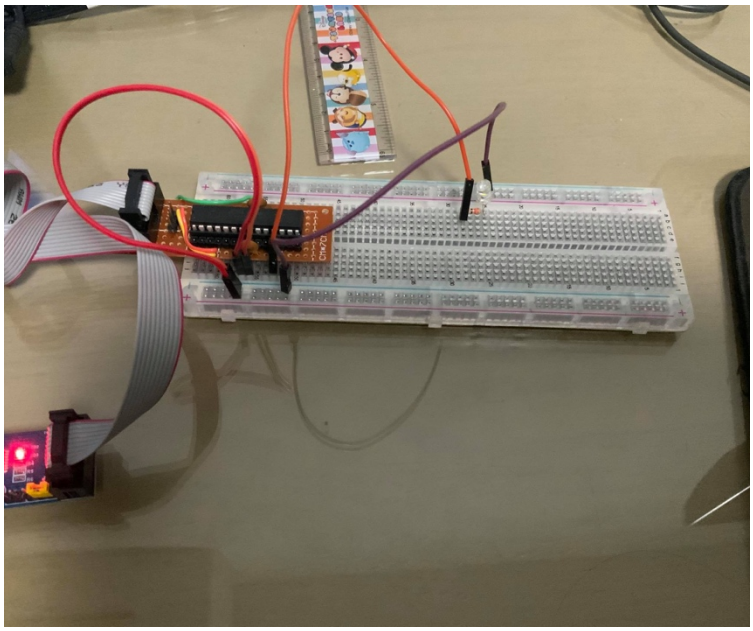
Faculty of Engineering, Department of Computer Engineering

CPE 328 Embedded System, 2/2020

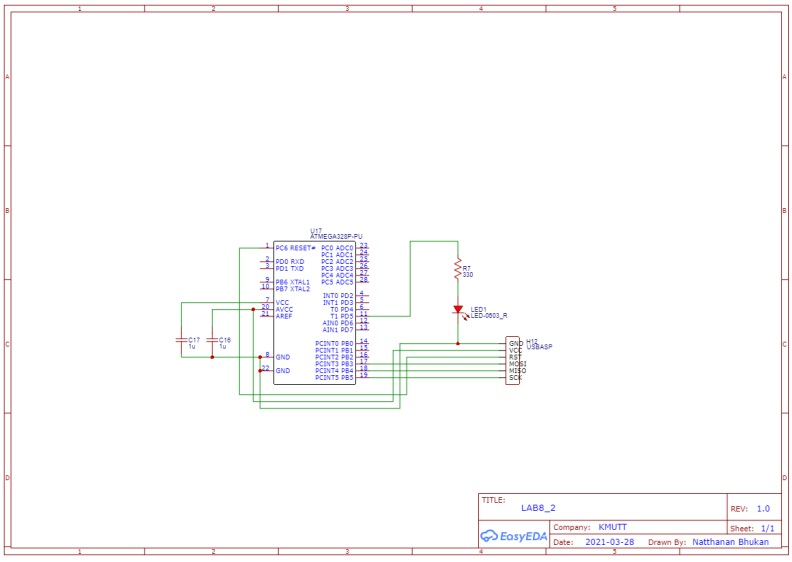
LAB Lecture 8: AVR Timer (Input Capture / PWM)

Assign Date: 24 Mar 2021 Due Date: 30 Mar 2021

On Board



Circuit



King Mongkut's University of Technology Thonburi
Faculty of Engineering, Department of Computer Engineering

CPE 328 Embedded System, 2/2020

LAB Lecture 8: AVR Timer (Input Capture / PWM)

Assign Date: 24 Mar 2021 Due Date: 30 Mar 2021

Code

```
#define F_CPU 8000000L

#include <avr/io.h>
#include <util/delay.h>

#define PMW_PIN PD5

void TIMER0_Init() {
    TCCR0A |= (1 << COM0B1) | (1 << WGM01) | (1 << WGM00);
    TCCR0B |= (1 << CS01) | (1 << CS00);
}

void PWM_LED(uint8_t dim){
    OCR0B = dim;
}

int main(void) {

    DDRD |= (1 << PMW_PIN);
    PORTD &= ~(1 << PMW_PIN);

    TIMER0_Init();
    int PWM_Vector[16] = {0,1,2,3,4,6,8,12,16,23,32,45,64,90,128,180,255};

    while (1) {
        for(int i = 0; i < 16; i++){
            PWM_LED(PWM_Vector[i]);
            _delay_ms(30);
        }
        for(int i = 15; i >= 0; i--){
            PWM_LED(PWM_Vector[i]);
            _delay_ms(30);
        }
    }
}
```

2. ให้นักศึกษา ออกแบบวงจรและเขียนโปรแกรมเพื่อ ปรับความสว่างของหลอดไฟ LED ให้ค่อย ๆ สว่างขึ้นและลงวงช้าไปเรื่อย ๆ

Result