



## โครงการ : เครื่องวัดระยะทาง Distance Meter

รหัสวิชา 03603323 วิชา Embedded Systems

### จัดทำโดย

นาย ปฐมพงษ์ อินตะใจ	รหัสบัณฑิต	5630300474
น.ส. ปภาวรา อาชีวะณิข	รหัสบัณฑิต	5630300482
นาย ไกรวิชญ์ ยาแก้ว	รหัสบัณฑิต	5730300085

### เสนอ

อาจารย์จิรวรรณ จิตประสูติวิทย์

คณะวิศวกรรมศาสตร์ศรีราชา สาขาวิศวกรรมคอมพิวเตอร์และสารสนเทศศาสตร์

มหาวิทยาลัยเกษตรศาสตร์ วิทยาเขตศรีราชา

## 1. ที่มาและจุดประสงค์

ปัจจุบันมีการนำเทคโนโลยีมาใช้อำนวยความสะดวกในชีวิตประจำวันแต่ในที่นี้ เป็นการนำ Sensor ที่วัดความเร็วเสียงที่อุณหภูมิห้องแล้วแสดงผลออกทั้งหน้าจอเพื่อเป็นการหาระยะทาง (distance) ระหว่างจุดสองจุด เนื่องจากปัจจุบันมีอุปกรณ์และเครื่องมือที่สนับสนุนเครื่องมือวัดผ่านระบบดิจิทัลทางเราจึงได้คิดที่จะทำโครงการนี้ขึ้นมา โดยการคาดคะเนที่มี Error ใกล้เคียงกับค่าที่วัดจริง และสามารถนำไปใช้ได้จริง

## 2. ขั้นตอนการทำงาน

### 2.1 อุปกรณ์ที่ใช้

2.1.1 pin Header

2.1.2 เซนเซอร์ Ultrasonic Module HC-SR04 Distance Measuring Transducer \ Sensor

2.1.3 Resister 500 ohm

2.1.4 LCD(Blue Screen) 16x2 LCD with backlight of the LCD Screen พร้อม I2C Interface

2.1.5 Lithium Battery Charging and protect one plate + protection 2in1 18650 lithium

battery charging board

2.1.6 สวิตช์

2.1.7 สายไฟ

2.1.8 แบตเตอรี่ แบบ Li-ion 18650 2850mah

2.1.9 Nucleo-32

2.1.10 pcb

2.1.11 กล่อง

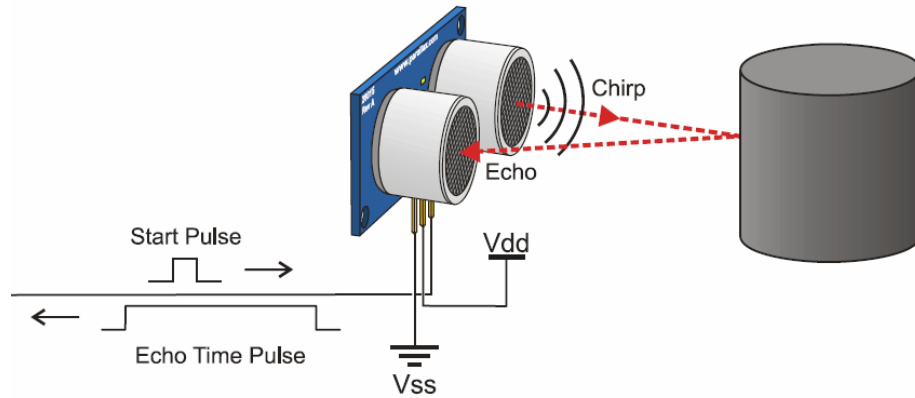
2.1.12 jumper 4 pin

2.1.13 dc dc stepUp Module

2.1.14 HR-SC04 Ultrasonic 4 pin

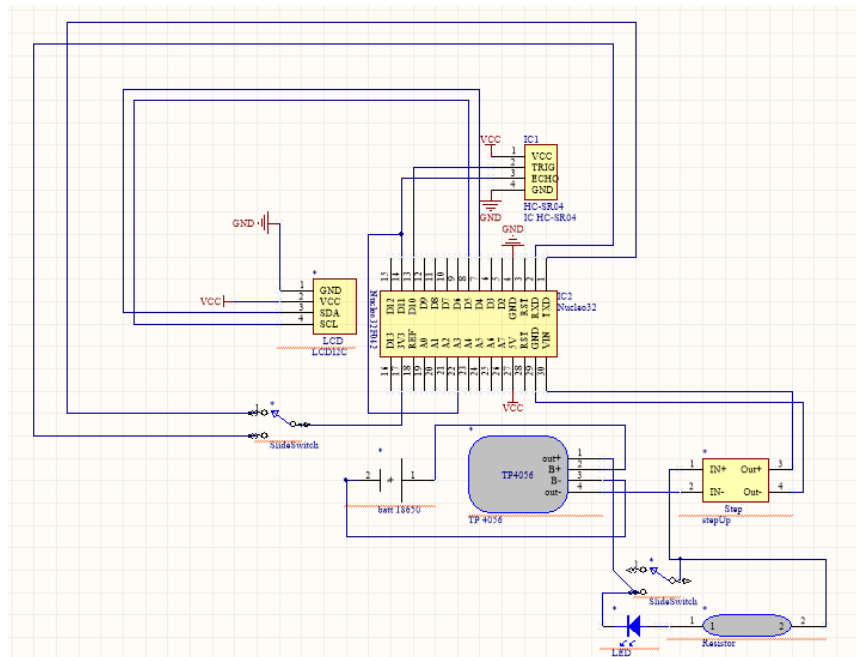
2.1.15 LED 3 mm

## 2.2 การทำงาน



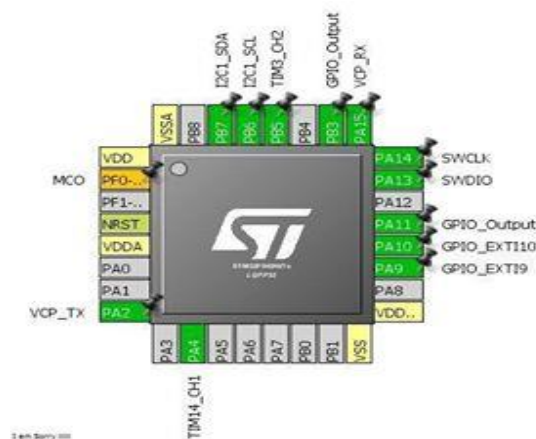
เครื่องวัดระยะทาง ใช้วัดระยะทางด้วยโมดูลอัลตราโซนิก เมื่อส่งสัญญาณเข้าที่ขา trig ของโมดูล เป็นการส่งคลื่นเสียงออกไปกระทบวัตถุและสะท้อนกลับมา จากนั้นจะได้สัญญาณทางขา Echo กลับมา โดยความกว้าง pulse ของสัญญาณดังกล่าวคือเวลาที่เสียงใช้เดินทาง mcu วัดความกว้าง pulse จาก TIM capture 2ค่า ขอบขาขึ้นและขาลง และไปคำนวณระยะทางโดยความเร็วเสียงคือ  $331+0.6(t)$  m/s เมื่อ  $t$  คืออุณหภูมิ สามารถคำนวณระยะทางได้จาก (ความเร็วเสียง x ระยะเวลาที่เสียงเดินทางไปและกลับ) / 2 เมื่อได้ค่าระยะทางแล้ว จะแสดงผลผ่านทางจอ LCD ที่มีการสื่อสารรูปแบบ i2c และ เข้าคอมพิวเตอร์ด้วย uart อีกทั้งยังมีส่วน interrupt จากสวิทช์ที่นำไปใช้ในการเปลี่ยนหน่วยวัดระหว่าง cm และ inch

## 2.3 การออกแบบวงจร

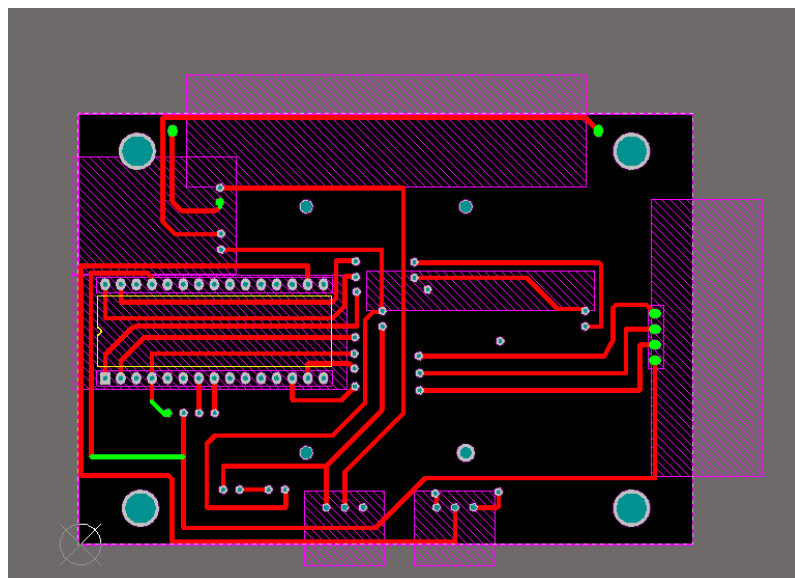


ภาพที่ 2.3.1 Schematic

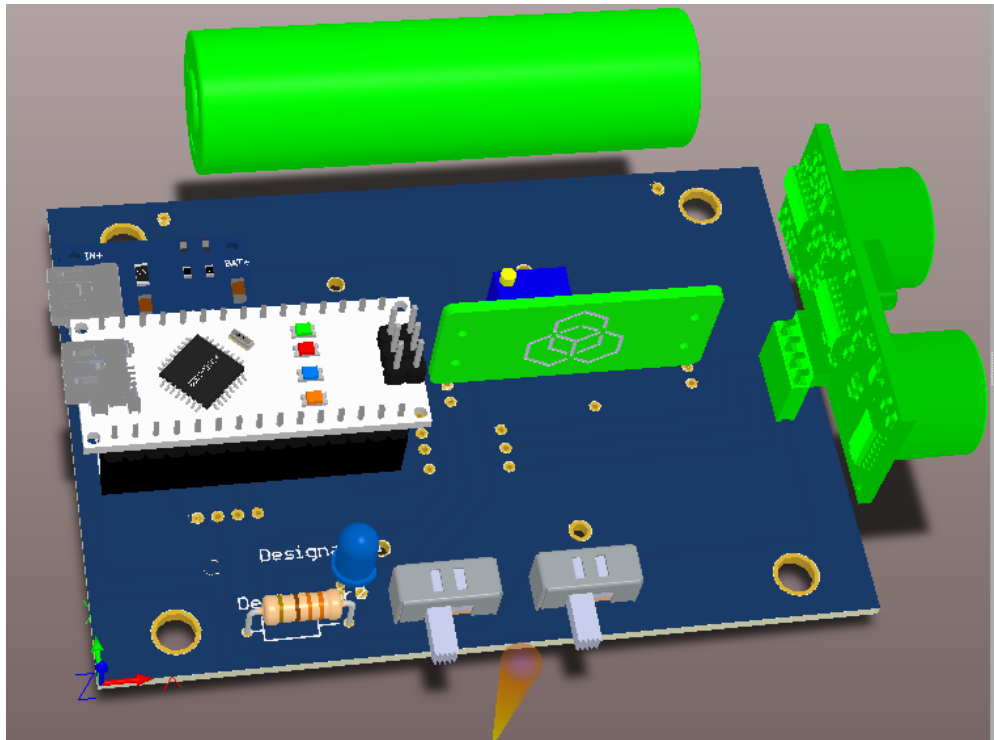
โมดูลชาร์ตทำหน้าที่ชาร์ตและจ่ายไฟจากแบตเตอรี่ มีโมดูล stepUP เพื่อจ่ายไฟให้ mcu โดยมีสวิตช์ และ led เพื่อทำการเปิดปิด ส่วนของ lcd ต่อใช้งานแบบ i2c,ultrasonic และสวิตช์เปลี่ยนการคำนวณ ใช้ gpio I/O



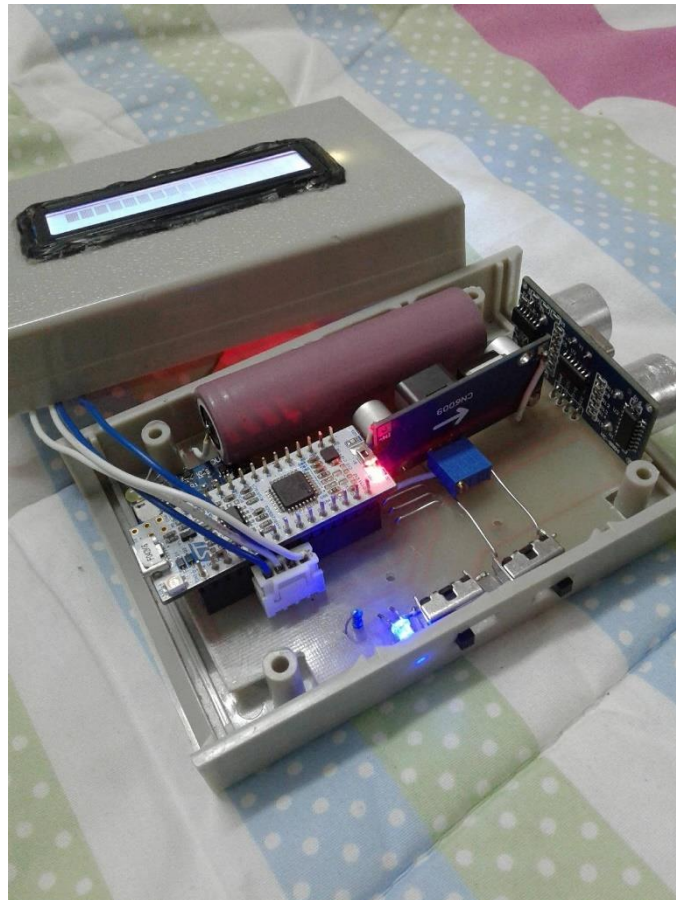
ภาพที่ ออกแบบใน CubeMx



ภาพที่ 2.3.2 วงจร 2 มิติ



ภาพที่ 2.3.3 วงจร 3 มิติ



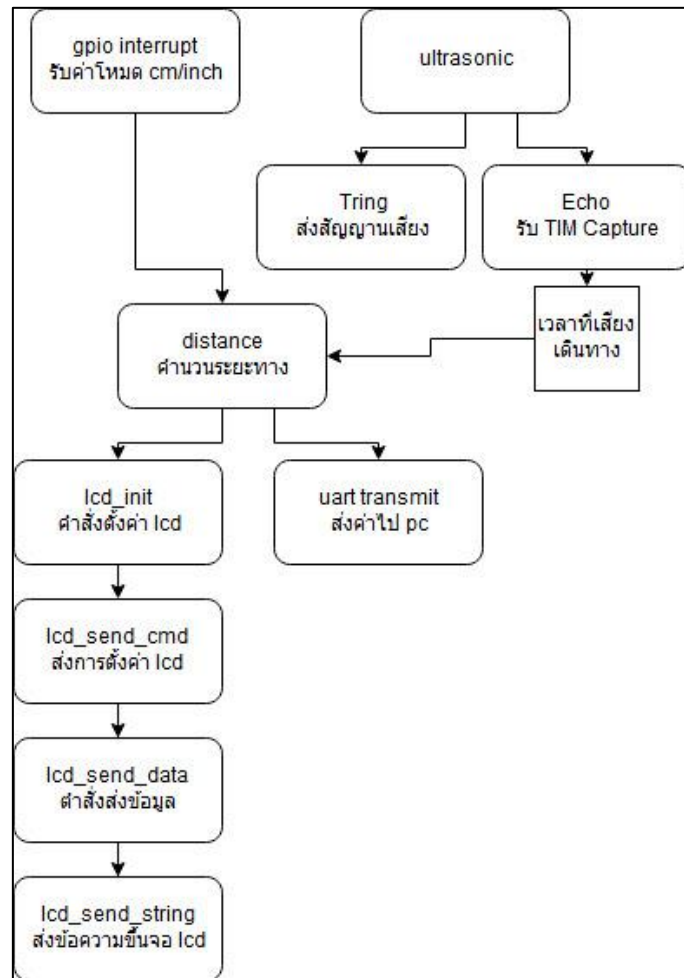
ภาพที่ 2.3.4 ลงอุปกรณ์จริง



วิดีโอการทดลอง :

<https://drive.google.com/file/d/1JE92YKRpQ2qpM0uttkVXJdVbRTdZCLQJ/view?usp=sharing>

## 2.4 โปรแกรม



ภาพที่ 2.4.1 ฟังก์การทำงาน

### Uart

```
sprintf(stringUsart, "Dist: %.3f CM\r\n", distance);  
HAL_UART_Transmit(&huart2, (uint8_t*)stringUsart, strlen(stringUsart), HAL_MAX_DELAY);
```

เป็นการนำค่าระยะทางที่อ่านได้ใส่ในสตริงเพื่อส่งทาง uart ไปยังคอมพิวเตอร์ที่  
ความเร็ว 115200



## LCD i2c

```
void lcd_send_cmd (char cmd)
{
    char data_u, data_l;
    uint8_t data_t[4]; //data transmit
    data_u = cmd&0xf0; // upper
    data_l = (cmd<<4)&0xf0; // lower
    data_t[0] = data_u|0x0c;; //en=1, rs=0
    data_t[1] = data_u|0x08; //en=0, rs=0
    data_t[2] = data_l|0x0c;; //en=1, rs=0
    data_t[3] = data_l|0x08; //en=0, rs=0
    HAL_I2C_Master_Transmit (&hi2c1, 0x4E,(uint8_t *) data_t, 4, 100);
}

void lcd_send_data (char data)
{
    char data_u, data_l;
    uint8_t data_t[4];
    data_u = data&0xf0; //upper
    data_l = (data<<4)&0xf0; //lower
    data_t[0] = data_u|0x0d; //en=1, rs=0    data_u|0x05;
    data_t[1] = data_u|0x09; //en=0, rs=0
    data_t[2] = data_l|0x0d; //en=1, rs=0
    data_t[3] = data_l|0x09; //en=0, rs=0
    HAL_I2C_Master_Transmit (&hi2c1, 0x4E,(uint8_t *) data_t, 4, 100);
}

void lcd_init (void)
{
    lcd_send_cmd (0x01);
    lcd_send_cmd (0x02);
    lcd_send_cmd (0x28);
    lcd_send_cmd (0x0c);
    lcd_send_cmd (0x80);

    //DISP_ON 0x0C //LCD control constants
    //DISP_OFF 0x08 //
    //CLR_DISP 0x01 //
    //CUR_HOME 0x02 //
    //ENTRY_INC 0x06 //
```

การสื่อสาร i2c lcd โดยส่งทีละ 4bit upper และ lower เมื่อนำมา or กับ controlcode ก็จะได้คำสั่งที่ต้องส่งให้ lcd (ใช้address 000)

## Ultrasonic

### -Trig

```
HAL_GPIO_WritePin(GPIOA, GPIO_PIN_11, GPIO_PIN_SET);
HAL_Delay(1);
HAL_GPIO_WritePin(GPIOA, GPIO_PIN_11, GPIO_PIN_RESET);
```

เป็นการสั่งให้ ultrasonic ส่งสัญญาณออกไปสะท้อนกับวัตถุ

### -Echo

```
void HAL_TIM_IC_CaptureCallback(TIM_HandleTypeDef *htim)//////////
{
    if(htim -> Instance == TIM3)
    {
        input_capture1 = __HAL_TIM_GET_COMPARE(&htim3,TIM_CHANNEL_2); //read tim3ch2 capture value
    }
    else if(htim -> Instance == TIM14)
    {
        input_capture2 = __HAL_TIM_GET_COMPARE(&htim14,TIM_CHANNEL_1); //read tim3ch2 capture value
    }
}
```

เป็นการ ใช้ TIM capture โดย TIM3 จับขอบขาขึ้น TIM14 จับขอบขาลง  
แล้วนำเวลาที่ได้มาลบกันก็จะได้ความกว้างของ pulse ของคลื่นที่เข้ามา

### -Calculate Distance

```
void distance(float time,int unit)
{
    float temp = 30.0;
    float speed_of_sound = 0.0;
    speed_of_sound = 331+0.6*(temp);
    float distance = 0.0;
    distance = (speed_of_sound * time)/2;
    distance = distance/1000;

    switch(unit)
    {
        case 0:
            //sprintf(stringLCD, "%d", Unit);
            sprintf(stringLCD, "Dist: %.3f CM",distance);
            lcd_init ();
            lcd_send_string (stringLCD);
            //printf("unit = %f m",distance);
            break;
        case 1:
            distance = distance*0.393701;
            sprintf(stringLCD, "Dist: %.3f Inch",distance);
            lcd_init ();
            lcd_send_string (stringLCD);
            break;
    }
}
```

การจะวัดความเร็วด้วยเสียงนั้นต้องกำหนด ค่าตัวแปร 1 ตัวมีหน่วย เมตร/วินาที ใน  
ที่นี้คือ Speed\_of\_sound ซึ่งประกาศไว้เก็บค่าความเร็วของเสียงโดยมีสูตรคำนวณทางฟิสิกส์คือ  
 $V(t) = 331 + 0.6(T)$  ; โดย T เป็นหน่วยอุณหภูมิอุณหภูมิในที่นี้คือ อุณหภูมิห้องเฉลี่ยที่ 30 องศา  
โดยมีการแปลงหน่วย 2 ค่า คือหน่วย เซนติเมตร และ หน่วยนิ้ว

## -Interrupt

```
void HAL_GPIO_EXTI_Callback(uint16_t GPIO_Pin)
{
    sw1 = HAL_GPIO_ReadPin(GPIOA,GPIO_PIN_9);
    sw2 = HAL_GPIO_ReadPin(GPIOA,GPIO_PIN_10);

    if (sw1 == 0 & sw2 == 0)
    {

    }
    else if (sw1 == 0 & sw2 == 1)
    {
        Unit = 0;

        sprintf(stringLCD, "Unit : Centimetre");
        lcd_init ();
        lcd_send_string (stringLCD);

    }
    else if (sw1 == 1 & sw2 == 0)
    {
        Unit = 1;
        sprintf(stringLCD, "Unit : Inch");
        lcd_init ();
        lcd_send_string (stringLCD);

    }
    else if (sw1 ==1 & sw2 == 1)
    {

    }

}
```

Function นี้เป็นการรับค่าสวิตช์ไปเปลี่ยนการคำนวณหน่วย cm กับ inch โดยเริ่มแรกต้องประกาศสวิตช์ sw1 , sw2 จากนั้นใส่เงื่อนไข if , else เพื่อเช็คค่า สวิตช์ที่เปลี่ยนนั้นต้องการให้แสดงหน่วย cm หรือ inch ถ้า sw1 เป็น 0 และ sw2 เป็น 1 ให้แสดงหน่วย cm, ถ้า sw1 เป็น 1 และ sw2 เป็น 0 ให้แสดงหน่วย inch และ ถ้า sw1 และ sw2 เป็น 1 กับ sw1 และ sw2 เป็น 0 จะไม่มีผลอะไร

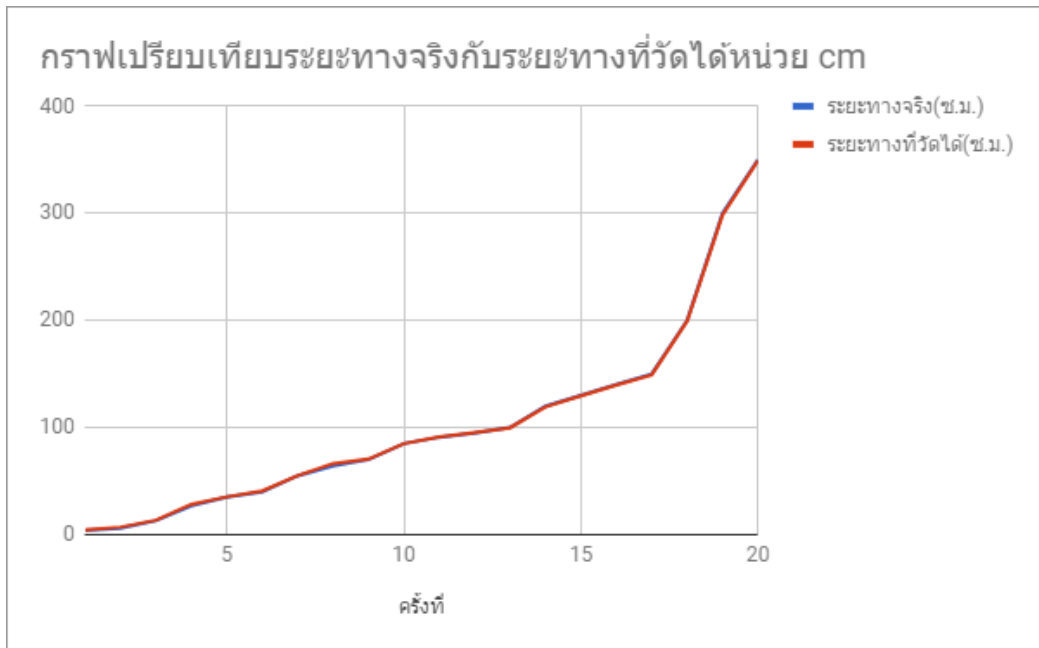
### 3.ผลการทดสอบ

- อุปกรณ์ที่ใช้วัดระยะคือ ตลับเมตร

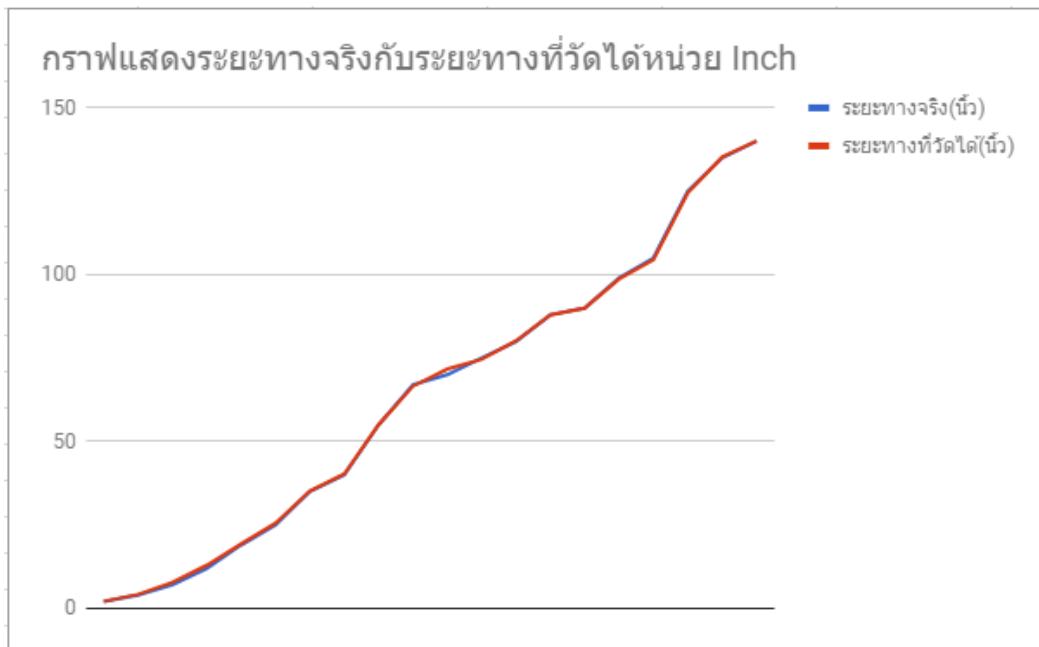


ตารางการทดสอบ (หน่วยวัด เซนติเมตร)			
ครั้งที่	ระยะทางจริง(ซ.ม.)	ระยะทางที่วัดได้(ซ.ม.)	% ความคลาดเคลื่อน
1	4	4.233	5.825
2	6	6.98	16.33
3	13	13.436	3.35
4	27	28.269	4.7
5	35	35.249	0.7
6	40	40.659	1.64
7	55	55.142	0.25
8	64	66.314	3.615
9	70	70.498	0.711
10	85	84.981	-0.02
11	91	91.085	0.093
12	95	95.103	0.108
13	100	99.841	-0.159
14	120	119.331	-0.55
15	130	129.654	-0.27
16	140	139.6	-0.28
17	150	149.167	-0.55
18	200	199.279	-0.36
19	300	298.395	-0.53
20	350	349.174	-0.23
เฉลี่ย			5.587

ตารางการทดสอบ (หน่วยวัด นิ้ว)			
ครั้งที่	ระยะทางจริง(นิ้ว)	ระยะทางที่วัดได้(นิ้ว)	% ความคลาดเคลื่อน
1	2	2.107	5.35
2	4	4.191	4.775
3	7	7.832	11.88
4	12	12.916	7.63
5	19	19.305	1.605
6	25	25.557	2.228
7	35	35.175	0.5
8	40	40.259	0.647
9	55	55.029	0.05
10	67	66.571	-0.65
11	70	71.729	2.47
12	75	74.609	-0.521
13	80	80.174	0.217
14	88	87.937	-0.07
15	90	89.998	-0.002
16	99	98.723	-0.27
17	105	104.425	-0.54
18	125	124.55	-0.36
19	135	135.222	0.16
20	140	140.012	0.0085
เฉลี่ย			3.357



ภาพการเปรียบเทียบระยะทางจริงกับระยะทางที่วัดได้(หน่วย เซนติเมตร)



ภาพการเปรียบเทียบระยะทางจริงกับระยะทางที่วัดได้(หน่วย นิ้ว)

#### 4.สรุป

โครงการนี้ได้นำความรู้จากการเรียนมาใช้จริง ทั้งการออกแบบ 2d 3d การออกแบบลายวงจร มีการสื่อสารผ่านโปรโตคอล i2c uart มีการนำ interrupt และ Timer มาใช้เพื่อสร้างเครื่องวัดระยะทางซึ่งสามารถใช้งานได้ตามผลการทดลองที่ระยะใกล้ความคาดเคลื่อนจะมีมากเมื่อระยะห่างออกไปค่าความคาดเคลื่อนก็น้อย และสามารถเปลี่ยนการวัดเป็น cm และ inch ได้ ในระยะที่ระบุไว้ในสเปคโมดูล 4-400cm จึงสามารถพกพานำไปใช้งานจริงในการวัดระยะต่างแทนไม้บรรทัดและตลับเมตรได้



## แหล่งอ้างอิง

[1] STM32F042K6 datasheet, สืบค้นเมื่อวันที่ 11 ธันวาคม 2560, จากชื่อเว็บไซต์:

[http://www.st.com/content/st\\_com/en/products/microcontrollers/stm32-32-bit-arm-cortex-mcus/stm32-mainstream-mcus/stm32f0-series/stm32f0x2/stm32f042k6.html](http://www.st.com/content/st_com/en/products/microcontrollers/stm32-32-bit-arm-cortex-mcus/stm32-mainstream-mcus/stm32f0-series/stm32f0x2/stm32f042k6.html)

[2] การใช้งาน LCD , สืบค้นเมื่อวันที่ 11 ธันวาคม 2560, จากชื่อเว็บไซต์:

<http://www.thaieasyelec.com/article-wiki/review-product-article/%E0%B8%81%E0%B8%B2%E0%B8%A3%E0%B9%83%E0%B8%8A%E0%B9%89%E0%B8%87%E0%B8%B2%E0%B8%99-character-lcd-display-%E0%B8%81%E0%B8%B1%E0%B8%9A-arduino-%E0%B8%95%E0%B8%AD%E0%B8%99%E0%B8%97%E0%B8%B5%E0%B9%882-%E0%B8%A3%E0%B8%B9%E0%B8%9B%E0%B9%81%E0%B8%9A%E0%B8%9A%E0%B8%81%E0%B8%B2%E0%B8%A3%E0%B9%80%E0%B8%8A%E0%B8%B7%E0%B9%88%E0%B8%AD%E0%B8%A1%E0%B8%95%E0%B9%88%E0%B8%AD%E0%B9%81%E0%B8%9A%E0%B8%9A-i2c.html>

[3] Hc-sr04 datasheet, สืบค้นเมื่อวันที่ 11 ธันวาคม 2560, จากชื่อเว็บไซต์:

<http://www.micropik.com/PDF/HCSR04.pdf>

[4] Ultrasonic, สืบค้นเมื่อวันที่ 11 ธันวาคม 2560, จากชื่อเว็บไซต์:

<http://www.mtechlog.com/2015/09/using-ultrasonic-sensor-with-arduino.html>

