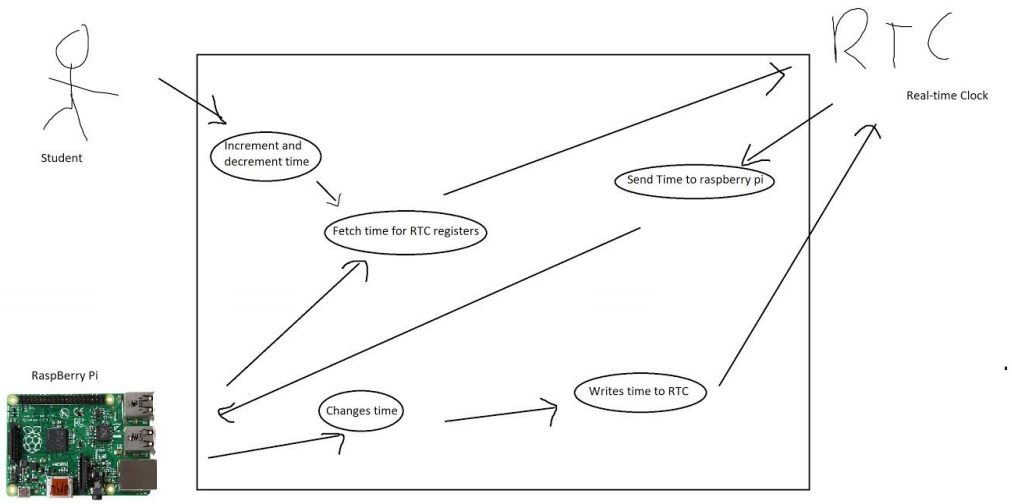
Prac 3 - I2C and PWM

Matthew Terblanche† and Michael Wetzel‡
EEE3096S Class of 2019
University of Cape Town
South Africa
†TRBMAT002 ‡WTZMIC001

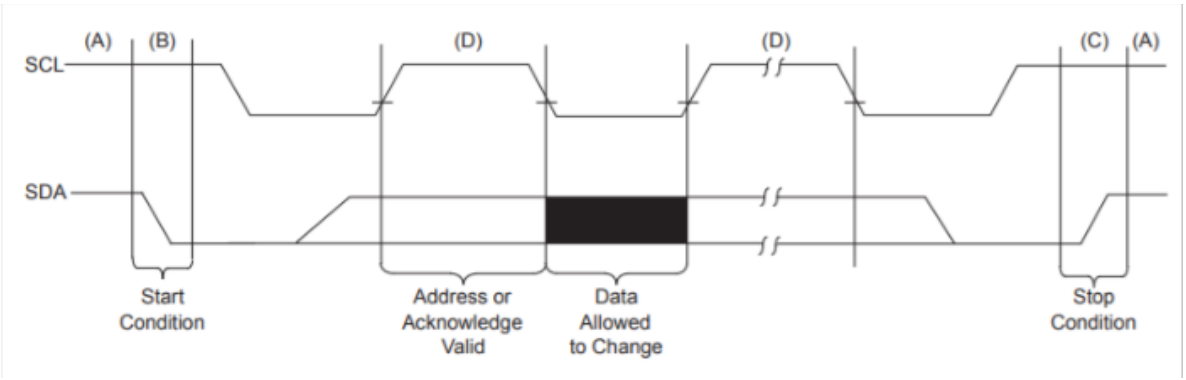
Introduction

A real-time clock (RTC) is a computer clock that is solely built for keeping time. It is usually in the form of an integrated circuit and keeps count of seconds, minutes, hours, days, months and years [1]. A Raspberry Pi does not have a real time clock so it retrieves the time from a Network Time Protocol Daemon. In practical 3 the Raspberry Pi will be interfaced with a RTC using I2C and the time will be set using buttons and interrupts.

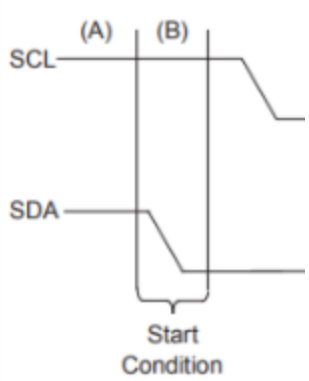
UML use-case diagram of the system



I2C communication using Wiring Pi:

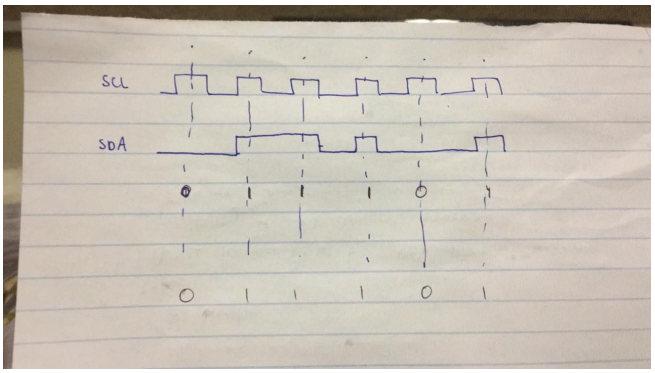


(a) Initialisation



For start condition to take place the SCL needs to be high while the SDA turns low. This is when the initialising takes place and is also why it is so important to use correct capacitors values for your crystal [2].
The command used here is: “int wiringPiSetup (void)” [3].

(b) Send data



Receiving data uses the same timing diagram as Send Data and the same procedure. Here the code used was “int wiringPiI2CWriteReg8 (int fd, int reg, int data)” . With the RTC address, the registry index, and the combination of bits you want to send/write.

(c) Receive Data

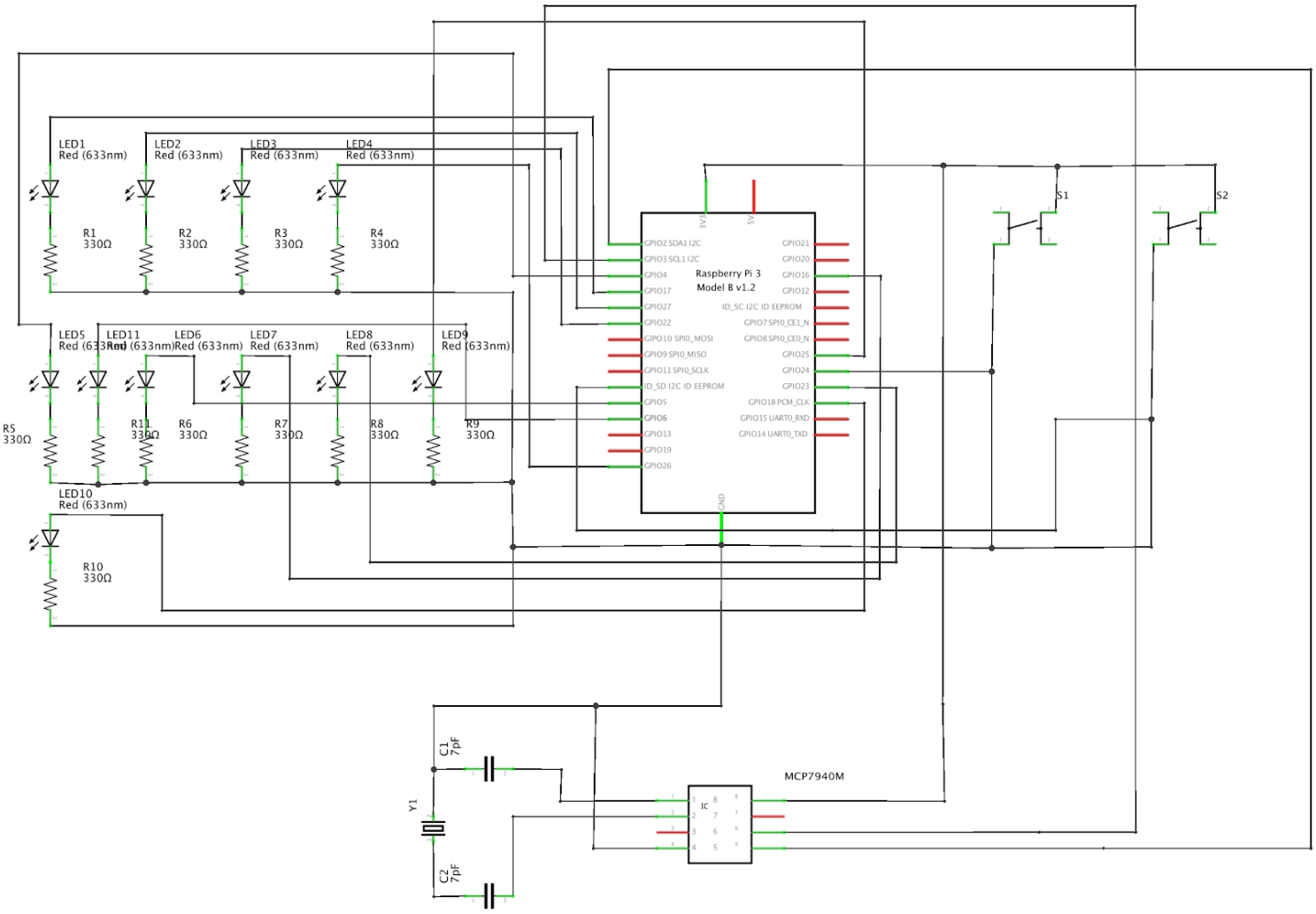
Data is read from the SDA in the middle of a clock pulse. If the SDA is high it is equivalent to a 1 and if it is low it is equivalent to a 0. Thus data gets transported. The following WiringPi function is called to send data “int wiringPiI2CReadReg8 (int fd, int reg)”. As parameters the RTC address and the registry index (hour,min,sec).

Interrupts and debouncing

When a button is pushed the signal from the button oscillates as contact between the metal parts are made. Debouncing is used to ensure the signal is only taken once when pushing the button [4]. An advantage of this is that one button press won’t be called multiple times.

An interrupt is used to stop a process to start another process. This can be used for a button push. When the button is pressed the interrupt will stop the process it is busy with and start the process connected to the button. Interrupts support multi-process multi-taking [5]. An advantage of this is that a program can be running and when a button is pressed it can immediately start running the function related to the button.

Circuit Diagram



References

[1] "What is a Real-Time Clock (RTC)? - Definition from Techopedia"
https://www.techopedia.com/definition/2273/real-time-clock-rtc.

[2] "Basics of the I2C communication protocol "
http://www.circuitbasics.com/basics-of-the-i2c-communication-protocol/

[3] "WiringPi Documentation"
http://wiringpi.com/reference/i2c-library/

[4] "What is debouncing? - Definition from WhatIs.com"
https://whatis.techtarget.com/definition/debouncing.

[5] " What is an Interrupt? - Definition from Techopedia"
https://www.techopedia.com/definition/3373/interrupt-computing.