Digital Pen - My Ideation

Narasimhan, EE20B087

Recap of the Problem Statement:

The aim of our project is to design a digital pen that can sense the motion of the nib and store whatever is written on the surface in a digital format. We must be able to read and edit the stored data anytime.

The various ideations done on github have sensors inside the pen and sensors outside as well. Factors such as size and practical usability must be taken into consideration while doing the ideation.

Some earlier ideations include sensors like CMOS optical mouse sensor, IMU sensor, Pressure sensor, ToF sensor, IR and Ultrasonic sensor.

The idea behind this task is to choose the best combination of sensors by carefully weighing the pros and cons. Also, we have to ideate on the components we want to use in the project.

A Mix of Sensors:

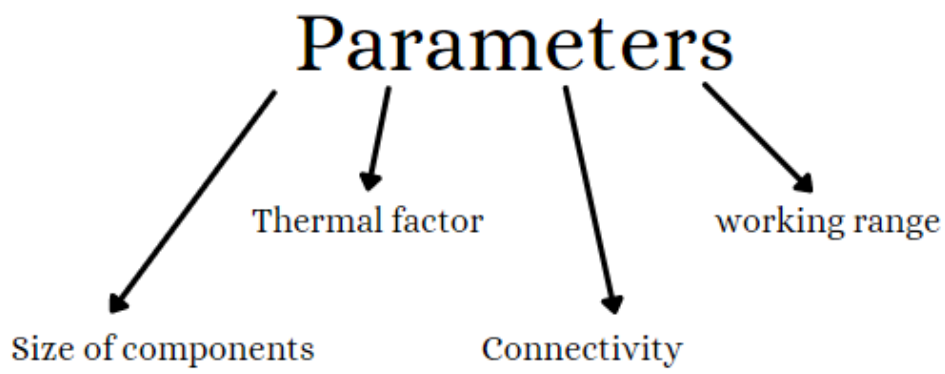
I feel that a number of sensors inside and outside the pen will be the best way to implement the project. The sensors

- inside the pen will be responsible for determining whether the user is writing or not, based on common movements that occur when a person is writing. These sensors must also capture data like acceleration and speed.
- outside the pen will be used to determine the location of the tip of the pen on the paper. This data will help in storing whatever the user is writing.

The choice of components depends on a few parameters:

- ➢ Size: sensors that will be used inside the pen must be small enough to fit in the pen. In fact the microcontroller itself must be small to fit the design. Variables like pen diameter, length, weight etc depend upon size of components.
- ➢ Thermal factor: continuous usage of the pen can heat various components like sensors, power supplying source or microcontroller. Extensive heating can burn out ICs present in the sensors. Hence, the position of components within the pen is crucial. The aim is to optimise the rate of heat transfer to the surroundings.
  Thermal factor decides the material that makes up the framework of the pen.
- ➢ Connectivity: A sound connection between all components is necessary in all systems. In the digital pen project, we need a proper method of connection between the components inside and components outside the pen. Thus, connectivity and compatibility becomes a parameter.
- ➢ Last but not the least, the working range of different sensors should match our project requirements.

There could be a few more minute parameters that could crop up. Those will be discussed there itself.



Sensors inside the pen:

Pressure sensor/force sensor:

- ➔ While writing, the user might do a lot of movements with the pen, knowingly or unknowingly. Some of these might not be involved in actual writing. The main aim of the sensors inside the pen is to distinguish between movements that are behind actual writing and movements that are not.
- ➔ For this we can use pressure sensor/CMOS sensor and IMU sensor.
  What are the movements are we looking for? We want the nib location to be determined only when the user writes.
- ➔ Pressure on the nib changes when the user is writing. This is the change detected by the pressure sensor. However , pressure sensors are expensive and also do not fit the design of our project.
- ➔ I feel that we can use simple force sensors instead of pressure sensors. They might not be significantly cheaper than pressure sensors, but they fit the purpose of our project very well. Their size can be adjusted and their analog output gives an indication of the force exerted on the sensing area (here the force provider must be the nib of the pen).
  https://github.com/Narasimhan07/EC04_Task2/blob/171bc957481350c1066cb1a7c796382d21c5552b/Pressure_force_sensors.md

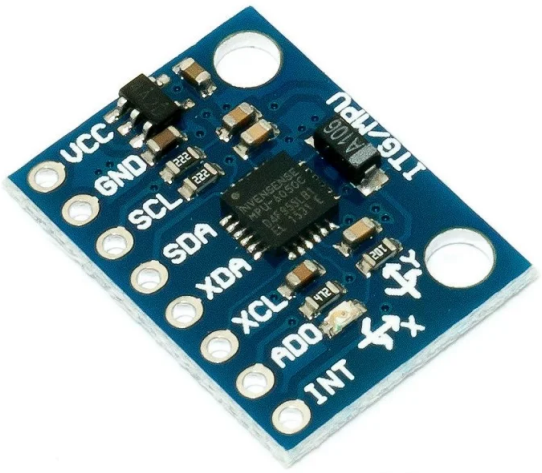
Force sensor

BMP180 Pressure sensor

Arduino Force Sensing Resistor (FSR) - Pi My Life Up - simple project that was the motivation for me to think about using a force sensor.

IMU Sensor:

- The IMU sensor is to keep track of the acceleration and speed of the pen. We require this sensor to determine any random/rash movements in the pen. These movements may be made by the user without intending to do so. These must not be included as writing. For example, moving from one line to the next one will have a sudden movement from one end of the page to the other end.
- Like the example in the previous point, we must find further unwanted motions possible in the pen and we must program the system to identify these movements. It will involve conditions like sudden change in altitude or sudden change in acceleration. The important thing here is to visualize these movements broadly in order to cover a wider range of users.
- I feel that the IMU sensor is the best available option. This coupled with the force sensor can make up the sensors within the pen.
- Size-wise, the MPU-6050 sensor is the best option. It is compact, uses I2C protocol to communicate with the master (in our case, the microcontroller). Cost is not a big issue here unlike the pressure sensor case. [Addicore GY-521 MPU-6050 3-Axis Gyroscope and 3-Axis Accelerometer IMU]



MPU-6050 IMU sensor

Why not use a CMOS sensor?

- I have suggested using a force sensor in my ideation. The CMOS sensor (complementary metal oxide semiconductor) performs the same functionality of the force sensor. It can be used to determine when the nib of the digital pen is in contact with the surface.
- CMOS sensor technology is used in optical mouses. The area that is in contact with the surface in case of an optical mouse is enormously large compared to the nib of a pen. The chances that light reflects off the surface into the sensor is significantly higher in an optical mouse.
- The CMOS sensor in the pen must be as close to the surface as possible just like a mouse. Ideally, it must be very close to the nib. This is very hard to implement (close to impossible).

- Also, if we need to have a CMOS sensor technology in our pen design, we need an infrared LED source, a semiconductor surface large enough to receive the reflected light and other parts of the sensor. We can install the exact same CMOS sensor that is used in the optical mouse (kind of second hand). But again, the space needed will be big enough to disrupt the pen design.
- Another issue will be the heat generated by an infrared LED in a CMOS sensor.

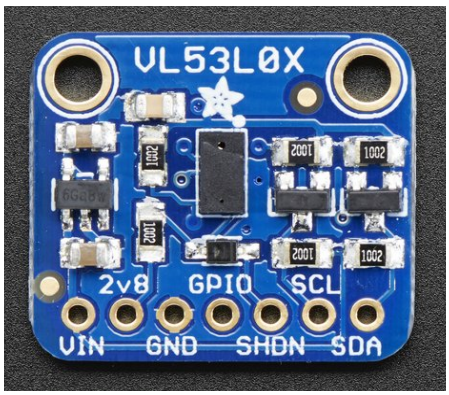
The issues mentioned above seem to outweigh the advantages that a CMOS sensor brings to the digital pen design.

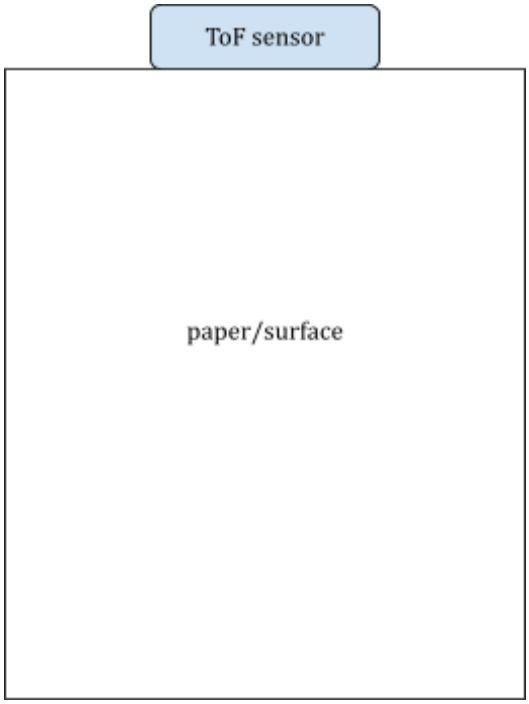
Sensors Outside the Pen:

Based on the outputs from the sensors inside the pen, the readings are taken. That is, if the person is determined as to be writing something, then outputs from the outside sensors are taken, else no.

A sensor to locate the exact nib coordinates:

There are a lot of options to choose from: Ultrasound sensors, IR sensors or ToF sensors.
The Ultrasound sensor and IR sensor are prone to false outputs in case there are multiple objects in front of the sensor.
A possible suggestion is to overcome this by having more than one IR sensor - possibly 4 sensors, one at each corner of the sheet. The reading in this will be more precise.
But again, 4 sensors can be very difficult to carry around. More difficult will be arranging these sensors to be exact corners of the sheet every single time. Writing becomes very hard and time-consuming.
The option I feel is the best is the ToF sensors. Their range is around 50-1200 mm, which is around the average size of a paper. We can keep the sensor right at the top of the paper. We can use 2 sensors, one for the X coordinate and one for the Y coordinate. The sensor is outside the pen. Hence, size is not a big parameter. The best choice is the VL6180X/VL53L0X ToF sensor. [Overview | Adafruit VL53L0X Time of Flight Micro-LIDAR Distance Sensor Breakout | Adafruit Learning System]





Sensor position w.r.t paper

[Interfacing VL6180 ToF Range Finder Sensor with Arduino for Distance Measurement (circuitdigest.com)]

Hence, the combination of sensors I have used in my ideation are:

- ➢ IMU sensor and Force sensor inside the pen and,
- ➢ ToF sensor(s) outside the pen.

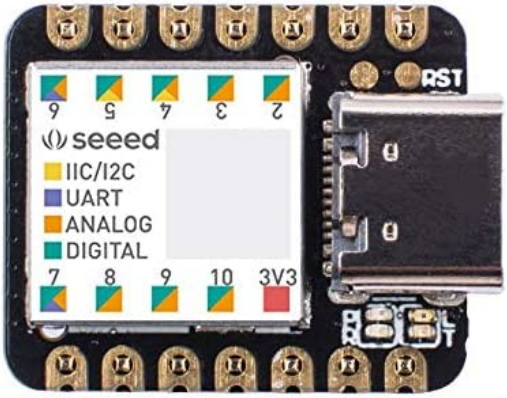
The Microcontroller:

- ➔ The microcontroller fits into the design smoothly because I2C protocol can be used between the microcontroller and each device. The main question the microcontroller must be able to satisfy is: *how to connect the external and internal sensors at the same time?*
- ➔ We may require both wired and wireless modes of connection because we have sensors inside and outside the digital pen.
- ➔ I feel it will be easier to have a microcontroller inside the pen and connect it wirelessly/through a wired pathway to the ToF sensor. The microcontroller has to be small and must have the capability to handle all three sensors at once.

Wired Mode Analysis:

https://github.com/Narasimhan07/EC04_Task2/blob/main/Communication.md#wired-mode-of-connection

In the wired mode, the microcontroller choice becomes simplified. We require the use of I2C protocol extensively. I feel the microcontroller based on SAMD21 will fit our needs.



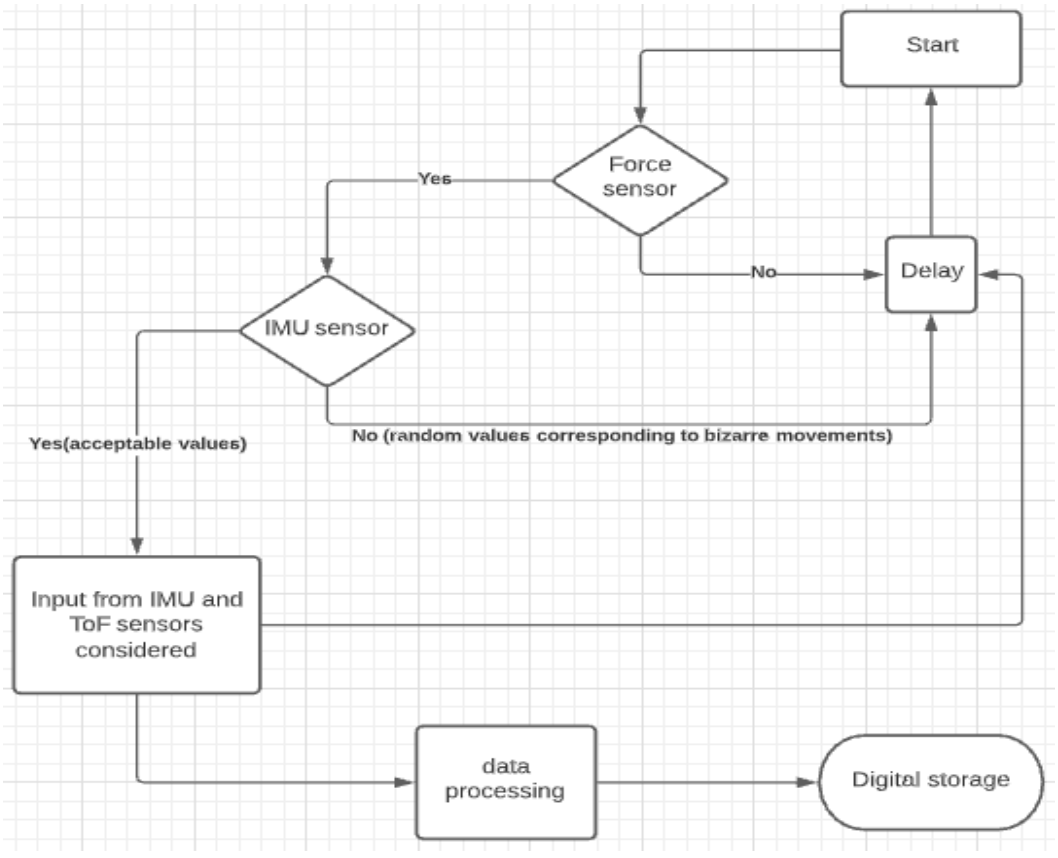
[Amazon.com: Seeeduino XIAO The Smallest Arduino Microcontroller Based on SAMD21,with Rich Interfaces, 100% Arduino IDE Compatible, desiged for Projects Need Arduino Micro, 3 pcs: Computers & Accessories]

Wireless Mode Analysis:

https://github.com/Narasimhan07/EC04_Task2/blob/main/Communication.md#wireless-mode-of-connection

As discussed in the above link, wireless mode seems slightly tougher to prototype and use. I feel the wired mode of connection will be better and easier to implement (and debug too).

Flowchart Visualisation of Component Ideation



A Brief overview of Software:

- ➢ It is not possible to track the motion of the pen continuously. We have to determine discrete points during the writing process i.e. sampling at regular intervals of time. This is what is visualized by the delay box in the flow chart.
- ➢ Smaller the interval of sampling/sampling time -> closer the sampled points -> stored data is more accurate.
- ➢ This is very much like joining the dots. Coordinates are sampled from the output given by the ToF sensor. Based on these dots we have to make small lines between consecutive dots.
- ➢ This entire process has been encapsulated in the data processing box in the flowchart above.
- ➢ Once again, there can be many more algorithms to implement the same step in the chart.

Possible Improvements:

I feel that the ideation in the 'sensors inside the pen' part is pretty strong. There can be slightly better arrangement/choice of sensors outside the pen. The only issue I feel might crop up is the inability of the ToF sensor to sense the nib at the left extremes or right extremes of the paper. The best possible way to overcome this incase it crops up is by using multiple ToF sensors.

Summary:

- ★ Sensors inside the pen: force sensor, IMU sensor
- ★ Sensor outside: ToF sensor
- ★ Microcontroller based on SAMD21 microchip
- ★ Wired connections used everywhere

Parts left for detailed ideation/further research: software and data transfer from microcontroller to processing unit (data processing box).