**TASK1**

**TASK2**

The code from task one is better because it uses the systems current time. This means it can be adapted to other circumstances more easily.

**TASK3**

1. If the board has been left on for a long period of time using an int for time would cause an error.
2. Interval would need to be changed to a long int. 10 minutes comes out as 600000 miliseconds, but int can only store numbers up to 32,767 or 65,535 unsigned.

**TASK4**

1. 1.8v
2. 0v

**TASK5**

const int switchPin = 2; // named constant for the switch pin

unsigned long currentTime;

unsigned long previousTime = 0; // store the last time an LED was updated

int switchState = 0; // the current switch state

int prevSwitchState = 0; // the previous switch state

int led = 3; // a variable to refer to the LEDs

int interval = 5000; // interval at which to light the next LED (5 s)

int s2 = 6;

int s2State = 0;

void setup() {

// set the LED pins as outputs and turn LEDs off

for (int i = 3; i < 6; i++) {

pinMode(i, OUTPUT);

digitalWrite(i, LOW);

}

// set the switch pin as input

pinMode(switchPin, INPUT);

pinMode(s2, INPUT);

}

void loop() {

s2State = digitalRead(s2);

// store the time since the Arduino started running in a variable

if (s2State == HIGH) {

currentTime = millis();

// compare the current time to the previous time an LED turned on

// if it is greater than your interval, turn next LED on

if (currentTime - previousTime > interval && led < 6) {

// save the current time as the last time you changed an LED

previousTime = currentTime;

// Turn the LED on

digitalWrite(led, HIGH);

// increment the led variable

// in 5 s the next LED will light up

led++;

}

}

// read the switch value

switchState = digitalRead(switchPin);

// if the switch has changed

if (switchState != prevSwitchState) {

// turn all the LEDs low

for (int i = 3; i < 6; i++) {

for (int i = 3; i < 6; i++) {

digitalWrite(i, LOW);

}

// reset the LED variable to the first one

led = 3;

//reset the timer

previousTime = currentTime;

}

// set the previous switch state to the current state

prevSwitchState = switchState;

}

}