

การอินเตอร์เฟสกับ DS1302 Real-time clock

วัตถุประสงค์

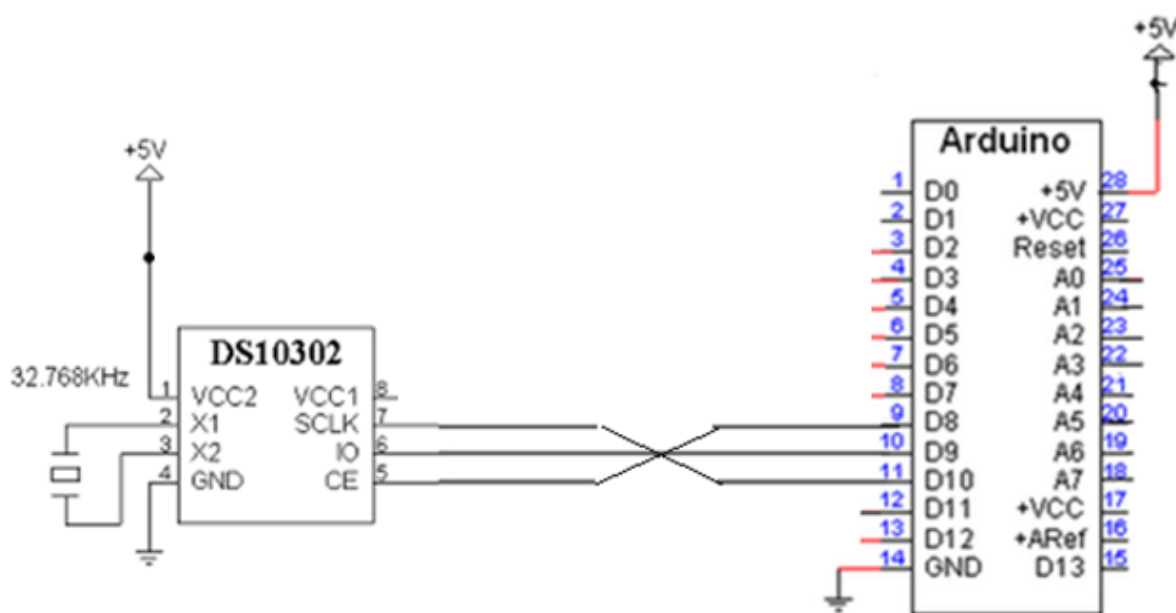
1. เพื่อให้เข้าใจถึงวิธีการอินเตอร์เฟสแบบ 3 Wire
2. เพื่อให้เข้าใจการทำงานของ Real-time clock
3. สามารถใช้งานไมโครคอนโทรลเลอร์ทำงานร่วมกับ Real-time clock โดยผ่านวิธีการอินเตอร์เฟสแบบ 3-Wire ได้

บทนำ

การทดลองนี้ ได้แสดงวิธีการใช้งานไอซี Real-time clock เบอร์ DS1302 ซึ่งมีการอินเตอร์เฟสแบบ 3-Wire โดยผ่านการควบคุมของไมโครคอนโทรลเลอร์ Arduino

การทดลองที่ 1

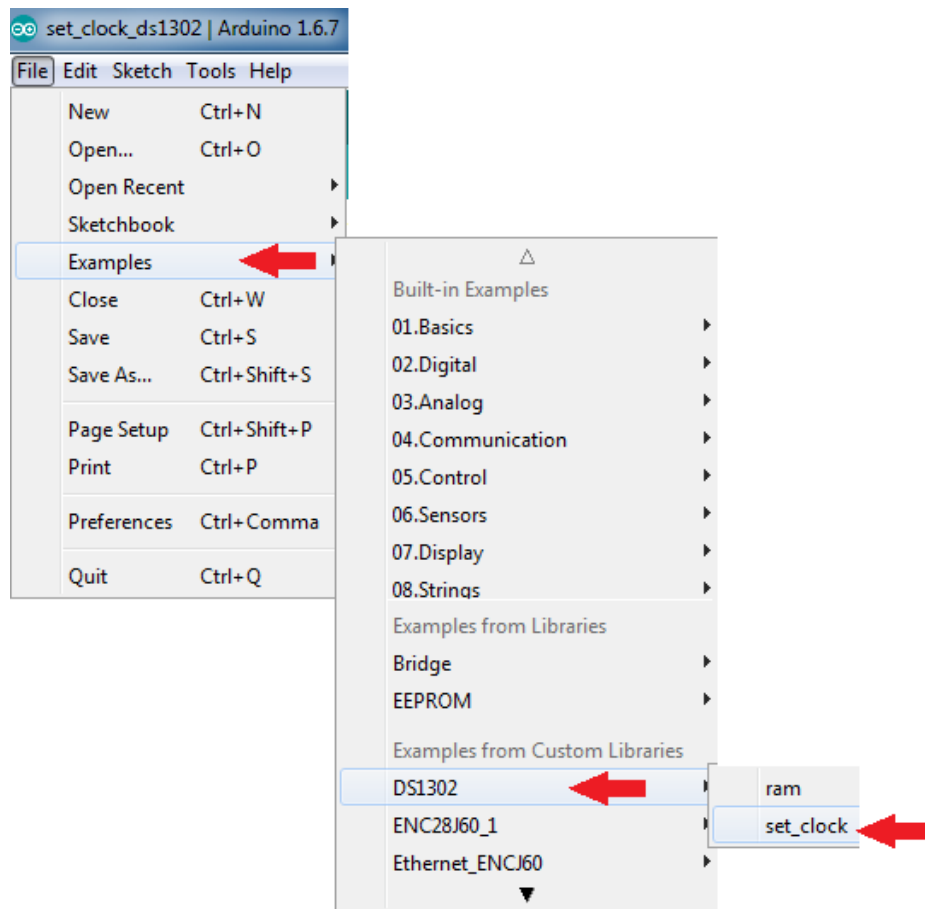
1. ประกอบวงจรตามรูปที่ 1



รูปที่ 1 การเชื่อมต่อ Arduino กับ Real-time clock DS1302

2. รัน Arduino แล้ว ให้ใช้เมนู File -> Examples มี DS1302 อยู่หรือไม่ ตามรูปที่ 2 ถ้ามีให้เลือกไฟล์ set_clock แล้วทำข้อ 6 แต่ถ้าไม่มีให้ทำข้อ 3 ถึงข้อ 5 ก่อน
3. ให้ไป download ไลบรารี DS1302 ชื่อ arduino-ds1302-master.zip มาจาก <https://github.com/msparks/arduino-ds1302>
4. ทำการ unzip ไฟล์ ไปเก็บที่ โฟลเดอร์ย่อย libraries ซึ่งอยู่ในโฟลเดอร์ Sketch ของ Arduino **ควรเปลี่ยนชื่อโฟลเดอร์เป็น DS1302 ด้วย**

5. ปิดโปรแกรม Arduino แล้ว รัน Arduino อีกครั้ง คราวนี้จะมีไลบรารี DS1302 และตัวอย่างแล้วให้ตรวจสอบตามข้อ 2 ใหม่ ถ้ายังไม่มีให้ทำข้อ 5 ใหม่



รูปที่ 2

6. เพื่อให้ตรงกับวงจรที่จะใช้ ให้แก้ไขโปรแกรม บรรทัดที่เขียนคำสั่ง

```
const int kCePin = 5; // Chip Enable
```

```
const int kloPin = 6; // Input/Output
```

```
const int kSclkPin = 7; // Serial Clock
```

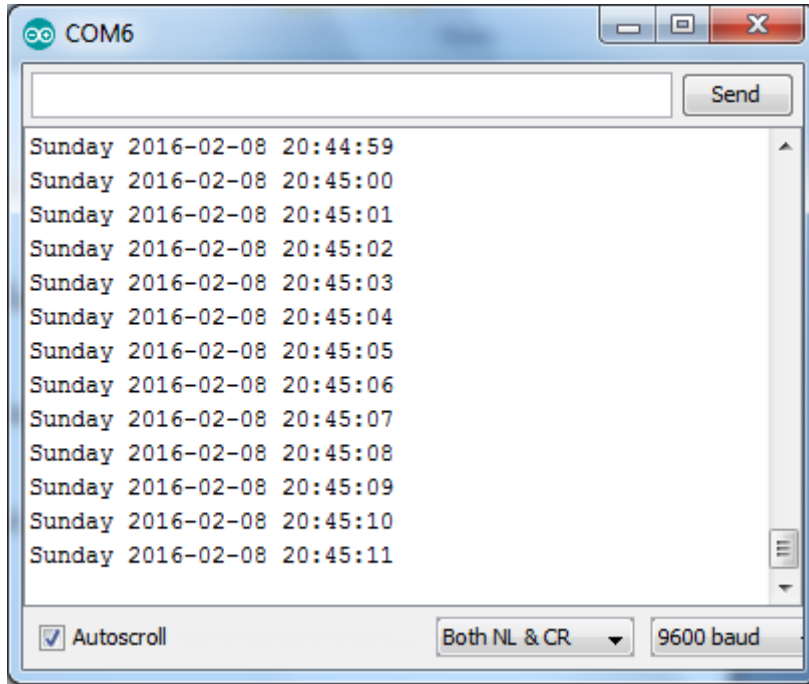
เป็น

```
const int kCePin = 8; // Chip Enable
```

```
const int kloPin = 9; // Input/Output
```

```
const int kSclkPin = 10; // Serial Clock
```

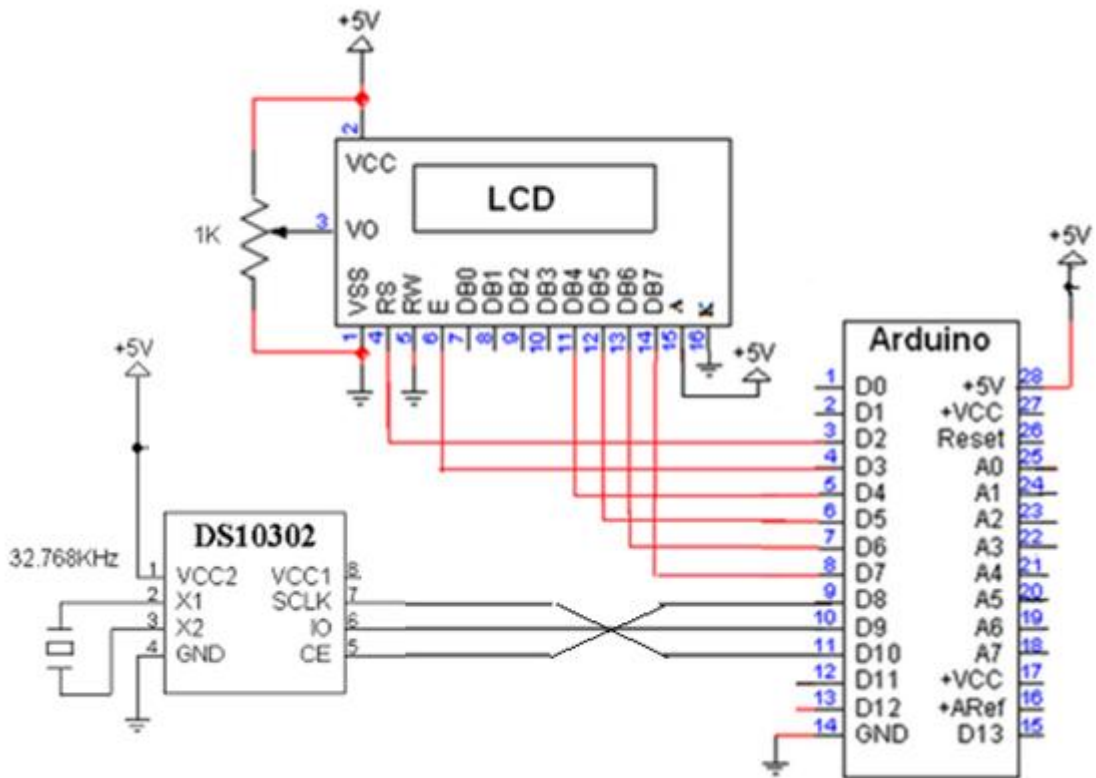
ให้รันโปรแกรมตัวอย่าง DS18x20 ตามรูปที่ 1 และเปิด Serial Monitor จะได้ผลลัพธ์ตามรูปที่ 3



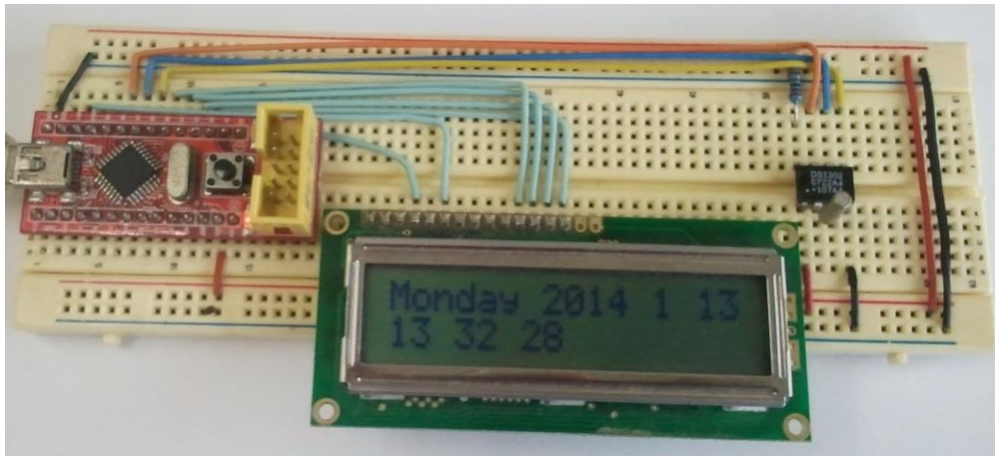
รูปที่ 3

การทดลองที่ 2

1. ให้ต่อวงจรตามรูปที่ 3 แล้วเขียนโปรแกรมแสดงวันที่และเวลาออกทาง LCD คล้ายตัวอย่างในรูปที่ 4



รูปที่ 3 การเชื่อมต่อ Arduino กับ Real-time clock DS1302 และ LCD



รูปที่ 4 การแสดงค่าวันเวลาออกทาง LCD

โปรแกรม

```
#include <stdio.h>
#include <string.h>
#include <LiquidCrystal.h>
#include <DS1302.h>

LiquidCrystal lcd(2, 3, 4, 5, 6, 7); // initialize the library with the numbers of the interface pins

/* Set the appropriate digital I/O pin connections */
uint8_t CE_PIN = 10;
uint8_t IO_PIN = 9;
uint8_t SCLK_PIN = 8;

/* Create buffers */
char buf[50];
char day[10];

/* Create a DS1302 object */
DS1302 rtc(CE_PIN, IO_PIN, SCLK_PIN);

void print_time()
{
    /* Get the current time and date from the chip */
    Time t = rtc.time();

    /* Name the day of the week */
    memset(day, 0, sizeof(day)); /* clear day buffer */
    switch (t.day) {
        case 1:
            strcpy(day, "Sunday");
            break;
```

```

case 2:
    strcpy(day, "Monday");
    break;
case 3:
    strcpy(day, "Tuesday");
    break;
case 4:
    strcpy(day, "Wednesday");
    break;
case 5:
    strcpy(day, "Thursday");
    break;
case 6:
    strcpy(day, "Friday");
    break;
case 7:
    strcpy(day, "Saturday");
    break;
}

/* Format the time and date and insert into the temporary buffer */
sprintf(buf, sizeof(buf), "%s %04d-%02d-%02d %02d:%02d:%02d",
        day,
        t.yr, t.mon, t.date,
        t.hr, t.min, t.sec);

/* Print the formatted string to serial so we can see the time */
Serial.println(buf);
}
void printtime2LCD()
{
    Time t = rtc.time();
    lcd.setCursor(0, 0);
    lcd.print(day);
    lcd.print(" ");
    lcd.print(t.yr);
    lcd.print(" ");
    lcd.print(t.mon);
    lcd.print(" ");
    lcd.print(t.date);

    lcd.setCursor(0, 1);
    lcd.print(t.hr);
    lcd.print(" ");
    lcd.print(t.min);
    lcd.print(" ");
    lcd.print(t.sec);
}
void setup()

```

```

{
  Serial.begin(9600);

  lcd.begin(16, 2); // set up the LCD's number of columns and rows:

  lcd.print("Real time clock"); // Print a message to the LCD.

  /* Initialize a new chip by turning off write protection and clearing the
  clock halt flag. These methods needn't always be called. See the DS1302
  datasheet for details. */
  rtc.write_protect(false);
  rtc.halt(false);

  /* Make a new time object to set the date and time */
  /* Monday, Jan 13, 2014 at 13:30:37.      */
  Time t(2014, 1, 13, 13, 30, 37, 2);

  /* Set the time and date on the chip */
  rtc.time(t);
  delay(1000);
}

/* Loop and print the time every second */
void loop()
{
  print_time();
  printtime2LCD();
  delay(1000);
}

```

งานมอบหมาย

ให้เขียนโปรแกรมเพื่อทำเป็นนาฬิกาที่บอกได้ทั้งเวลาและวันที่ และสามารถตั้งเวลาได้โดยใช้
โปรแกรม Processing (ออกแบบรูปให้สวยงาม)