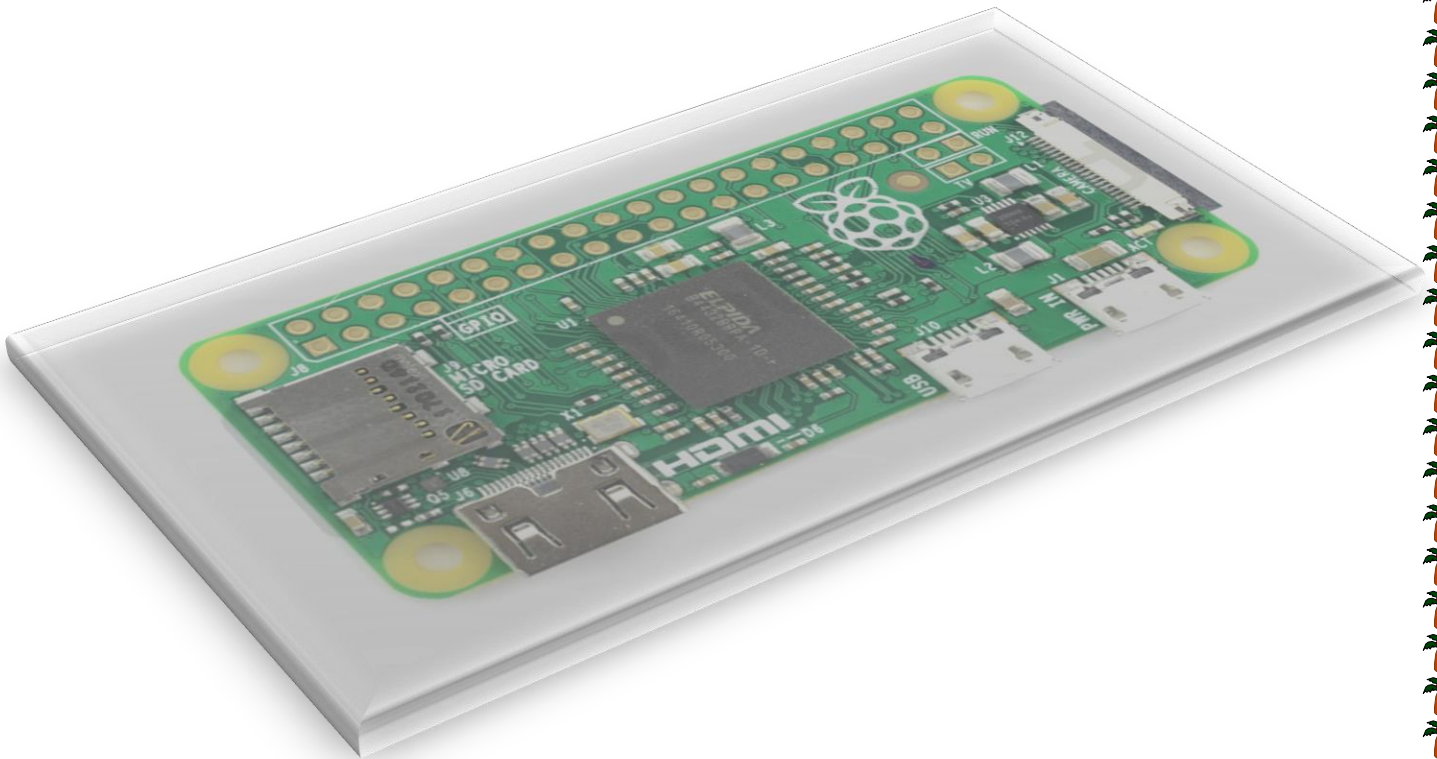




PRACTICAL 1

Using A Real-Time Clock



AUGUST 18, 2021

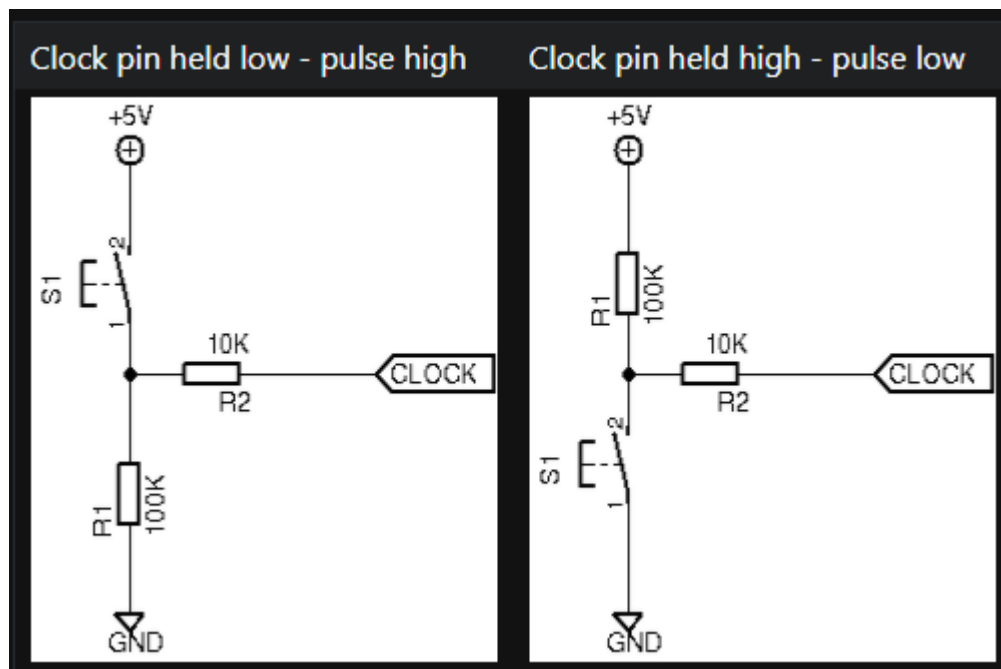
BY CLXBHE005 AND KMPKWEO02

1.1.7 Submission

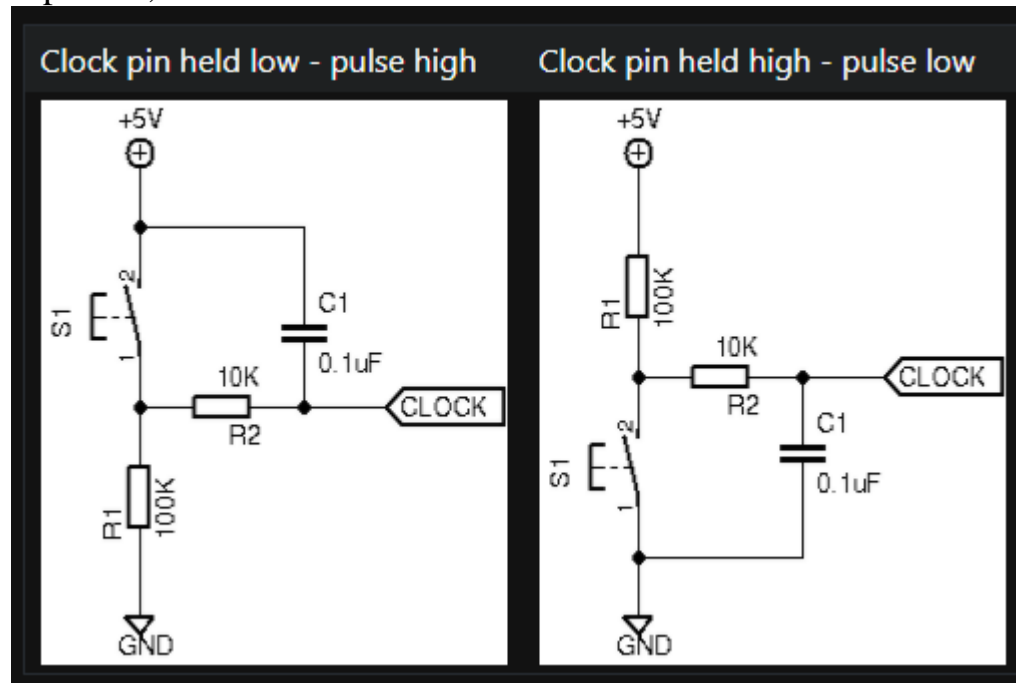
1. I2C (Inter-grated Circuit Protocol) is a protocol intended to allow multiple peripheral digital integrated circuits to communicate with one or more controller chips.
2. An interrupt is a signal to the processor emitted by hardware or software indicating an event that needs immediate attention. Interrupts are important in the world of embedded systems in a way that a CPU can enter a low power idle when there is no task running. Hardware interrupts will wake the CPU when there is some useful work to be done.
3.
 - a) To ensure the values of an I/P is always in a known state. The function of the pull-up resistor is that it pulls the signals state unless it is driven low, and the pull-down resistor pulls the signal to a low state unless it is driven high.
 - b) Hardware Debouncing is ensuring that only a single signal will be acted upon for a single opening or closing of contact using hardware (e.g., Capacitor) while Software Debouncing is ensuring that only a single signal will be acted upon for a single opening or closing of contact by writing a code in a program using interrupts.

Hardware debouncing

This is a circuit with no bounds at all, this circuit will produce a lot of bouncing once the switch is on



This is how we implement hardware debouncing by adding a ceramic capacitor, once the switch is on the circuit below will filter bounces.



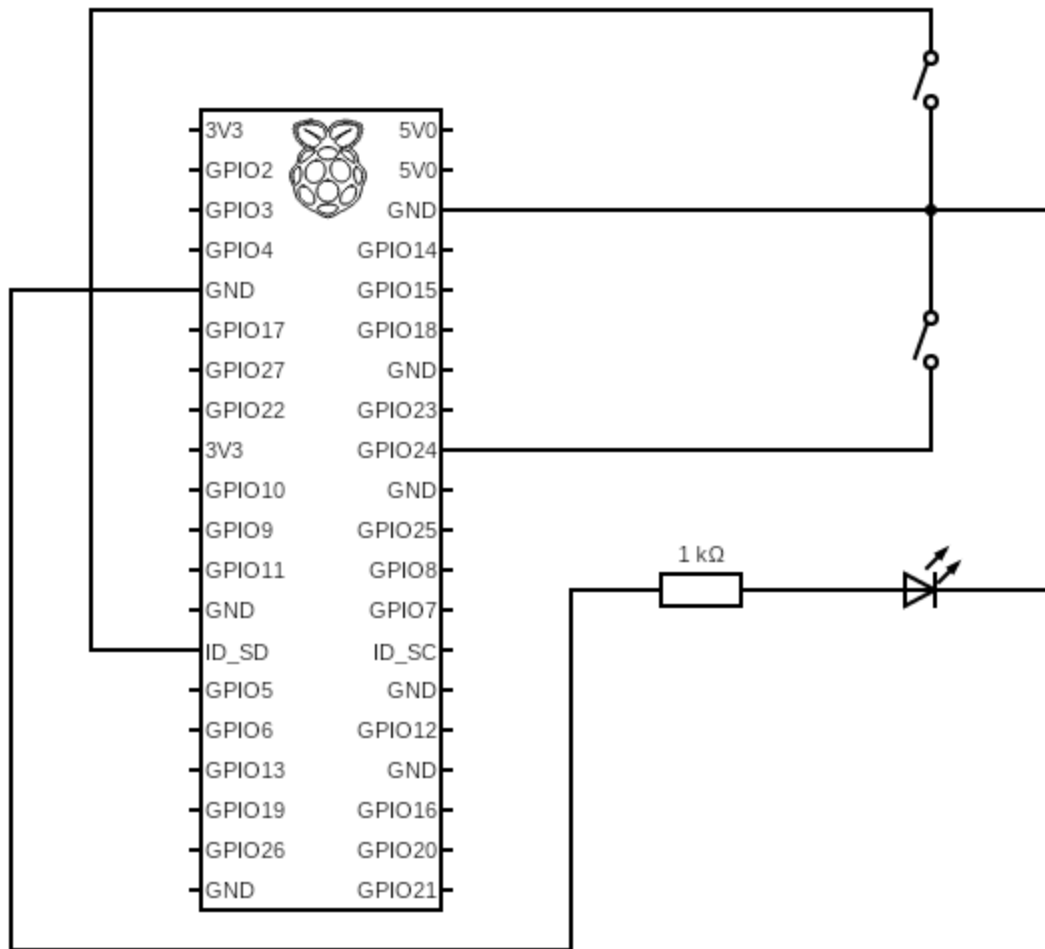
Software debouncing

The code below shows a snippet code of how we can implement debouncing

```
1. void hourInc(void){
2.     //Debounce
3.     long interruptTime = millis();
4.
5.     if (interruptTime - lastInterruptTime>300){
6.         printf("Interrupt 1 triggered, %x\n", hours);
7.         //Fetch RTC Time
8.         //Increase hours by 1, ensuring not to overflow
9.         //Write hours back to the RTC
10.        addHours(1);
11.    }
12.    lastInterruptTime = interruptTime;
13. }
```

- c) Polling means you don't know when the data is ready, but you can get it when you are ready, so in polling CPU continuously checks the status of the devices to find whether they require attention. The advantage is the performance of the microcontroller is far better in the Interrupt method than Polling Method.

4. The circuit diagram



5. Core Function

```
1. void blinkLED(void){
2.     digitalWrite(LED, HIGH);
3.     delay(1000);
4.     digitalWrite(LED, LOW);
5. }
6.
7. void addHours(int hours){
8.     int hourValue = wiringPiI2CReadReg8(RTC, HOUR_REGISTER);
9.     hourValue +=hours;
10.    if(hourValue>=24){
11.        hourValue = 00;
12.    }
13.    wiringPiI2CWriteReg8(RTC, HOUR_REGISTER, hourValue);
14. }
15.
16. void addMinutes(int minutes){
17.     int minValue = wiringPiI2CReadReg8(RTC, MIN_REGISTER);
18.     minValue +=minutes;
19.     if(minValue>=60){
20.         minValue = 00;
21.         addHours(1);
22.     }
23.     wiringPiI2CWriteReg8(RTC, MIN_REGISTER, minValue);
24.
25. }
```

6. Shared GitHub Repository: <https://github.com/BhekananiCele/EEE3095S.git>