B31DG Assignment 1 Report

Table Of Contents

1. Contact Details

Name: Ewan Wills

HW Number: H00373054

Student Email: ew2006@hw.ac.uk

2. Revision History

Version Number	Date	Author Name	Notes
1.0	24/01/25	Ewan Wills	Initial Version
2.0	30/01/25	Ewan Wills	Added oscilloscope section and formatting

3. Calculation of Application Parameters

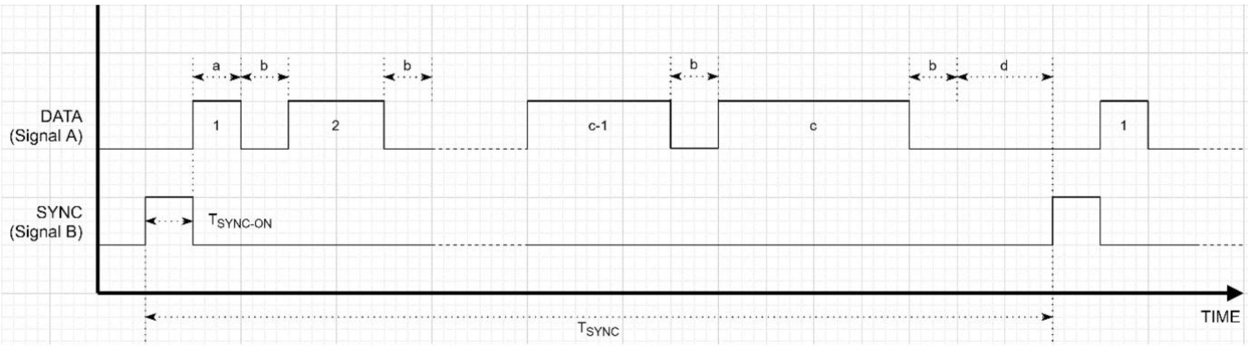

Figure 1 – Output Timing Diagram

With reference to the diagram provided (Figure 1), all of the timing parameters are assigned to either a, b, c, d or $T_{SYNC-ON}$.

Values for a, b, c, and d are determined by the alphanumeric mapping of the students surname. For WILLS: W=4, I=9, L=12, L=12, S=8.

$$W = 4 \ \therefore a = 4 * 100\mu S = 400\mu S$$

$$I = 9 \ \therefore b = 9 * 100\mu S = 900\mu S$$

$$L = 12 \ \therefore c = 12 + 4 \ = 16 \ (pulses)$$

$$L = 12 \ \therefore d = 12 * 500\mu S = 6000\mu S$$

The alphanumerical value for S is used for determine the alternative output behaviour of the DATA line.

$$S = 8 \ \therefore Option = 8\%4 + 1 = 1$$

For normal data output, the pulses 1, 2, ..., c (16) increase by a set amount given by the calculation:

$$T_{ON(n)} = a + \left((n-1) * 50\mu S\right), \qquad for\ n > 0$$

This is implemented in code using a **for** loop. This is for PRODUCTION mode. In DEBUG mode DataPeriodOn1 will be multiplied by 1000 prior to the calculations, therefore

$$DataPeriodOn[n] = \mathrm{DataPeriodOn1}\ + \left((n-1) * 50 * 10^{-3}\right)$$

```
//calculate all DataPeriodOn's for NumPulses for PRODUCTION
for (int n=1; n<NumPulses; n++){
  DataPeriodOn[n] = DataPeriodOn1 + (n-1)*(50*(10^-6));
}
```

Figure 2 - DataPeriodOn(n) loop

Figure 3 shows the initialisation of NumPulses (C) and the DataPeriodOn list. PRODUCTION is a Boolean value to implement a conditional code-compile.

```
//define if program will compile in production or debug
#define PRODUCTION false

const int NumPulses = 16;

int DataPeriodOn[NumPulses+1];
```

Figure 3 – Application parameters A

Figure 4 shows the assignment of values for PRODUCTION and DEBUG mode. All values are in microseconds so that they can all be integers.

```
//all values in microseconds
if (PRODUCTION){
    DataPeriodOff = 900; //900uS
    DataPeriodOffFinal = 6000; //6mS
    DataPeriodOn1 = 400; //400uS
    SyncPeriodOn = 50;//50uS
    //calculate all DataPeriodOn's for NumPulses for PRODUCTION
    for (int n=1; n<=NumPulses; n++){
        DataPeriodOn[n] = DataPeriodOn1 + (n-1)*(50);
    }

}else{
    DataPeriodOff = 900*1000; //900mS - b
    DataPeriodOffFinal = 6000*1000; //6S - d
    DataPeriodOn1 = 400*1000;//400mS - a
    SyncPeriodOn = 50*1000;//50mS
    //calculate all DataPeriodOn's for NumPulses *1000 for DEBUG
    for (int n=1; n<=NumPulses; n++){
        DataPeriodOn[n] = DataPeriodOn1 + (n-1)*(50*(1000));
    }
}
```

Figure 4 – Application parameters B

4. Oscilloscope Screen Captures

a. Oscilloscope Images

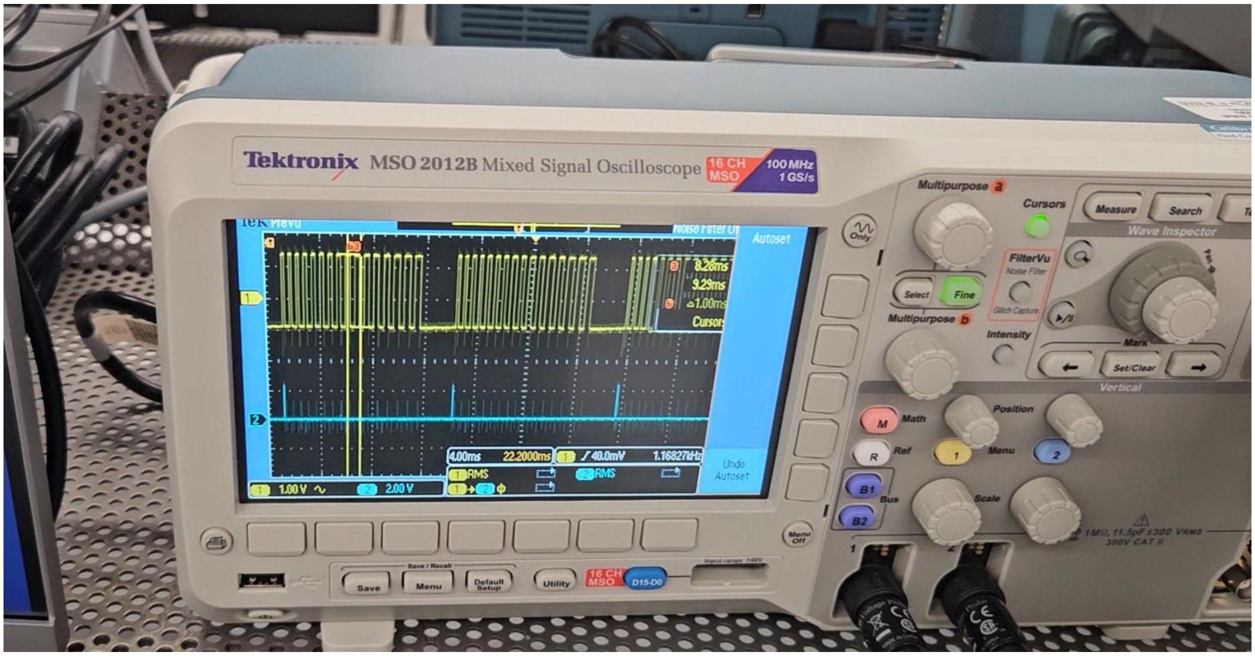

Figure 5 - Oscilloscope capture 1

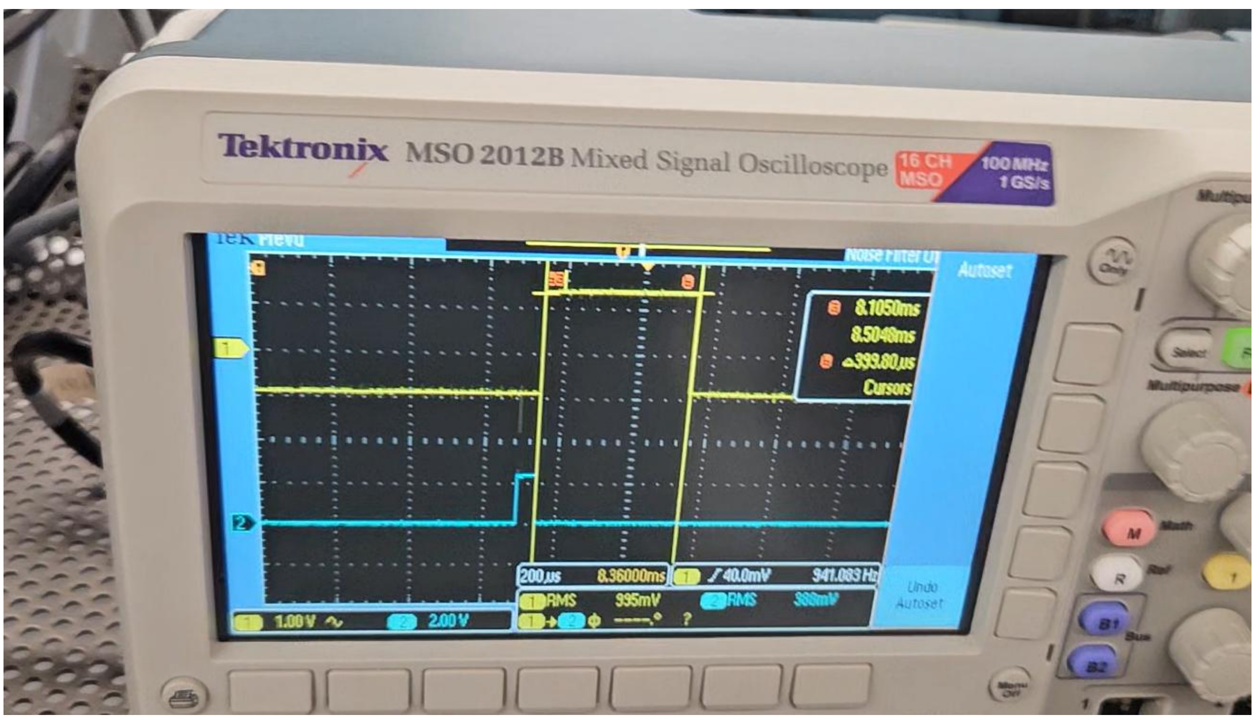

Figure 6 - Oscilloscope capture 2

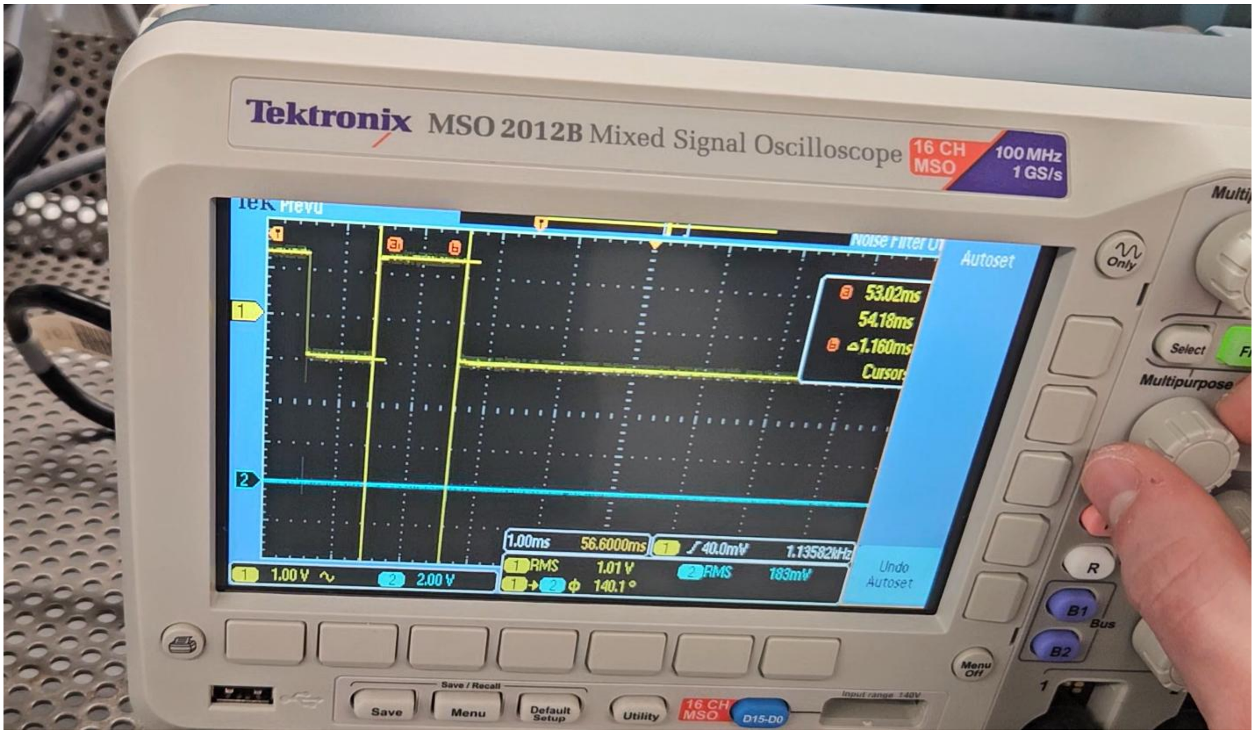

Figure 7 - Oscilloscope capture 3

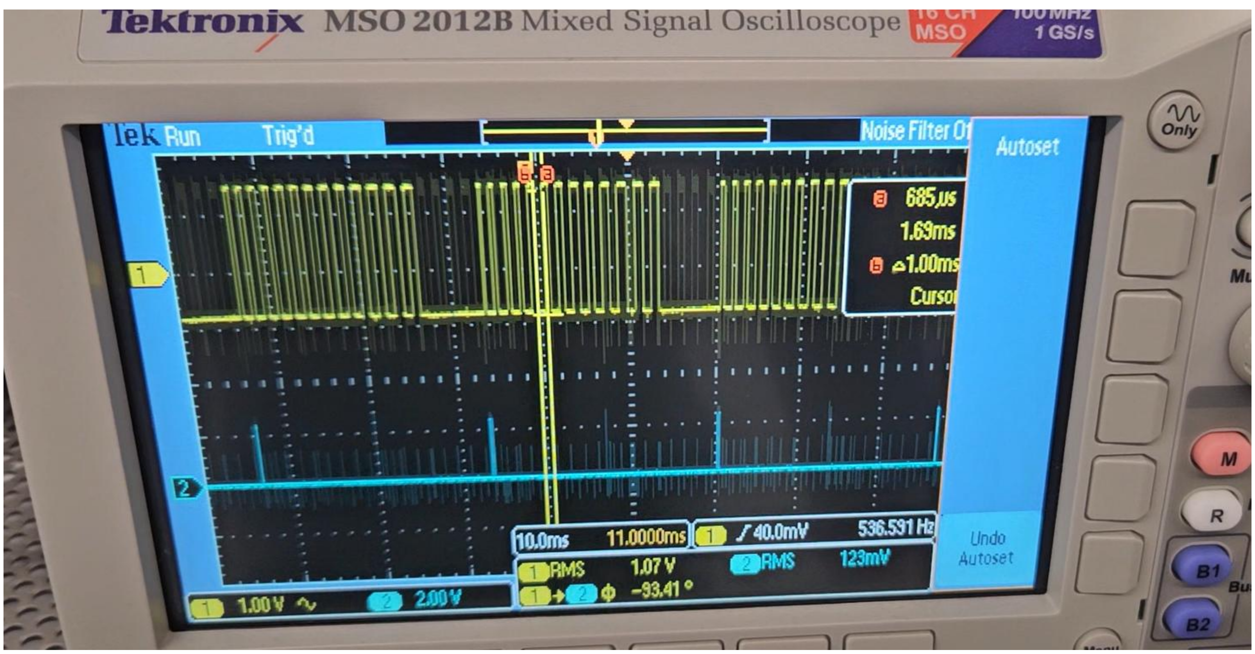

Figure 8 - Oscilloscope capture 4

Figure 9 - Oscilloscope capture 5

Figure 5 shows output_enable true output_select false. There are 16 pulses.

Figure 6 shows a close up of the first pulse in the sequence. First pulse is shown to be 400uS. Figure 7 shows the final pulse (pulse 16) in the sequence and it is shown to be 1150uS.

Figure 8 shows where output_select is true. This removes the final 3 pulses, resulting in 13 pulses in the sequence. Figure 9 shows output_enable false. Here the data line is always low but the timing for sync line is still maintained.

Figure 5 and figure 8 both show at least two Sync Pulses.

b. Oscilloscope video

Video showing operation in PRODUCTION mode with oscilloscope is uploaded unlisted on YouTube at: https://youtu.be/6riSa781030

5. Hardware Circuit

a. Hardware Images

No resistors are required due to pullup resistors within esp32. The pullup resistors can be used when setting the pin modes during setup.

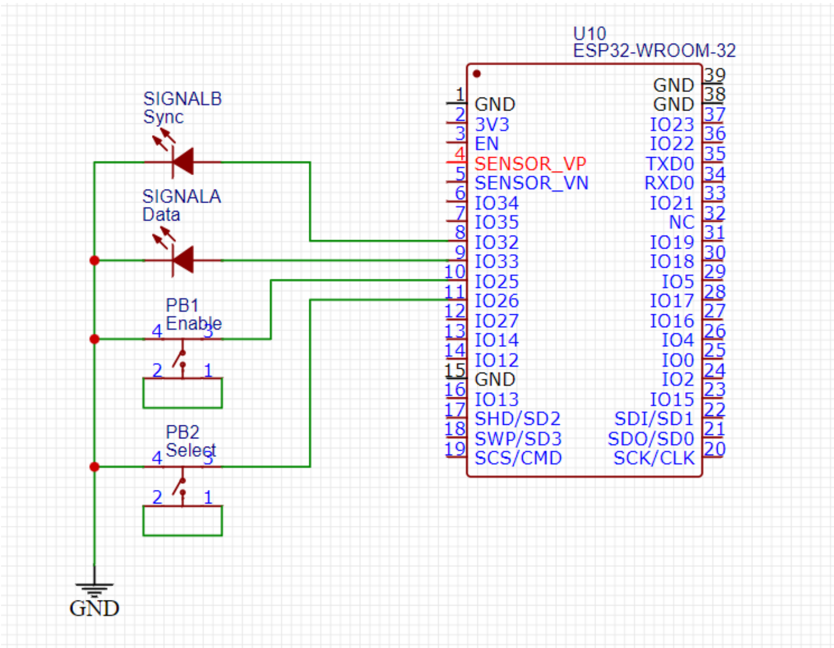

Figure 10 - Hardware Schematic

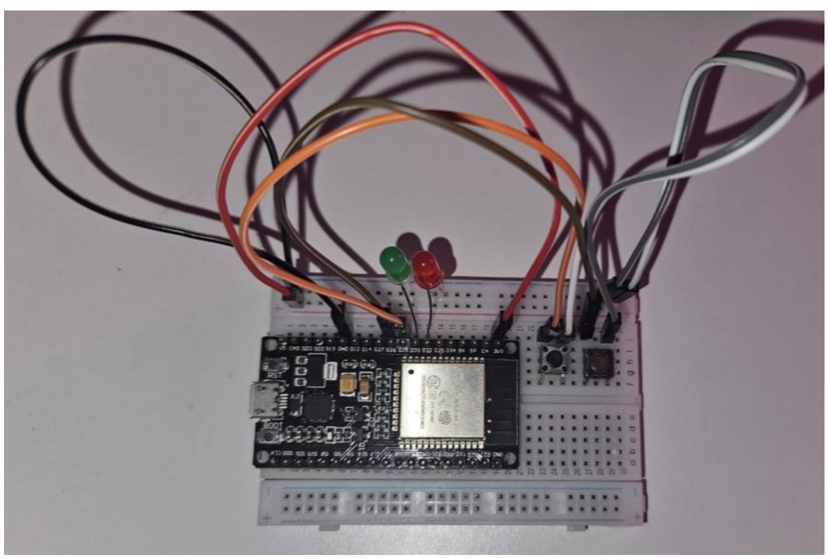

Figure 11 - Physical Hardware Circuit

b. Hardware Demo Video

Video showing operation in DEMO mode is uploaded unlisted on YouTube at:
https://youtu.be/NMU1bU0y8ls

6. Flowchart

a. Flowchart

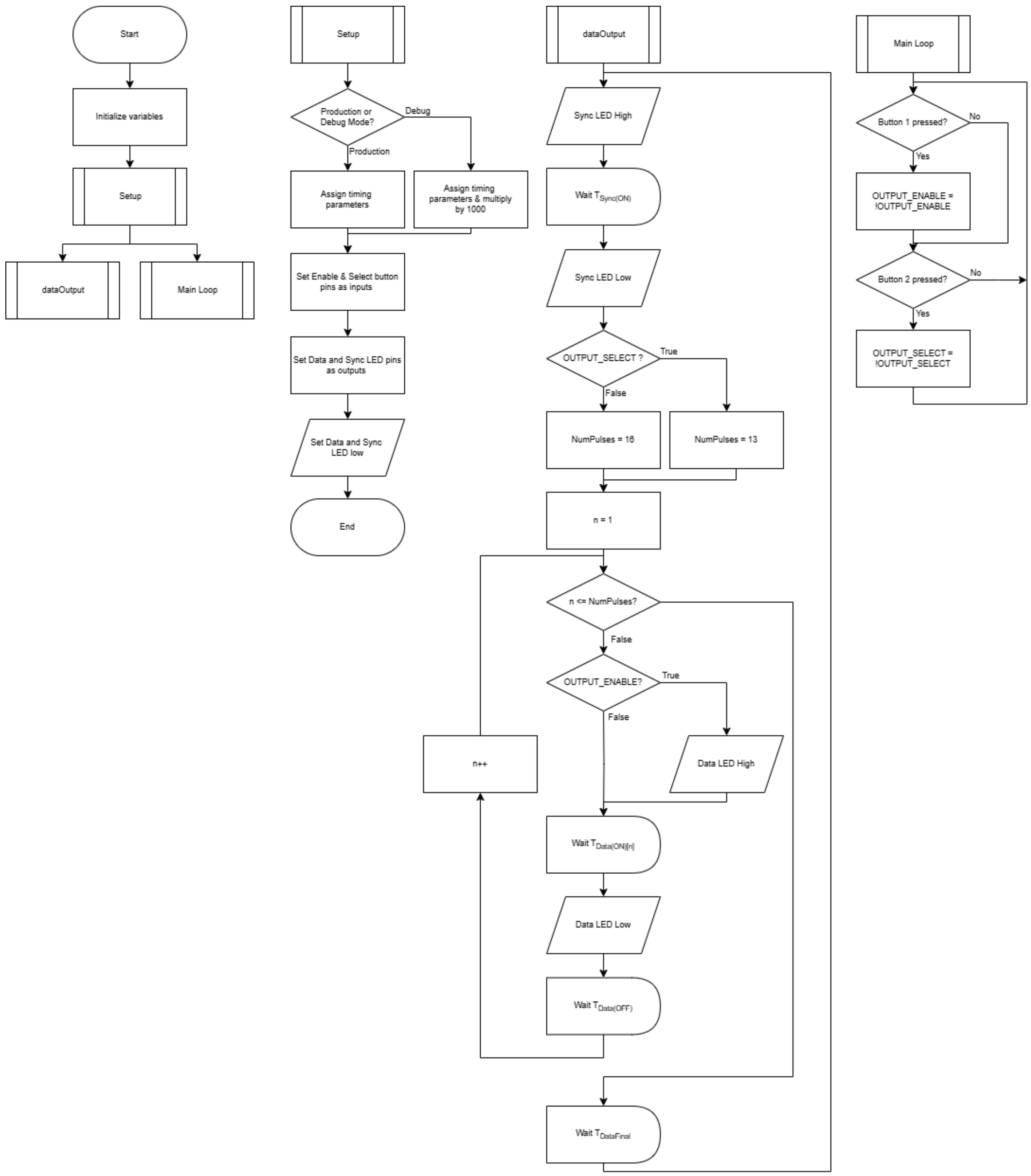

Figure 12 - Application Control Flow Flowchart

b. Nassi Shneiderman diagram

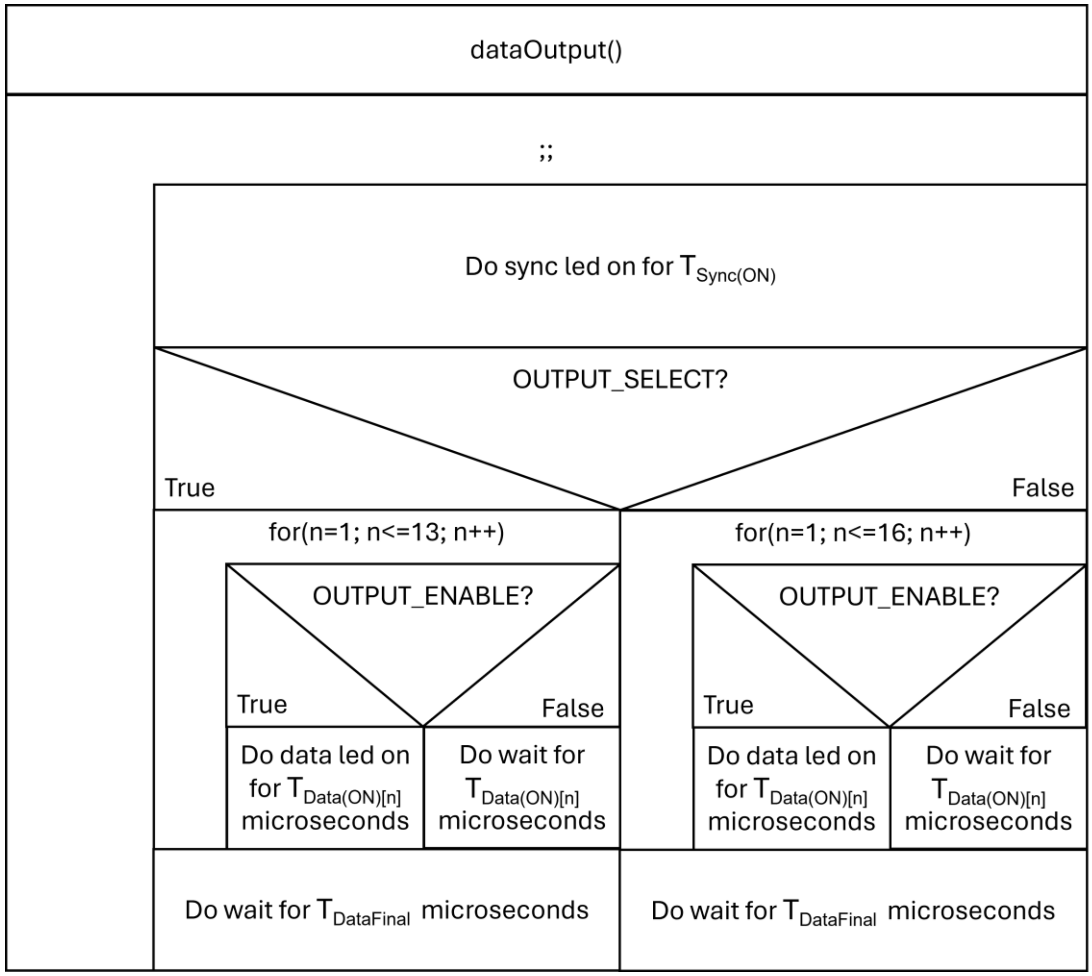

Figure 13 - Nassi Shneiderman diagram

7. Table of Figures