Minimal Stop Watch

September 26

2019

Student Assessment Cover Sheet

WelTec
Wellington Institute of Technology
Te Whare Wānanga o te Awakairangi

MAIL TO: WelTec, Private Bag 39814, Lower Hutt 5045, New Zealand 0800 935 832 www.weltec.ac.nz

Legal Surname or Family Name(s)
KEEZHUTTU ALEX

Legal First or Given Name(s)
ASHIN

Preferred Name(s)

WelTec Student ID
2 1 9 1 0 5 6

Assessment Title
Assignment1 - Minimal Stop Watch
(Please enter your Assessment Title and Due Date)

Due Date
28/09/2019

School of
ENGINEERING
(Please enter the name of the School your Programme belongs to)

Office Use Only

Programme Code and Title
HV4530 Graduate Diploma in Engineering
(Please enter your Programme code eg HVXXXX)

Course Code and Title (and class if needed)
MG7013 EMBEDDED SYSTEMS
(Please enter your Course code eg IDXXXX)

Tutor
FRANK BEINERSDORF
(Please enter your Tutor's name)

Received on behalf of the School (date stamp)

Declaration:
This assessment is my own original work. I have acknowledged contributions from other sources using the APA reference format.

Student's Signature

Date
27/09/2019

Student to complete this section.
This part of the Assessment Cover sheet is retained by the School as proof that your assessment was handed in.

Assessment Title
Assignment1 - Minimal Stop Watch
(Please enter your Assessment Title and Due Date)

Due Date
28/09/2019

Programme Code and Title
(Please enter your Programme code eg HVXXXX)

Office Use Only

Course Code and Title (and class if needed)
MG7013 EMBEDDED SYSTEMS
(Please enter your Course code eg IDXXXX)

First Name(s) **Surname or Family Name(s)**
ASHIN KEEZHUTTU ALEX

Preferred Name(s)

Received on behalf of the School (date stamp)

WelTec Student ID
2 1 9 1 0 5 6

BPM 3.05/SF020 Student Assessment Cover Sheet August 2013

CONTENTS

INTRODUCTION……..02

OBJECTIVE……02

PROGRAM CONCEPT……02

COMPONENTS USED……..03

CIRCUIT DIAGRAM……..06

FLOW CHART……07

DESIGN PROCEDURE……..07

METHODOLOGY……..08

LIMITATIONS……08

HARDWARE MODEL……08

PROGRAM……..10

RESULT……12

CONCLUSION……12

DATA SHEETS……12

INTRODUCTION

The stopwatch is a hand held timepiece designed to measure the amount of time that elapses from a particular time between it is activated and deactivated. The timing functions are traditionally controlled by two buttons. Pressing the start button starts the timer running, and pressing the button a second time stops it, leaving the elapsed time displayed. A press of the reset button then resets the stopwatch to zero." *(Ref: - "Stop Watch-Wikipedia)*

"The second part of the project focus on reaction time, the reaction time is the interval between the reception of a stimulus and the initiation of the response." *(Ref: - "Reaction Time, Wikipedia)*

OBJECTIVE

The objective of the project is to design and construct a minimal stop watch using the bubble display with a start & stop switch, reset buttons, the teensy 3.2 board and bubble display.

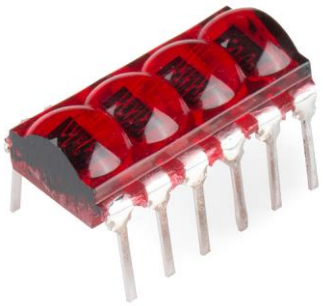
PROGRAM CONCEPT

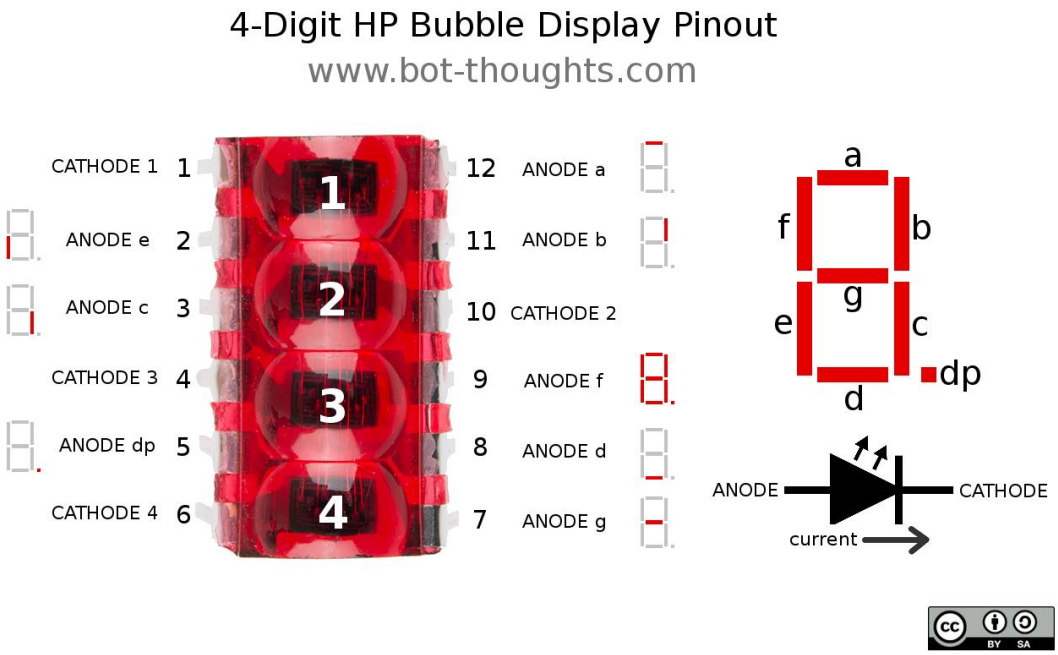
The main part of this project ids to write the program using Version Studio Code as per the project requirements which includes:- a start and Stop button, a reset button, and replace the delays with Millis().As the external interrupt is being used in this project, so we have to consider the denouncing time for this project.

The second part of the project is designing a hardware prototype which can be done within a breadboard. As per the designed circuit the connections for the bubble display has to be made on the bread board by connecting 8 output pins of the teensy board to the8 anode pins of the bubble display using an current limiting resistors to protect the LED of the display. The N –channel Enhancement type MOSFET is used to switch the cathode corresponding to each digits of the bubble display controlled by the pull-down (to keep the MOSFET OFF during start up) resistors and inputs from the teensy 3.2 development board.

COMPONENTS USED

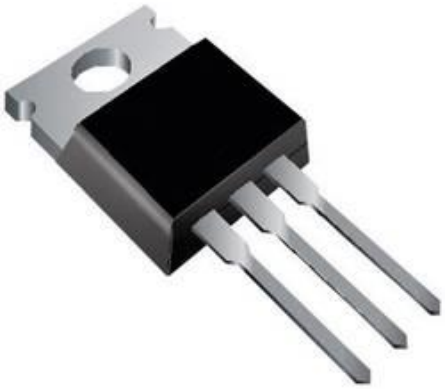
- BUBBLE DISPLAY (QDSP-6064)

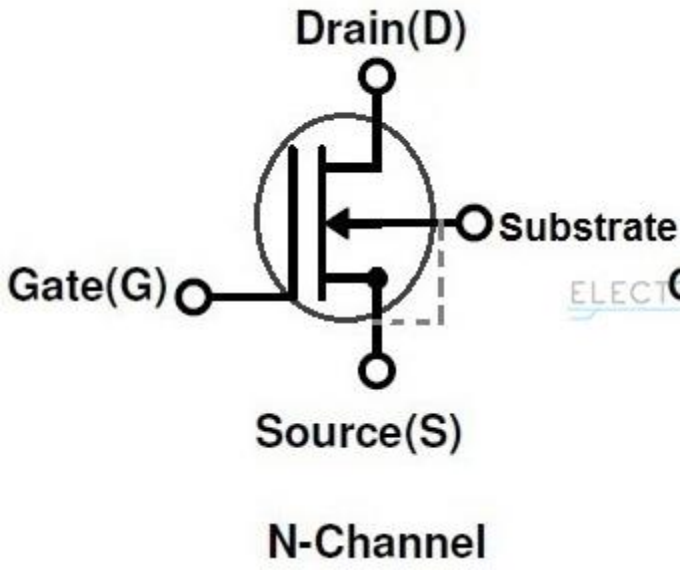




The bubble display is used to display 4 digits at a time. Here the bubble display is the combination of 4- 7 segment display with common cathode type which can program according to what we want to display The brightness of the display can be adjusted during the programming

- MOSFET (IRLU8743)





The MOSFET is a voltage controlled field effect transistor that has oxide gate which insulated from the semiconductor. Here a pull-down resistor is used to avoid the by limiting the current due to the high capacitive gate which can draw large amount of current when turns ON

- RESISTOR

The limiting resistors and Pull-down are used here to protect the device and Avoid Floating of MOSFET.

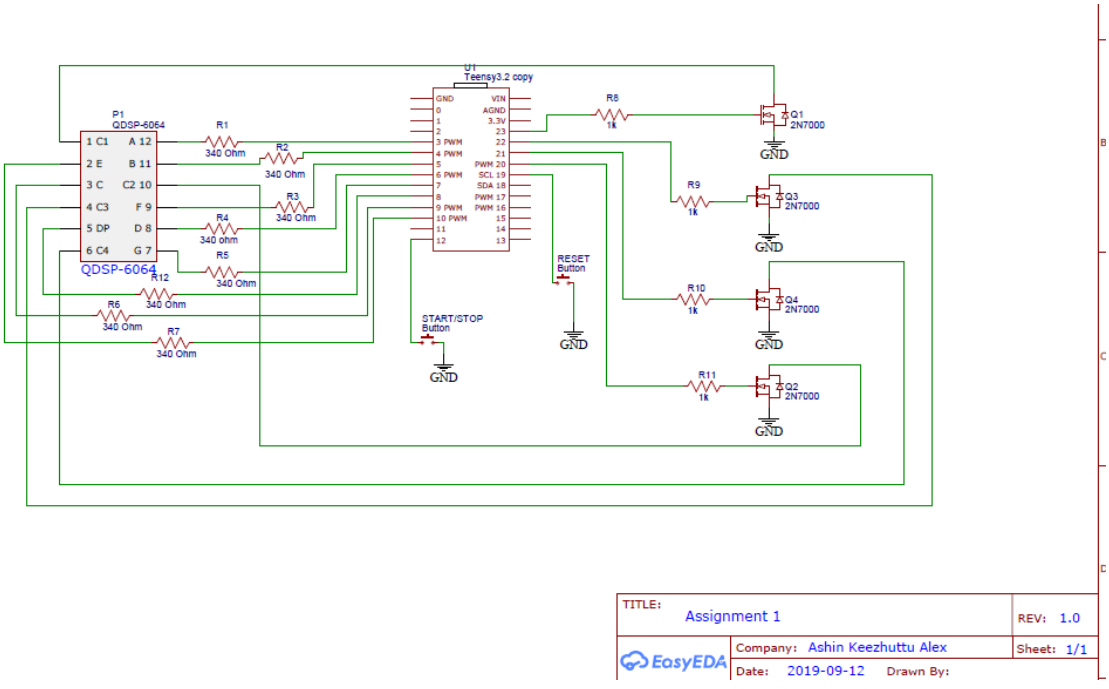
- PUSH BUTTONS – 2Nos

The Push buttons are used for Start / Stop and Reset

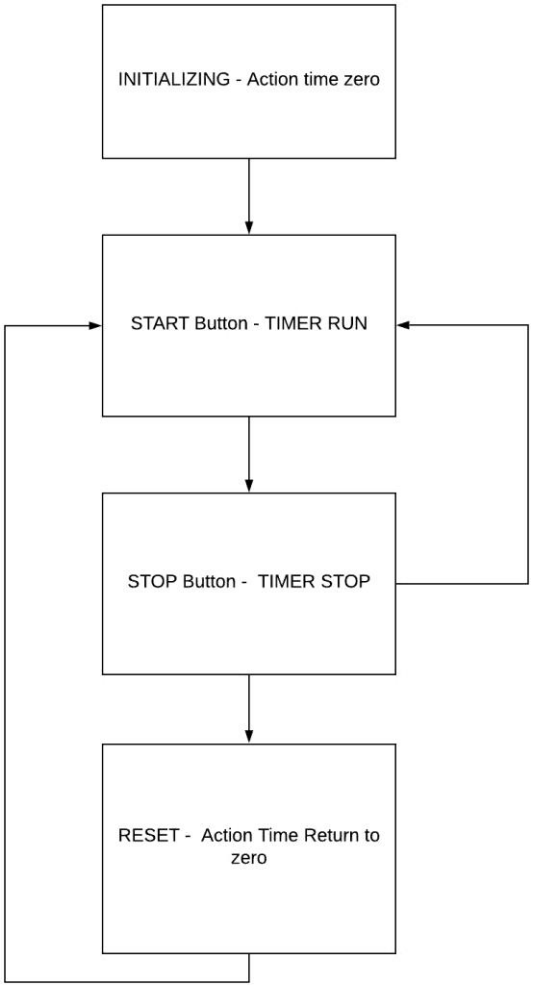
DEVELOPMENT BOARD

Teensy 3.2

CIRCUIT DIAGRAM



FLOW CHART



DESIGN PROCEDURE

- Power up the teensy 3.2.
- The value of the resistor is determined by adjusting the output current in the range of mA. Here the Voltage is 3.3V and the maximum current will give the value of the resistor as 340 Ohm. Hence the above resistor will limit the current to the led in the display
- Pull-down resistor of 10Kohm is connected to MOSFET to avoid floating of the MOSFET
- The Push buttons are used to control Start or Stop and reset of the timer.

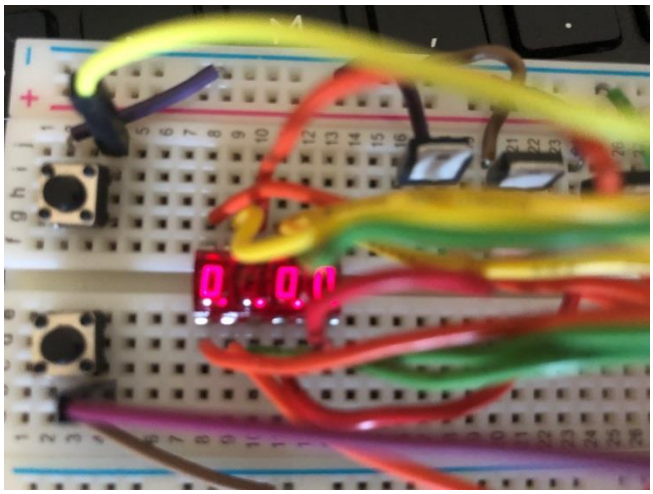
METHODOLOGY

- Design a concept to implement the project objective
- Design the circuit for the hardware using the provided components to achieve the task
- Using Version Studio code, write the program using the ''SevnSeg'' library to implement the required task
- Upload the program to Teensy 3.2 development board
- Make sure that the minimal stop watch is function properly using Start/Stop Switch and reset button

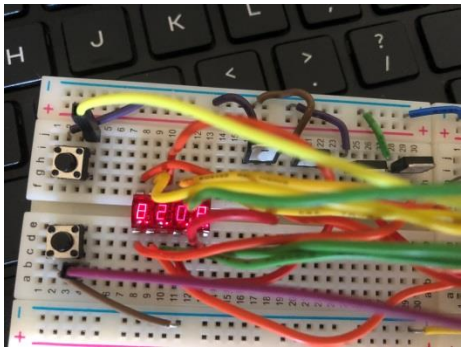
LIMITATIONS

- The limitation of the project is there some problem with debouncing the buttons
- The timer sometimes stops without any error
- The second display goes intermittently blank once reset button is pressed once the start button is pressed it starts displaying.

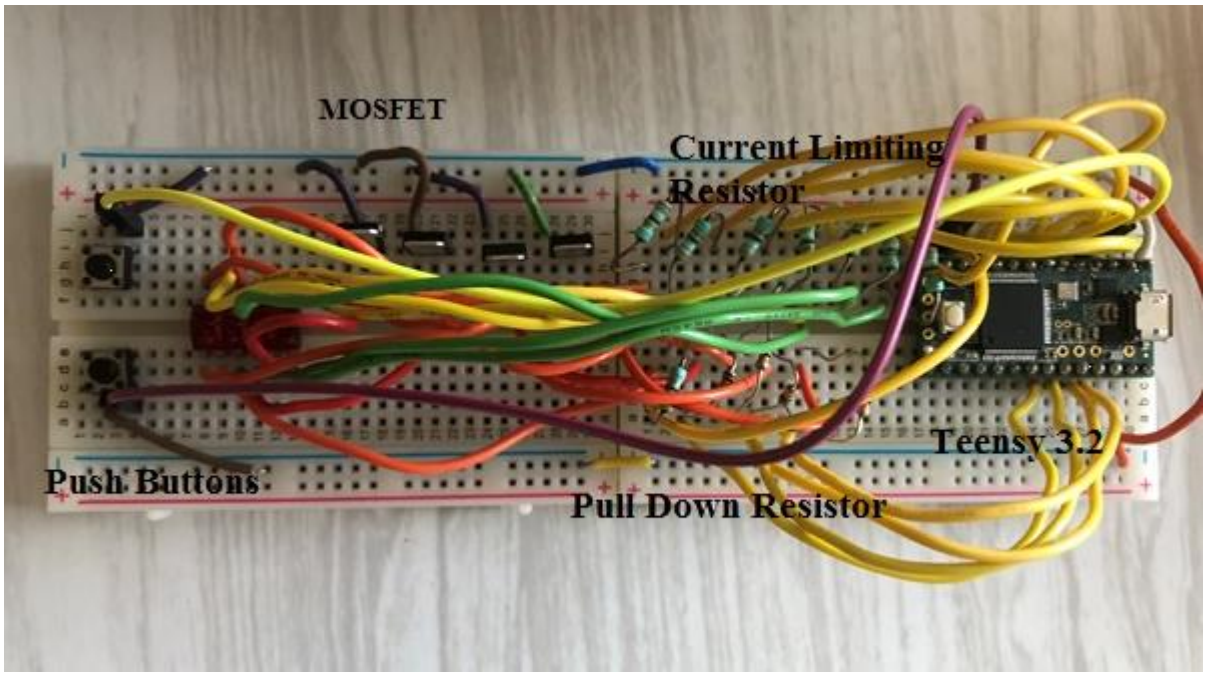
When Reset



When Restart



HARDWARE MODEL



PROGRAM

```
#include <Arduino.h>
#include "SevSeg.h"

u_int8_t Reset_Button = 2;// Reset Interupt button pin
u_int8_t  StartStop_Button = 11;// Start or Stop Interupt button pin

// Global Variables
int Previous_StartStopButton= 0;
int flag=0;
int Previous_ResetButton= 0;
unsigned long timer;
int deciSecond=0;
void StartStop_Interrupt();
void Reset_Interrupt();

// An instance of object
SevSeg myDisplay;

void setup() {

int displayType = COMMON_CATHODE; //display is common cathode

//The pins that are connected to the mosfet gate pins that are sink from catho
de to ground

 int digit1 = 21; //Pin 1
 int digit2 = 23; //Pin 10
 int digit3 = 22; //Pin 4
 int digit4 = 20; //Pin 6
  // put your setup code here, to run once:

  // Pins connected to anode
 int segA = 3; //Pin 12
 int segB = 4; //Pin 11
 int segC = 9; //Pin 3
 int segD = 6; //Pin 8
 int segE = 10; //Pin 2
 int segF = 5; //Pin 9
 int segG = 7; //Pin 7
 int segDP= 8; //Pin 5

 int numberOfDigits = 4 ;// 4 digit display

myDisplay.Begin(displayType, numberOfDigits, digit1, digit2, digit3, digit4, s
egA,
```

```
segB, segC, segD, segE, segF, segG, segDP);

 myDisplay.SetBrightness(100); // set brightness level to 100%

 timer = millis(); // timer variable store the time the microcontroller start
working

 pinMode(2, INPUT_PULLUP); // Assigned  pullup resistor attached to the button
 pinMode(11, INPUT_PULLUP);//Assigned  pullup resistor attached to the button

// Run the interupt pin start/stop function when pin goes low
 attachInterrupt
(digitalPinToInterrupt(StartStop_Button ), StartStop_Interrupt, FALLING); //In
terrupt activated when Start_Stop_Button pressed

 // Run the interrupt pin start/stop function when pin goes low
 attachInterrupt(digitalPinToInterrupt( Reset_Button), Reset_Interrupt, FALLIN
G); // Interruptactivated when Reset_Button Pressed

}

void loop() {

  // Display the floating point number that the time of stop watch associated
with

  char tempString[10]; //Used for sprintf to store the string form of the time
r
 sprintf(tempString, "%4d", deciSecond); //Convert deciSecond into a string th
at is right adjusted

 myDisplay.DisplayString(tempString, 3); //(numberToDisplay, decimal point loc
ationin binary number [4 means the third digit])


 //Check if 10ms has elapsed
 if (millis() - timer >= 1000 && flag==0)
 {
 timer = millis();// keeps updating the start time point untill start time is
pressed
 deciSecond++;// This just gets the variable going for the loop to handlr next
 time
 }
 }


 void StartStop_Interrupt() // Interrupt activated when Start_Stop_Button pres
sed
```

```
{
 int Previous_StartStopButton = digitalRead(StartStop_Button);// check if the
pushbutton is pressed.
 // if it is, the buttonState is HIGH:

 if (Previous_StartStopButton== LOW && flag==0)
 {

 flag=1;
 }
 else if(Previous_StartStopButton== LOW && flag==1)
 {
 flag=0;
 }
 }

 void Reset_Interrupt() // Interrupt activated when Reset_Button pressed
 {
 int Previous_ResetButton=digitalRead (Reset_Button);

 if (Previous_ResetButton==LOW)

 {

 deciSecond=0;

 }
  }
```

RESULT

Video Link: https://youtu.be/KSmPkJjt17w

CONCLUSION

The minimal stopwatch is developed using Teensy 3.2 development board and Bubble display, Start/Stop button and Reset Push buttons.

DATA SHEET

https://www.alldatasheet.com/datasheet pdf/pdf/1009960/ISC/IRLB8748.html

https://cdn.sparkfun.com/datasheets/Components/LED/BB_QDSP_DS.pdf