2DX4: Microprocessor Systems Project
Final Project 2021

Instructors: Dr. Boyle, Dr.Haddara and Dr. Shirani

Vaibhav Gogia
L05
MacID: gogiav

As a future member of the engineering profession, the student is responsible for performing the required work in an honest manner, without plagiarism and cheating. Submitting this work with my name and student number is a statement and understanding that this work is my own and adheres to the Academic Integrity Policy of McMaster University and the Code of Conduct of the Professional Engineers of Ontario.

Links to the Project Videos

Project Demo Video or
https://drive.google.com/file/d/1MGzFx0Z4EA7hBhiPz44pxFv453sTJnSD/view?usp=sharing

Note: The demo video itself was also added to the final .zip because google drive had issues processing this one video.

Question 1 response or
https://drive.google.com/file/d/1jNthZnfgp9Z3USsp5o9XfejHH9ar8NcX/view?usp=sharing

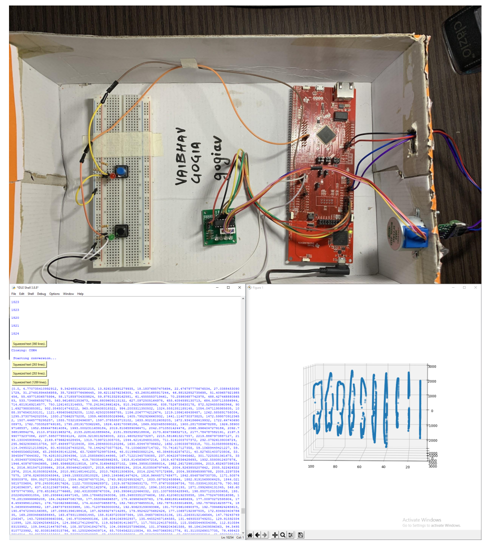
Question 2 response or
https://drive.google.com/file/d/1jJC5Gab05pTYtM1pqAYDpGLibOLSZdGR/view?usp=sharing

Question 3 response or
https://drive.google.com/file/d/15-J9jn8omXbdZzRIIgb88582RoY4qQEg/view?usp=sharing

1. Device Overview:

a) Features

This 360° 3D scanner device was built using a MSP432E401Y microcontroller with a Cortex MF4 as the processor (built into the microcontroller). The microcontroller combines with a VL53L1X time-of-flight (ToF) module and a stepper motor (MOT-28BYJ48) to function. The cost of building this device was C$179.99.



The specifications are listed below:

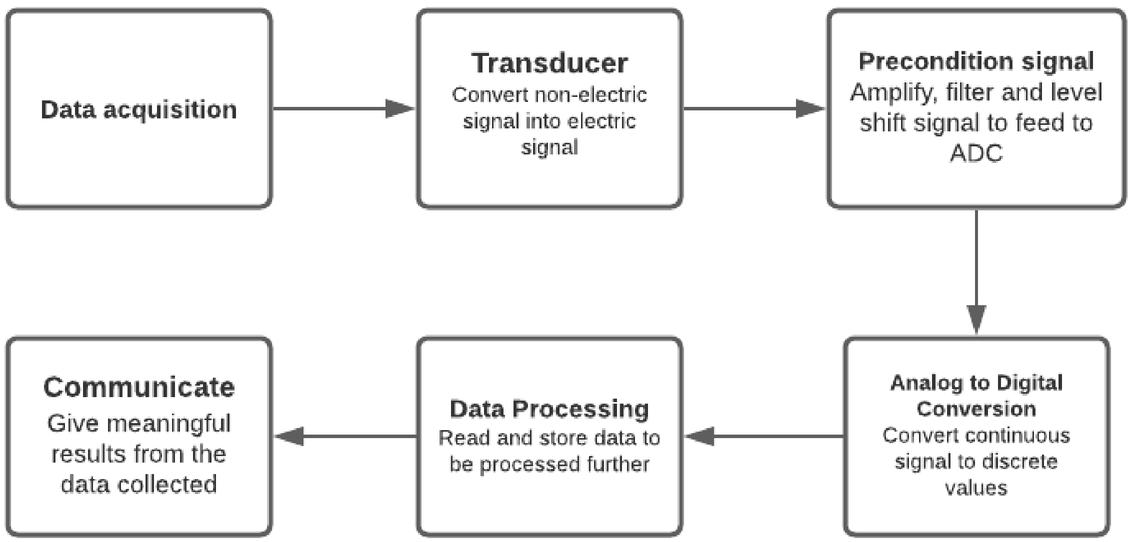
- Default bus speed = 120 MHz, brought down to 30 MHz for functioning of the device)
- 1024KB of flash memory
- 256KB of SRAM
- 6KB EEPROM
- Operating voltage = 2.5 - 5.5 V
- 2x 12-bit SAR ADC modules
- 8x UARTs, each having a baud rate of upto 7.5 MBPS (regular speed) and 15 MBPS (high speed)
- The ToF module can measure distances up to 4000mm away at a frequency of up to 50 MHz with an operational voltage of 2.6 - 3.5V.

b) General Description

The basic function of the device is to 3D map indoor settings, such as a hallway. The device is intended to be used in a variety of different systems. A breakdown of its core components is listed below:

- Momentary push button: Digital I/O used to control data acquisition.
- LEDs: Get status of each distance measurement.
- ToF sensor: Measure distance in the Y-Z plane.
- Stepper Motor: Used to facilitate the 360° rotation of the ToF sensor to get measurements.
- PC: The data is collected to be sent to a PC for further processing.
- Open3D/Matplotlib: Used for 3D visualization of the data collected.
- Python and C: Programming languages used for the building of this device.

c) **Block Diagram (Data flow graph)**



2. Device Characteristics Table:

Component	Detail
Motor Driving pins	PE0-PE3
SCL/SDA Pins	PB2/PB3
Ext. push button pin	PM0
Bus Speed	30 MHz
Communication Port	COM4
Communication Speed	115.2 kbps
CCW Rotation	PL0

3. Detailed Description:

a) Distance Measurement

The VL53LIX is the module used for distance measurement in this device.It uses a stepper motor to provide a 360-degree measurement of distance within a single vertical geometric plane. Collecting distance measurement and processing it to provide useful outputs usually requires three components: 1. A transducer, it is used to interpret non-electric signals as electric signals. 2. A pre-conditioning circuit, used to filter the said electric signals to make them easier to convert into a digital form. 3. An Analog-to-Digital Converter (ADC), to discretize the conditioned analog signals into a digital form.

The VL53L1X ToF sensor emits pulses of infrared light and determines the distance to objects by measuring the amount of time the pulse is “in-flight” before being detected. A general equation for ToF sensors is D= $(t/2)\cdot c$, where D is the measured distance, c is the speed of light and tis the travel time of the emitted light in seconds. The sensor calculates the distance by converting the measured time value from analog to digital and then passing it through that equation while using its configuration settings to check if the final distance value is out of range or otherwise incorrect. It sends this data to the microcontroller via I2C.

The process of acquisition of distance measurements to the visualization of this very data on the computer is depicted in the flowchart attached at the bottom of this document. Before the visualization part, the data acquired has to be converted into XYZ coordinates. The X values remain the same (since it is the forward distance). However, the Y and Z values are calculated in the python code using the following formulas:

$$y = \frac{distance}{1000} * \sin(angle)$$

$$z = \frac{distance}{1000} * \cos(angle)$$

b) Visualization

The visualization was performed using a WINDOWS 10 Personal Computer, running an Intel i5 processor. The programming language used was Python 3.8.8 with the following libraries:

- serial
- math
- numpy
- Matplotlib
- Axes3D

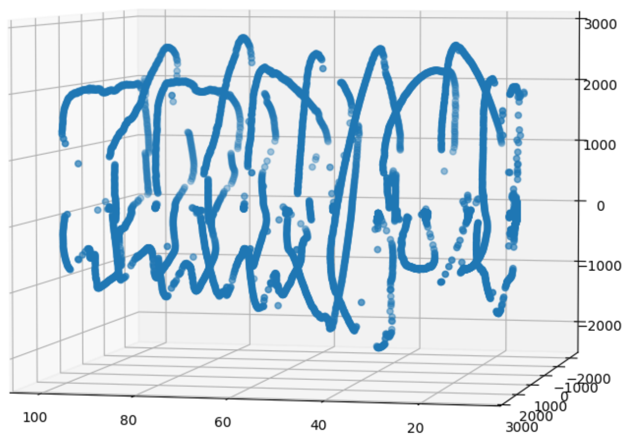


Figure: A 3D visualization using matplotlib.

The distance data is converted into[X, Y, Z] values, as mentioned in the section above, this data is then stored into three separate arrays. A total of 10 measurements were taken with a constant delta value of 10 cm. A point cloud was generated using these measurements.
Each plane had a total of 512 data points. Since the stepper motor rotates a complete 360-degree, this meant that the ToF was making 512/360 = 1.42 measurements per degree of rotation.
After processing this data, to achieve this 3-dimensional visualization, the following functions from the imported python libraries were used:

```
fig = plt.figure()
ax = fig.add_subplot(111, projection = '3d')
ax.scatter(x,y,z)
plt.show()
```

4. Application Example

For the following application to work correctly, Python should be set up on the user's computer and the packages 'serial' and 'matplotlib' should be correctly installed.
The following example uses Python 3.8.8
The example is performed with default settings, which are, 512 data points per rotation, 10 total number of samples, and a constant delta value of 10 cm.

Note: The XYZ plane is defined in the following manner: YZ plane for the vertical slices and X for the forward movement

1. The schematic diagram provided at the end of this document must be followed to set up the device.
2. Use a micro-USB cable to connect the microcontroller to the Computer's USB port. (The micro-USB connects to the XDS 110 debugger unit, shown in the top right corner of the image below.)

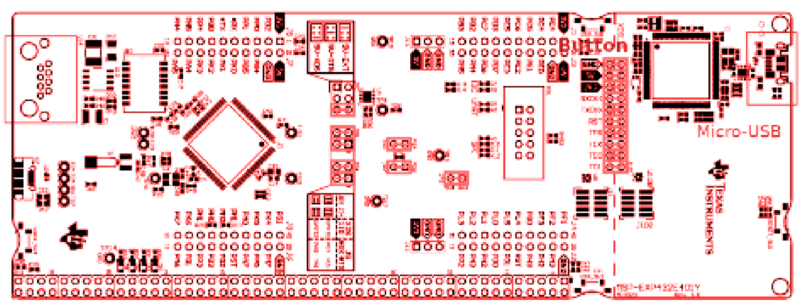


Figure: MSP432E401Y uC Board

3. After this connection is made, all 4 of the on-board LEDs should start blinking which indicates that the setup process has been correctly initialized.
4. Check which COM port is to be used. This is the port the computer has assigned to the device.
   The following command can be used to determine the Ports available:
   Python -m serial.tools.list_ports -v

```
C:\Users\doyle>python -m serial.tools.list_ports -v
COM3
    desc: USB Serial Device (COM3)
    hwid: USB VID:PID=0451:BEF3 SER=6 LOCATION=1-2:x.0
COM4
    desc: USB Serial Device (COM4)
    hwid: USB VID:PID=0451:BEF3 SER=6 LOCATION=1-2:x.3
2 ports found
```

Figure: Port 4 was used for this example

5. Press the on-board button on the MSP432E401Y in use and run the python file plot.py.
    - Press the Start/Stop button for the device to stop operating.

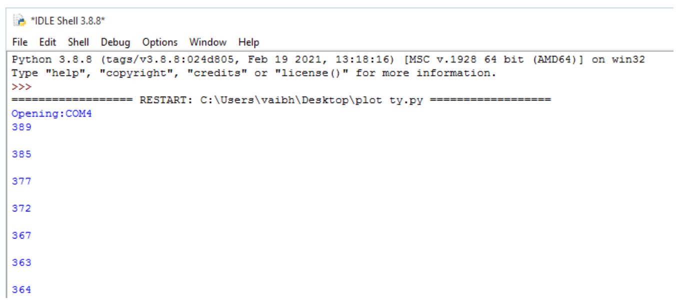


Figure: The ToF will start measuring and sending the distance data to the PC.

6. Once this sample is complete, Move the device 10cm ahead and press the Start/Stop button to start another sample. Use the other press button, if needed, to untangle the cables.
7. Repeat this process 10 times and at the end of which the data acquired would be automatically converted into XYZ coordinates and displayed on the computer screen.

Figure: Data being processed for visualisation

8. After the data is processed completely (should take just a few seconds), the user should see a 3D plot.

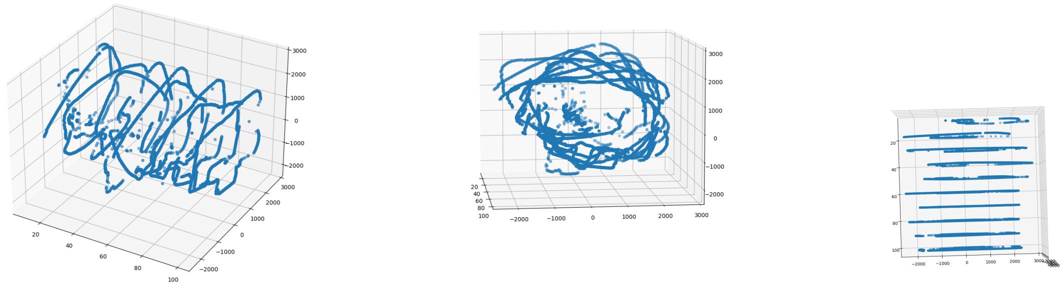


Figure: The normal view, front view, and the top view of the plot.

5. Limitations

The following limitations were noted for the various components used in this device:

- The stepper motor 28BY-J48 experienced severe overheating issues whenever more than 5 samples were measured.
- The VL53L1X ToF sensor was unable to take readings whenever facing a reflective surface.
- The ToF sensor also seemed to be less accurate when the light in the room was too bright. A time during late evening was hence chosen to avoid these disturbances.
- The resolution for the ToF is 1 mm and the maximum quantization error is 2.6/2^16 = 3.97e-5
- The MSP432E401Y has a Floating-Point Unit (FPU) that can support single-precision (32-bit) addition, subtraction, multiplication, division, multiply-and-accumulate, and square root operations. The FPU can also perform conversions between fixed-point floating-point data formats, and floating-point constant instructions. Trigonometric functions from the math.h library can be applied on float variables. Because the FPU is only 32-bits wide, 64-bit (double precision) operations will take more than 1 instruction to complete as the FPU must split it up into 32-bit words.
- The maximum standard serial communication rate that can be implemented with the PC is 128000 bits per second.
- The Nyquist Rate must be at least double the frequency of the bus speed, which is 80MHz. Therefore, the Nyquist Rate is equal to 160MHz.A higher sampling rate may increase the clarity of the signal read at the cost of more power/memory.

6. Circuit Schematic

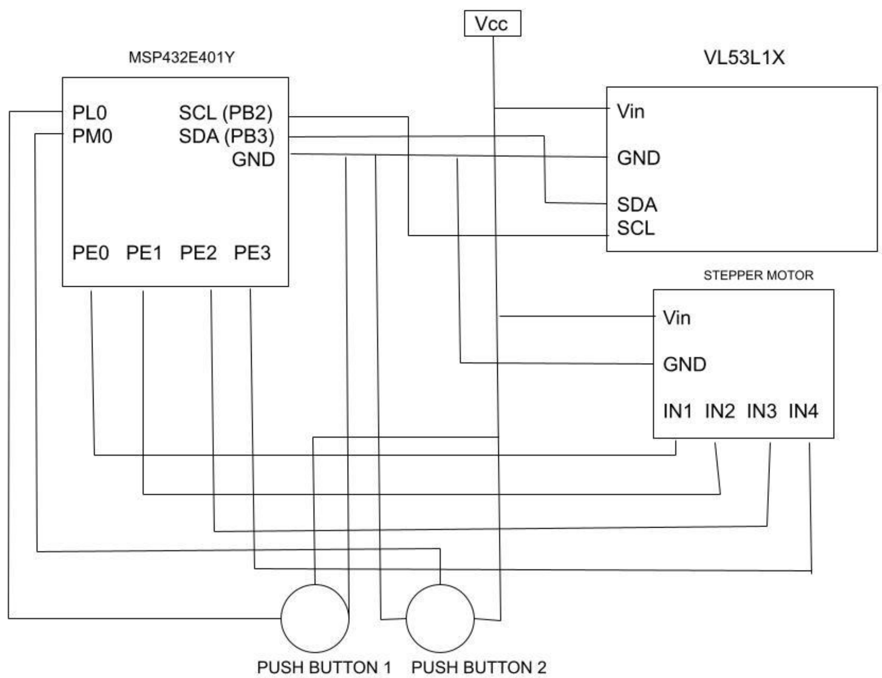


Figure: The circuit schematic

7. Logic Flowcharts

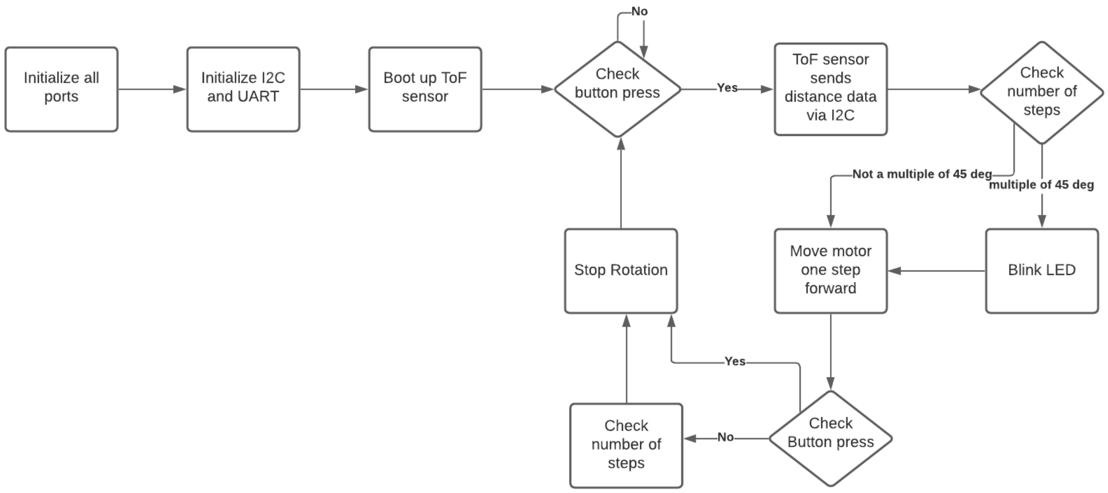


Figure: Microcontroller flowchart

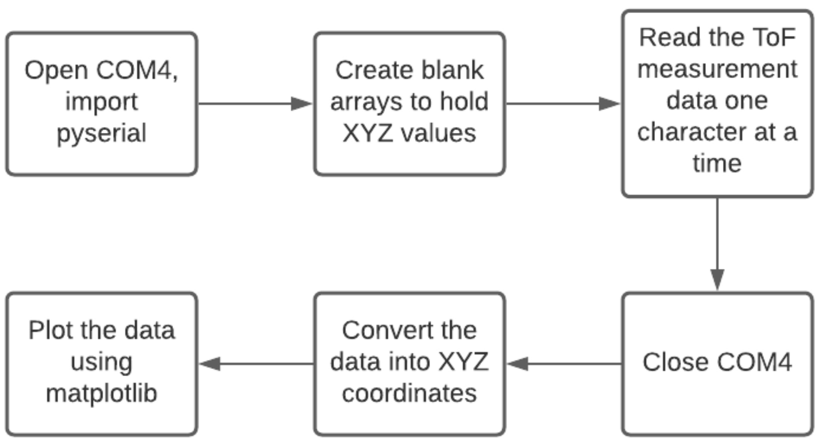


Figure: Python code flowchart