

```

#include <Wire.h>
#include <LiquidCrystal.h>

const int LED = 8;
const int VALVE = 9;
const int RESET = 13;

int SWstate = 0;
int LEDstate = 0;
int waterLV = 0;
int DIPSW = 0;

unsigned long base_time = 0;
unsigned long previous_time = 0;
unsigned long second = 0;
unsigned long minute = 0;
unsigned long hour = 0;

LiquidCrystal lcd = LiquidCrystal(12,11,10,5,4,3,2);

#ifdef ARDUINO_SAMD_VARIANT_COMPLIANCE
#define SERIAL SerialUSB
#else
#define SERIAL Serial
#endif

unsigned char low_data[8] = {0};
unsigned char high_data[12] = {0};

#define NO_TOUCH                0xFE
#define THRESHOLD                100
#define ATTINY1_HIGH_ADDR      0x78
#define ATTINY2_LOW_ADDR       0x77

void getHigh12SectionValue(void)
{
    memset(high_data, 0, sizeof(high_data));
    Wire.requestFrom(ATTINY1_HIGH_ADDR, 12);
    while (12 != Wire.available());

```

```

    for (int i = 0; i < 12; i++) {
        high_data[i] = Wire.read();
    }
    delay(10);
}

void getLow8SectionValue(void)
{
    memset(low_data, 0, sizeof(low_data));
    Wire.requestFrom(ATTINY2_LOW_ADDR, 8);
    while (8 != Wire.available());

    for (int i = 0; i < 8; i++) {
        low_data[i] = Wire.read(); // receive a byte as character
    }
    delay(10);
}

void check()
{
    int sensorvalue_min = 250;
    int sensorvalue_max = 255;
    int low_count = 0;
    int high_count = 0;
    int wataerLV = 0;

    uint32_t touch_val = 0;
    uint8_t trig_section = 0;
    low_count = 0;
    high_count = 0;
    getLow8SectionValue();
    getHigh12SectionValue();

    for (int i = 0; i < 8; i++) {
        if (low_data[i] > THRESHOLD) {
            touch_val |= 1 << i;
        }
    }
}

for (int i = 0; i < 12; i++) {

```

```

        if (high_data[i] > THRESHOLD) {
            touch_val |= (uint32_t)1 << (8 + i);
        }
    }

    while (touch_val & 0x01)
    {
        trig_section++;
        touch_val >>= 1;
    }
    lcd.begin(16, 2);
    lcd.print("water = ");
    if (trig_section == 0 ){
        lcd.print("0-100cc");
        lcd.setCursor(0,1);
        lcd.print("Unmeasurable");
    }
    else{
        lcd.print(trig_section * 20 + 100);
        lcd.println(" cc ");
    }
    waterLV = trig_section;
    delay(10);
}

void checkDIP()
{
    pinMode(A0, INPUT);
    pinMode(A1, INPUT);
    pinMode(A2, INPUT);
    pinMode(A3, INPUT);
    if(digitalRead(A0) == 1 && digitalRead(A1) == 0 && digitalRead(A2) == 0 &&
digitalRead(A3) == 0){
        DIPSW = 0;
    }else if(digitalRead(A0) == 0 && digitalRead(A1) == 1 && digitalRead(A2) == 0 &&
digitalRead(A3) == 0){
        DIPSW = 5;
    }else if(digitalRead(A0) == 0 && digitalRead(A1) == 0 && digitalRead(A2) == 1 &&
digitalRead(A3) == 0){
        DIPSW = 10;
    }
}

```

```

        }else if(digitalRead(A0) == 0 && digitalRead(A1) == 0 && digitalRead(A2) == 0 &&
digitalRead(A3) == 1){
            DIPSW = 15;
        }else if(digitalRead(A0) == 1 && digitalRead(A1) == 1 && digitalRead(A2) == 1 &&
digitalRead(A3) == 1){
            DIPSW = 20;
        }
    }
}

```

```

void setup() {
    pinMode(LED, OUTPUT);
    pinMode(VAIVE, OUTPUT);
    pinMode(RESET, INPUT);

    digitalWrite(LED, HIGH);
    digitalWrite(VAIVE, HIGH);
    digitalWrite(RESET, HIGH);

    SERIAL.begin(9600);
    Wire.begin();
}

```

```

void loop()
{
    check();
    checkDIP();

    previous_time = millis() - base_time;

    if(previous_time >= 86400000) {
        base_time = millis();
        digitalWrite(RESET, LOW);
        waterLV = trig_section;

        }else if(waterLV > DIPSW){
            digitalWrite(VAIVE, LOW);
            digitalWrite(LED, LOW);
            check();
        }else{
            digitalWrite(VAIVE, HIGH);

```

```
        digitalWrite(LED, HIGH);
        waterLV = trig_section;
        check();
    }
    digitalWrite(VALUE, LOW);
}

second = previous_time / 1000;
minute = second / 60;
hour = minute / 60;

Serial.print(hour);
Serial.print(":");
Serial.print(minute);
Serial.print(":");
Serial.println(second);
Serial.print(DIPSW);
check();
}
```