-4 …Need more details on R/V table, see comments there

Otherwise, excellent job!
56/60

Invisible Canvas Digital Pen

ECE 445 Fall 2015

Design Review

Pichamon Meteveravong
Perut Boribalburephan

TA: Cara Yang

September 29, 2015

Table of Contents

1 Introduction

1.1 Statement of Purpose

The project goal is to develop a digital pen that can define any flat surface of arbitrary size as a canvas and digitally draw on it. This effectively turns a spatial space into a temporary digital workspace with minimal dependency.

The stylus allows user to define the size of the canvas and is able to record drawing sessions even unconnected to host. Pencil lead will also be appended to the tip of stylus to display the drawing and provide user feedback on the canvas chosen. The calculations for position, pressure, error compensation, and canvas space definition will be done in-house. We expect to provide power using hard-wired voltage supply, and then change to micro-USB rechargeable battery later. The plug-and-play, draw-on-any-surface design allows for many creative uses, such as in a lecture room where lecturers and students can collaborate in a graphical discussion without having to walk up to the blackboard.

Our project fixes the problem of having a screen too small to draw on or to achieve desired level of detail. Additionally, using a stylus or a finger for drawing may block the view of the whole image, but using this pen, you will be able to collaborate with others on a touch screen without unexpected touch triggers and blocking. Hence, the invisible canvas digital pen can be a valuable platform for applications that require sharing interaction among users.

1.2 Objectives

1.2.1 Goals & Benefits:

- Eliminates the need of receiving-end hardware
- Setup of canvas dimension is transparent to the host
- Allows collaborative sharing of information/drawings
- Minimal dependency

1.2.2 Functions & Features:

- (x,y,z,r,p,y) location sensor
- User feedback on canvas shown by attached pencil lead
- Buttons for canvas definition
- Switch for turning on/off circuit
- Pressure sensor
- Lightweight
- Auto-Calibration
- Balanced design for portability
- Bluetooth module for intercommunications
- Micro-USB rechargeable battery

2 Design

2.1 Block Diagrams

2.1.1 System Overview

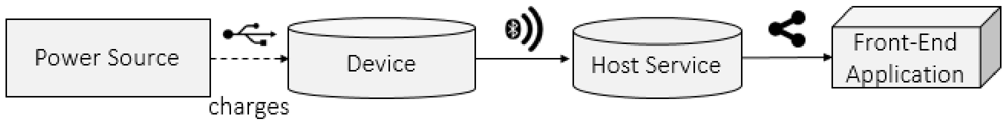


Figure 1. Top-Level System Block Diagram

2.1.2 Device

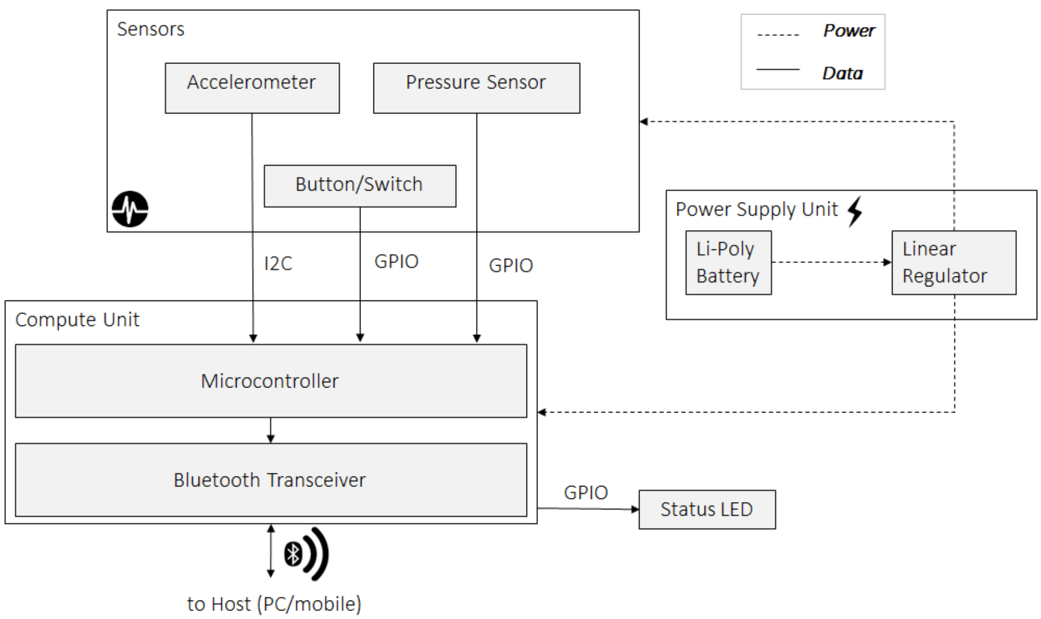


Figure 2. Device Block Diagram

2.1.3 Host Software

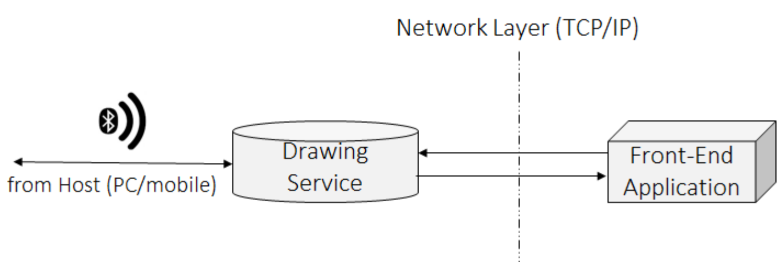


Figure 3. Block Diagram for Host-Side System Setup

2.2 Block Descriptions

2.2.1 Microcontroller (ATmega328P-PU)

Input: 1.8-5.5V power input, 3-axis acceleration and angular velocity data, voltage measurement for force, buttons and switch status bits

Output: Commands to read/write 3-axis data, commands to transmit position and pressure over serial bluetooth data

This unit polls the acceleration and orientation data from MPU-6050 3-Axis Accelerometer through I2C interface (SCL and SDA pins), measures analog voltage from an analog GPIO pin connected to a Force Sensitive Resistor (FSR), and reads from digital GPIO pins for switch and button states. The acquired measurements are then processed to yield appropriate values to be sent wirelessly to the host (HC-05 bluetooth module). Data processing models are discussed further in "Simulations and Calculations" section.

The ATmega328 microcontroller will be powered by a +3V source regulated by a +3.7V battery and integrated with the accelerator, FSR, switches, and bluetooth module on a PCB. A pin layout summary of these interfaces is shown in Figure 4:

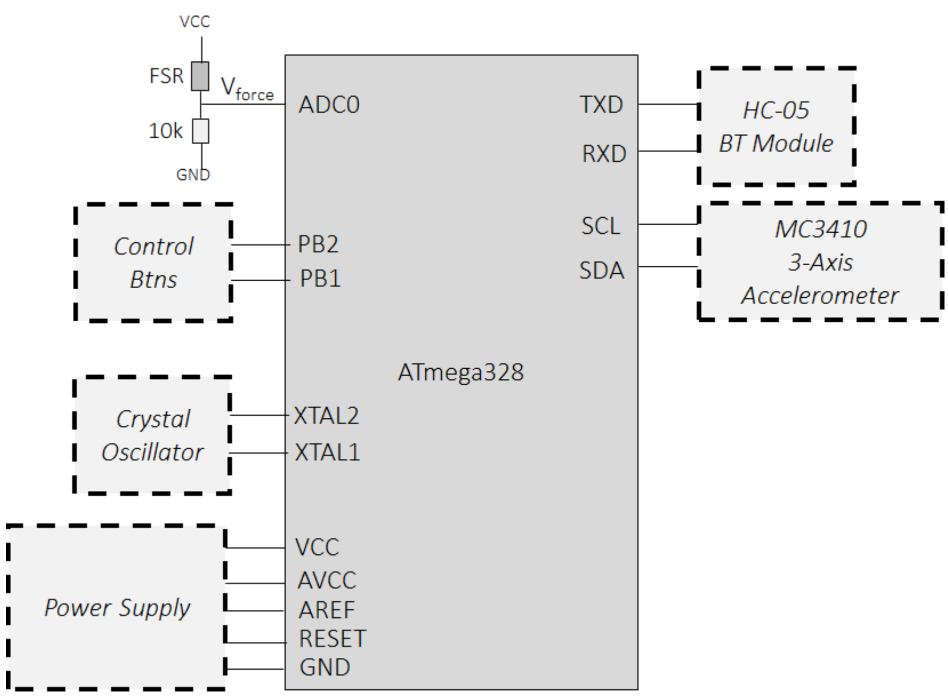


Figure 4. ATmega328 interfaces to other components

Pin	Function	Connection
1	RESET signal, active high	to reset button
2	RXD, serial receive	to TXD of HC-05 BT module
3	TXD, serial send	to RXD of HC-05 BT module
7,20	VCC/AVCC, digital/analog voltage supply	to regulated supply (+3V)
8,22	GND, ground	to ground supply
9	XTAL1, external clock input 1	to oscillator
10	XTAL2, external clock input 2	to oscillator
15	PB1, digital GPIO pin	to debug button
18	PB4, digital GPIO pin	to pen control button
21	AREF, analog reference voltage (top of input voltage range)	to regulated supply (+3V)
23	ADC0, analog pin	to force sensor circuit
27	SDA, I2C data line	to SDA of MC3410
28	SCL, I2C clock line	to SCL of MC3410

Table 1. Pin connection for the ATmega328 to other components

2.2.2 Pressure Sensor (Force Sensitive Resistor)
Input: Physical force on sensor
Output: Resistance change of the sensor component

The pressure sensor is a Force Sensitive Resistor (FSR) whose resistance changes based on applied pressure. The component will be connected serially to a 10kΩ resistor to the ground, allowing the resistance of the FSR to be calculated based on the voltage measured from the junction using the voltage divider rule. The debug and pen control buttons will be implemented along with the FSR circuit, as shown in Figure 5. Force calculation from junction voltage will be further discussed in 'Simulations and Calculations' section.

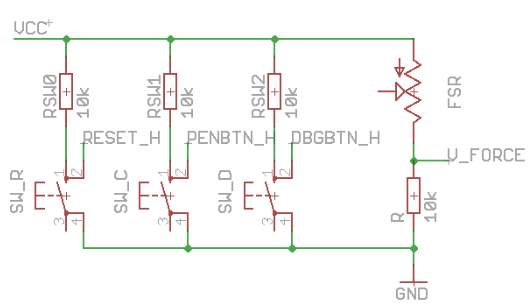


Figure 5. FSR circuit and push buttons schematic

2.2.3 Accelerometer (MPU-6050 Module 3-Axis Gyroscope + Accelerometer)

Input: Physical movement and orientation of the sensor chip
Output: Digital acceleration and orientation data through I2C data line

The MPU-6050 is a low-noise, integrated digital output 3-axis gyroscope + accelerometer. This unit will be assembled on PCB with microcontroller, wireless transmitter, and power supply.

In this project, we will need to detect the sufficient ranges of ±2g for accelerometer. The available range of gyroscope is from 250-2000 degree/sec. The sample rate can be set to 1024 samples/second and the low pass filter is programmable from 8 to 512 Hz bandwidth. The output data will be digitized into 16 bits and feed into microcontroller via I2C communication protocol. A pin function and connection summary is shown below in Table 2.

Pin	Function	Connection
6	AUX_DA (I2C Master serial data)	To external sensors
7	AUX_CL (I2C Master serial clock)	To external sensors
8	VLOGIC	1.71V to VDD
9	AD0	I2C Slave address LSB
12	INT (Interrupt digital output)	Totem pole or open-drain
13	VDD	To power supply
18	GND	To Ground
23	SCL	To microcontroller SCL
24	SDA	To microcontroller SDA

Table 2. Pin connection for the MPU-6050

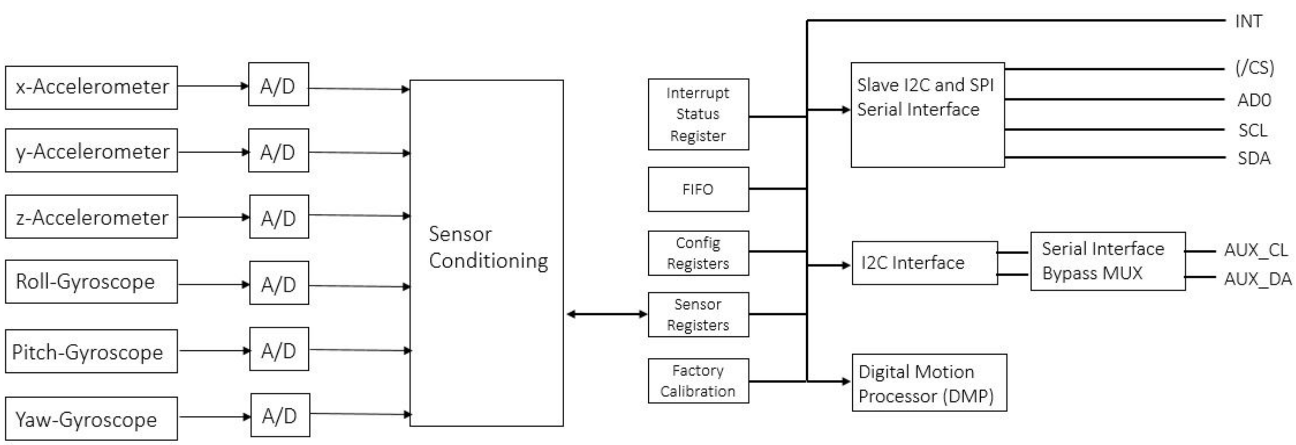


Figure 6. Motion Sensing Unit

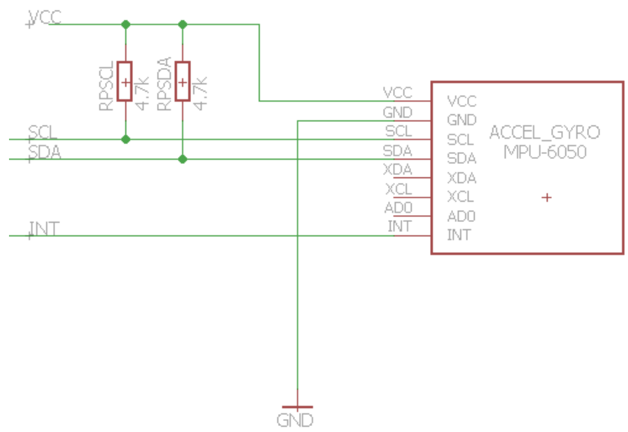


Figure 7. MPU-6050 3-Axis gyroscope + accelerometer schematic

2.2.4 Buttons/Switch
Input: Physical button and switch triggers
Output: Digital signal to ATmega328 GPIO pins on the microcontroller

Buttons and switch components serve as active user input of this device.

Switch
The switch is an on/off switch of the whole circuit. When the switch is held for a few seconds, the circuit will power off.

Reset Button (to ATmega328 pin 1)
The reset button resets ATmega328 microcontroller through an active-high reset pin.

Pen Function Button (to ATmega328 pin 15)
Each press on this button toggles operation mode of the device, consisting of canvas definition mode and drawing mode. When on drawing mode, the movement of the device will be processed and sent to the host. While in canvas mode, the device will appear as stationary to the host (i.e. the most recent data while on drawing mode is repeatedly sent to the host with pressure data overwritten to 0).

Debug Button (to ATmega328 pin 16)
The debug button triggers a software routine implemented in a microcontroller. This will be used to trigger useful blink patterns for debugging, and is designed to be an optional button.

2.2.5 Power Supply Unit

Input: Rechargeable +3.7V Li-Poly Battery
Output: +3.3V±5% for MPU-6050 3-Axis Gyroscope + Accelerometer
+3.3V±5% for ATmega328 Microcontroller
+3.3V±5% for HC-05 Wireless Bluetooth module

Our pen receives power input from the +3.7V battery. However, +3.7V is higher than the voltage needed for some modules, so resistors will be included in the power module to pull down +3.7V to other useful voltages, in order to supply power to the microcontroller, wireless transceiver, and accelerometer. Additionally, more resistors will be added as needed to ensure the current through the FSR does not exceed +1mA.

The power requirements for all the components from the device are shown below in Table 3:

Devices	Voltage/Current Requirements
MPU-6050 3-Axis Gyroscope + Accelerometer	VDD: 2.375-3.46V VLOGIC: 1.8V or VDD
ATmega328P-PU Microcontroller	Max Voltage: +3.3V or +5V
30ft Wireless RF Transceiver Bluetooth	Max Voltage: +3.3V input
Round Force-Sensitive Resistor (FSR) - Interlink 402	Max Current Density: 1 mA/cm^2
Mini Push Button Switch	Max: 50mA, 24V DC
LP2953 Low-Dropout Voltage Regulator	Input Voltage) Max: +30V, Min: -20V Output Voltage) Max: +29V, Min: +1.23V Fixed Output) Options: +3.3V, +5V
Lithium Ion Polymer (Li-Poly) Battery	Output Voltage: +3.7V

Table 3. Power Requirements for all components from the device

The rechargeable battery requires +3.7V to charge. A linear regulator will be used to provide the reference voltage at +3.3V, as shown below in Figure 8.

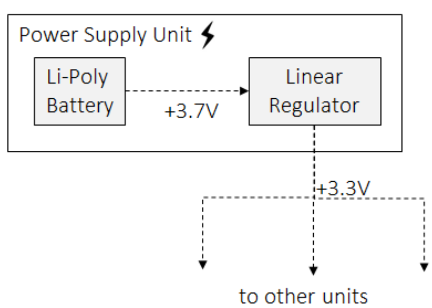


Figure 8. Power Module Block Diagram

The linear regulator forces a fixed voltage to appear at the regulator output terminal. The output voltage is controlled by a feedback loop. LDO (Low-Dropout) regulator is chosen specifically for our project due to its compatibility with battery-powered applications. Not only that LDO regulators utilize available input voltage more fully, but they also reduce power dissipation and hence improve cost savings.

LP2953 is a micropower voltage regulators that give very low dropout voltage. It has two fixed output options, +3.3V and +5V, which +3.3V will be used for our circuit design. Along with the power supply, the linear regulator will be sufficient to supply +3.3V for other components, such as the Bluetooth module and microcontroller.

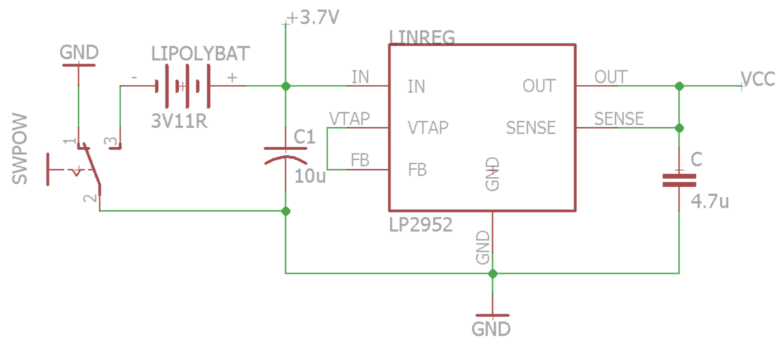


Figure 9. LP2952 Linear Regulator schematic

Pin	Function	Connection
1,8,9,16	GND, Ground	to ground supply
2,7,10,11	NC, no connections	-
3	OUT, regulated output voltage	to other components as VCC, connect to SENSE and capacitor to ground
4	SENSE, feedback voltage sense input	connect to OUT and capacitor to ground
13	VTAP, internal resistor divider input	short to FEE
14	FB, error amplifier noninverting input	short to VTAP
15	IN, regulator power input	to battery

Table 4. Pin connection for LP2952 Linear Regulator

2.2.6 Bluetooth Transceiver

Input: +1.8V±5% power supply, pre-processed canvas position and actions from the microcontroller
Output: data transmitted to host device (bluetooth protocol)

The serial port Bluetooth modulates 3Mbps enhanced data rate with complete 2.4GHz radio transceiver and baseband.

PIN Name	PIN	Pad Type	Description
GND	13,21,22	VSS	Ground pot
3.3 VCC	12	3.3 VCC	Integrated 3.3V supply with on-chip linear regulator output within 3.15-3.3V
AIO0, AIO1	9,10	Bi-Directional	Programmable input/output line
PIO0, PIO1	23,24	Bi-Directional RX EN (23) / TX EN (24)	Programmable input/output line, control output for LNA (23) / PA (24) (if fitted)

Table 5. Pin connections for HC-05 bluetooth transceiver

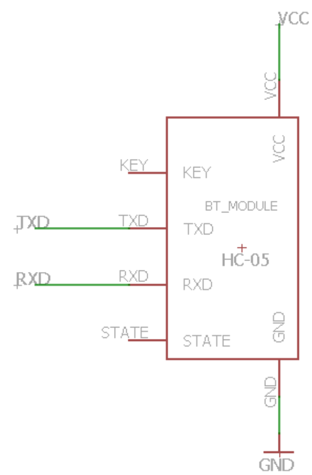


Figure 10. HC-05 Bluetooth Transceiver schematic

2.2.7 Status LEDs

Input: Signals from button/switch
Output: Two LED lights, Digital signals to ATmega328 microcontroller

Green LED (LEDG, to ATmega328 pin 12)
This LED indicates that the data is successfully being transmitted to the host device via Bluetooth. In each transmission event the LED will blink once.

Orange LED (LEDO, to ATmega328 pin 13)
This LED indicates that the user is activating the pen to be in a drawing mode. This is controlled directly by the microcontroller and the button/switch integrated into the circuit.

2.3 Schematics of Overall System

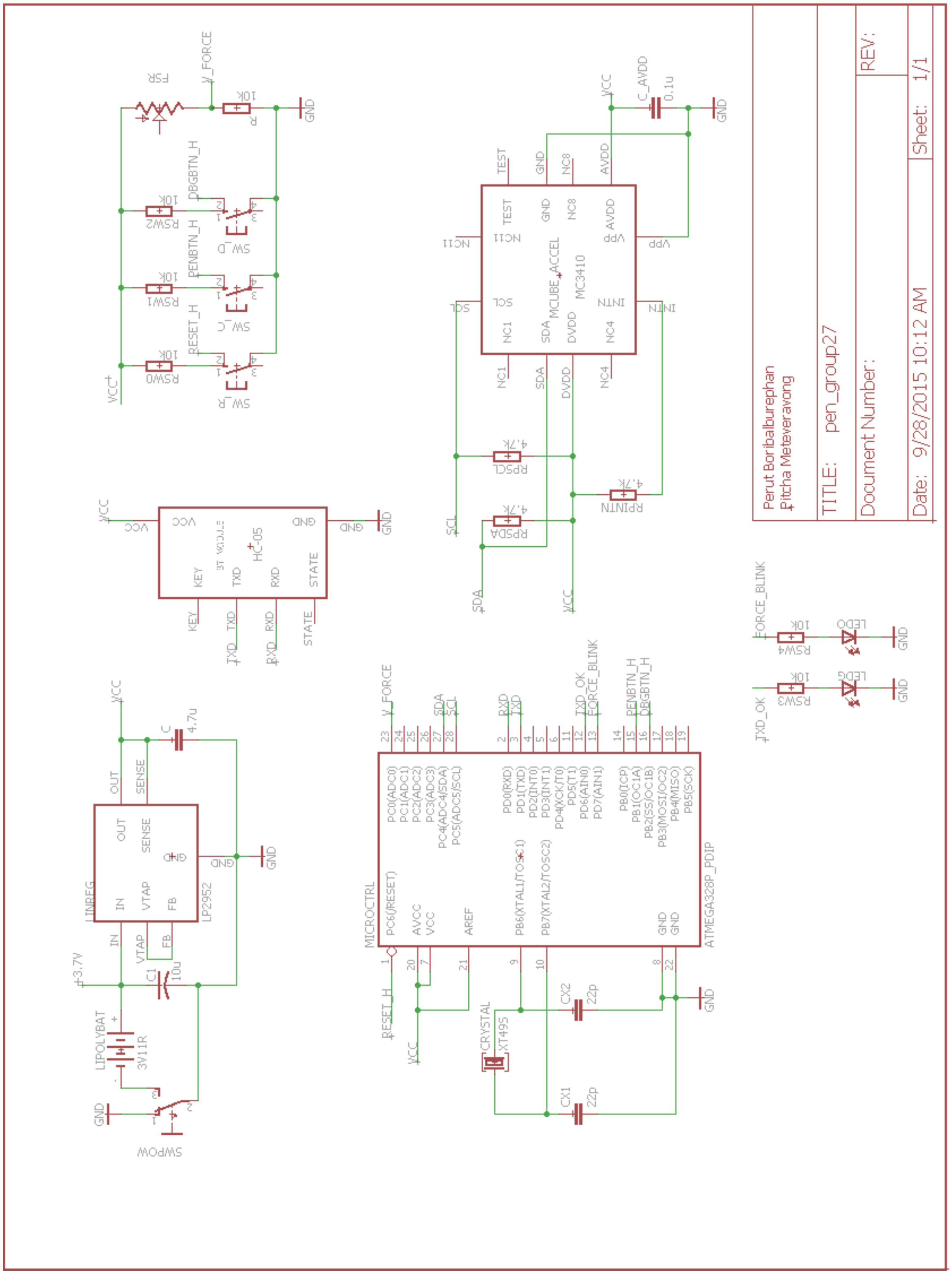


Figure 11. Circuit Schematic

2.4 Software Flowchart

2.4.1. Device Software

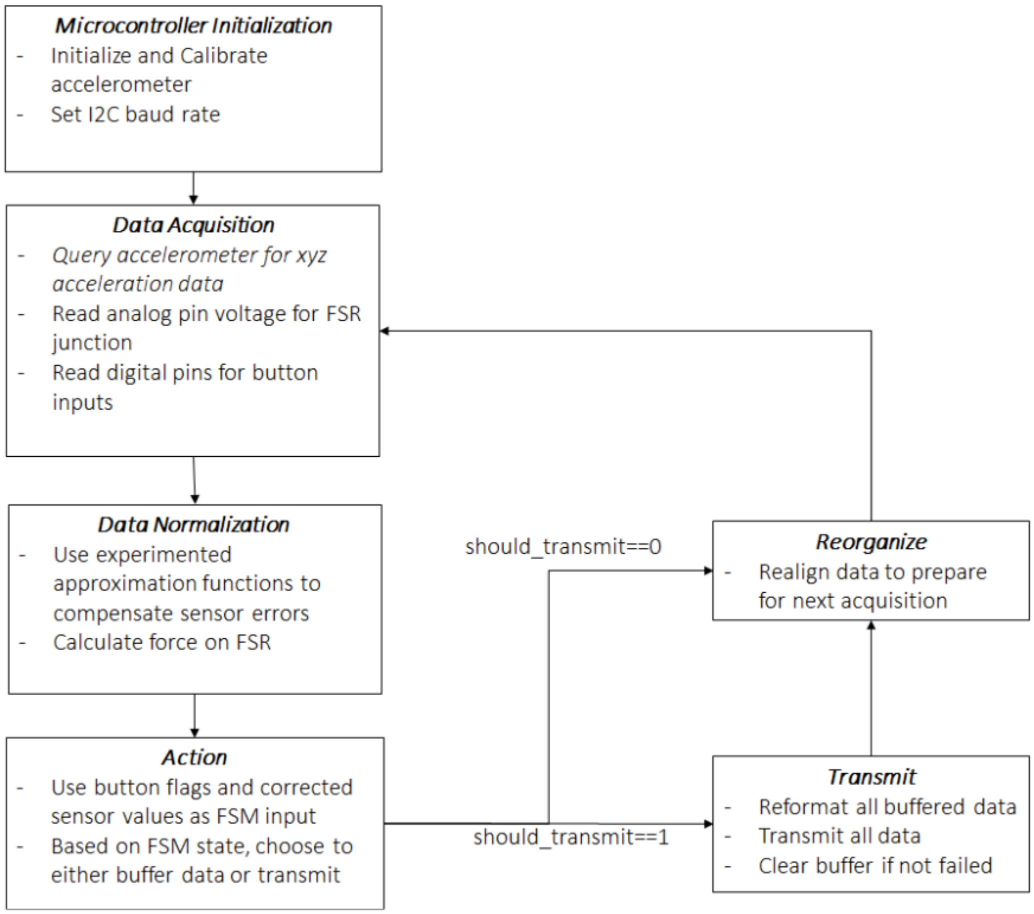


Figure 12. Top-Level Device Software Algorithm

2.4.2 Device Software Finite State Machine

This figure demonstrates the logic in the Action block shown in Figure 12.

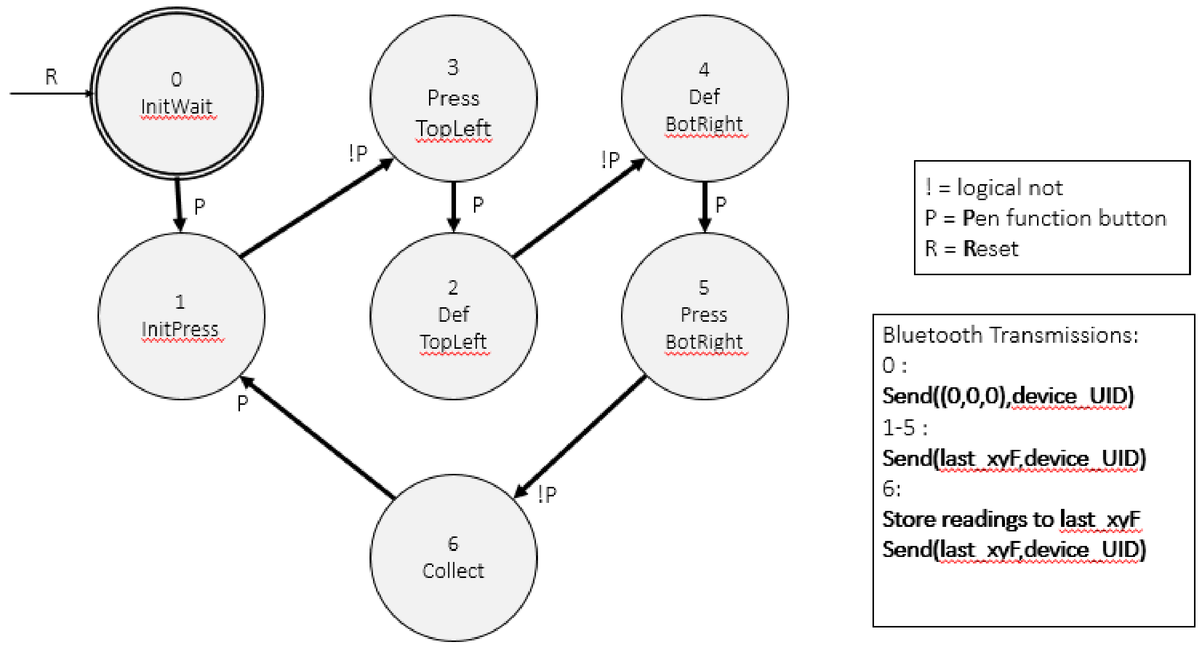


Figure 13. Finite State Machine of the device

2.4.3 Host Service Software

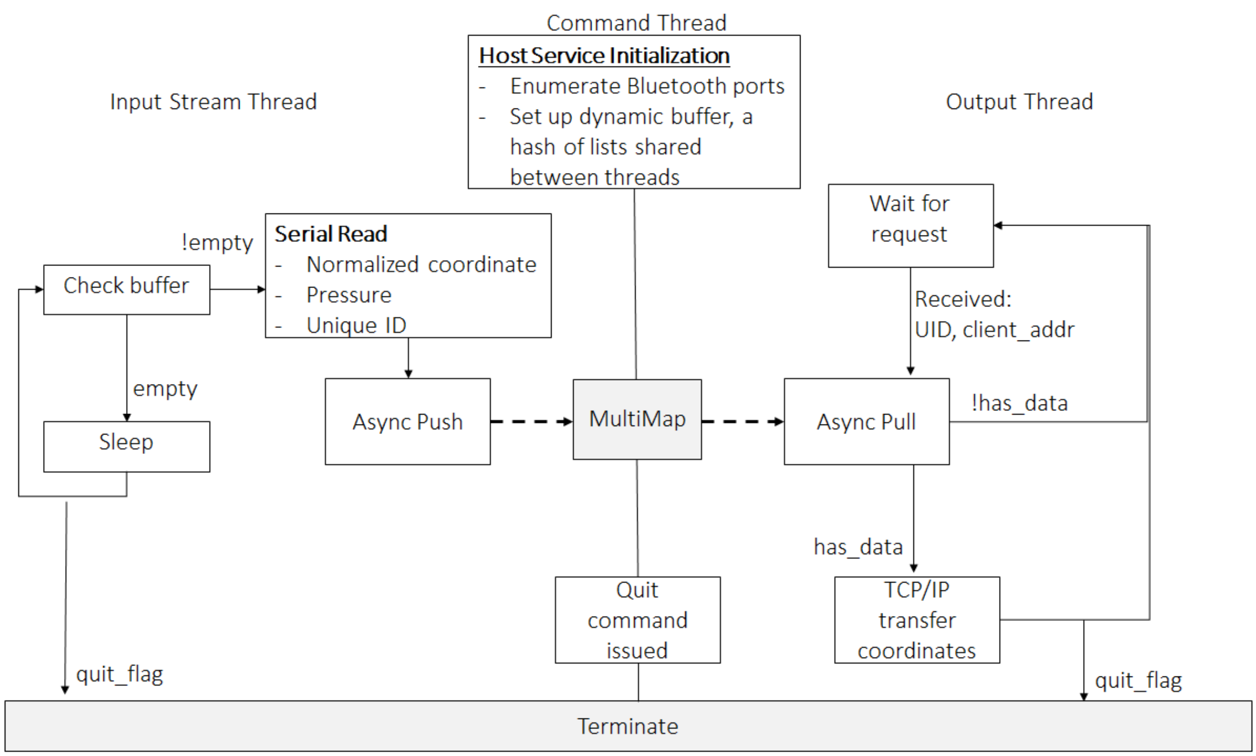


Figure 14. Host Service Software Flowchart

2.4.4 Front-End Application

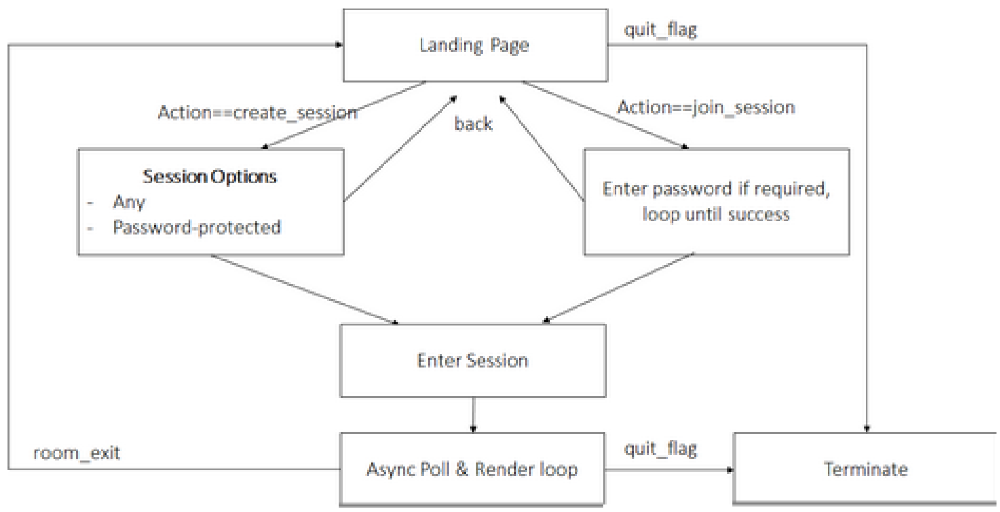


Figure 15. Front-End Application flowchart

2.5 Simulations and Calculations

2.5.1 Sensor Resolution

Acceleration

The accelerometer sensor provides 16-bit resolution for ±2g, ±4g, ±8g reading ranges. The range ±2g is chosen as the magnitude range is sufficient for stylus usage and it also provides the smallest granularity among all possible ranges.

$$\frac{2 \cdot 9.81}{2^{14}} = 1.19\ mm/s^2$$

Force

The force-sensitive resistor provides continuous and smooth analog junction voltage. However, information is lost due to conversion to digital range. The microcontroller converts the voltage from range 0 to V_{ref} to unsigned integer from 0 to 1023, since V_{ref} is 3.3V, the finest voltage unit that can be measured is

3.3/1024 = 3.22 mV per unit

Since the relationship between voltage and force applied is nonlinear, as discussed in section 2.5.4, each voltage increment corresponds to a larger force increment the the previous increment.

The discrete sampling of data leads to another design decision. The force values can be stored in flash memory as a table with 1024 entries of pre-calculated values, using the unsigned integer range as table index. This simplifies a relatively long calculation to a memory load and saves SRAM space at an expense of flash memory. Additionally, the microcontroller has more limited SRAM space (2KB) flash memory space (32KB, with 2KB used for bootloader).

2.5.2 Power Consumption
Battery: Nominal voltage provides 3.75V
Power Capacity [Wh]: Voltage that battery provides x Current it provides during the time it is providing
- Capacity: 1.2Ah x nominal voltage = 1.2x3.75 = 4.5Wh
- Power Capability [C]: A-hr/1hr
- For Li-Poly battery, 1200mAh = 1.2Ah, hence C = 1.2A

Battery	**Power**
Li-Poly Battery	+1200mAh

Load	**Voltage**	**Current (Active Mode)**	**Current (Sleep Mode)**	**Power Consumption (Active Mode)**	**Power Consumption (Sleep Mode)**
Microcontroller (15 pins)	+1.8V	+0.2mA	+0.1uA	+0.2mA*15 = +3mA	+0.1uA*15 = +1.5uA
LED (2)	+1.8-2.2V	+18mA/part	+0mA	+18mA*2 = +36mA	+0mA
Force-Sensitive Resistor	about +1.8V assuming 0.25N force, 25KΩ resistance	+1mA/cm2 (max) +0.18mA through FSR+R	+0mA/cm2	+1mA per press (max) +0.18mA assuming 0.25N force, 25KΩ resistance	+0mA
EIO Button	+24V (max)	+50mA (max)	+0mA	+50mA (max)	+0mA
LDO Regulator	+3.3V	+19mA	+170uA	+19mA	+170uA
Bluetooth Module	+1.8V	+30mA	+0mA	+30mA	+0mA
Accelerometer + Gyroscope	+1.8V	+500uA+3.6mA	+10uA+5uA	+4.1mA	+15uA
Total				+143.1mA	+186.5uA

Table 6. Battery and load power summary.

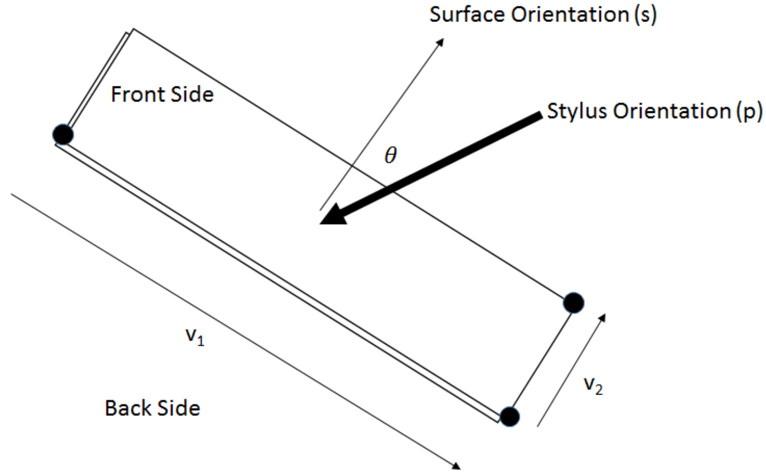
Total power input (One Li-Poly Battery): +1200mAh
Assuming worst case possible, total load current:
- **Active Mode**: (3+36+1+50+19+30+4.1)mA = +143.1mA
- **Sleep Mode**: (1.5+0+0+0+170+0+15)uA = +186.5uA
- **30:70 Mode** (assuming 30% active state 70% sleep state as an estimated typical usage): (+143.1mA)*0.3 + (+186.5*10^-3)*0.7 = +43.06mA

Calculations have shown that the battery still provides sufficient power for the circuit despite assuming maximum possible power consumed by components.

Standard discharge current: 0.5C5A = 0.6A = +600mA → 1200mAh/600mA = 2 hours
Standard charge current: 0.2C5A = 0.24A = +240mA → 1200mAh/240mA = 5 hours
<u>Time for Usage</u>: 1200mAh/(143.1mA+600mA) = 1.61 hours
<u>Time for Standby</u>: 1200mAh/(186.5uA+600mA) = 2 hours

2.5.3 Plane of Writing, Position and Orientation Determination

Plane of Writing



The plane of writing is defined to be a flat rectangular surface with front side and back side. The stylus will only register write gestures to the front side. To be able to uniquely identify the plane of writing, we need:

<u>During Calibration</u>

a. Three points of reference which will lie on the same plane. This is the minimum number of points required to uniquely identify a plane.
b. The orientation of the stylus at the time the three points of reference are defined. This is used to determine whether the plane should be written from under or over the surface. The surface normal will be set during calibration to be perpendicular to the surface, so $\mathbf{s} = \pm \mathbf{v_1} \times \mathbf{v_2}$. The sign for the calculation will be chosen so that $\mathbf{s}^T\mathbf{p} < 0$ for all unit vectors **p** sampled during calibration (i.e. the surface points back to the stylus), or the calibration fails and restart.

<u>During Usage</u>

a. Always track the position and orientation of the stylus in space after calibration is done (done by processing accelerator data).
b. At any given time, check the unit vector dot product $-\mathbf{s}^T\mathbf{p} = \cos\theta$ (minus sign due to how **p** is defined) to determine the angle of the stylus with respect to the surface normal,

$$\theta = \mathrm{acos}(-\mathbf{s}^T\mathbf{p})$$

c. If $\cos\theta < 0$ then the pen is either held upside down or being written from the back side, in which case any pressure applied to the stylus will be ignored. Otherwise, it is written from the front side. If θ is larger than a threshold θ_c then the stylus is writing on the front side with a physically impossible angle. Otherwise if $\theta < \theta_c$, and the stylus is in allowed orientation for writing.
d. If the stylus is held upside-down by the definition in part c. or is in a physically impossible angle for a long time, put the device to sleep. The device wakes up after $\theta < \theta_c$ and $\cos\theta > 0$, that is when the stylus is back in allowed orientation again.
e. While the device is not sleeping and the position of the stylus lies on the plane (with tolerance of about 5%), any pressure above threshold that is applied to the tip is registered as a press.

Position and Orientation Calculation

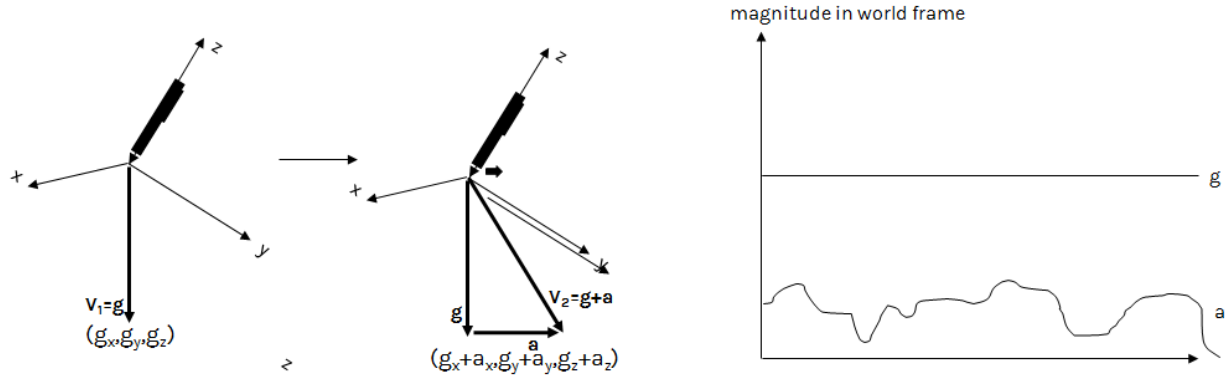


Figure 16. Stylus orientation

With orientation as shown in Figure 16, we have that an accelerometer experiences both gravitational acceleration and net linear acceleration done by the environment. The value of **g** does not fluctuate quickly compared to linear acceleration **a**, therefore we can extract linear acceleration from gravitational acceleration using a high-pass filter [9] (and conversely extract gravitational acceleration using low-pass filter) e.g.

$$\boldsymbol{g}_n^* = \alpha \boldsymbol{g}_{n-1}^* + (1-\alpha)\boldsymbol{x}_n$$
$$\boldsymbol{a}_n^* = \boldsymbol{x}_n - \boldsymbol{g}_n^*$$

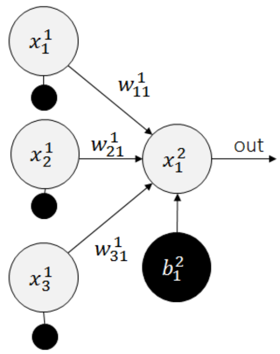
where $\mathbf{x}_n = \mathbf{g}_n + \mathbf{a}_n$ is a sampled acceleration vector that combines both gravity and linear components. Once we have the predicted acceleration, we send it through the correction unit along with absolute position the microcontroller has tracked so far.

Correction

Measured values for position and orientation depend on the absolute position itself, the orientation, the rate of change of those two parameters, and higher derivatives, and so on. Sampling the data into digital form is the main source of the error as it truncates the precision of the measurements (or, from another point of view, generates alias). Extended use of the measurements without error correction will accumulate the error up significantly. B. Deepak et al. [3] has approached this problem using a support error machine (SVM) as a model for error compensation characterization. The work achieved higher correction accuracy compared to the conventional error modeling approach.

In this system, the error correction algorithm will be implemented using a subset of artificial neural network (ANN), called perceptron. Perceptron is a feedforward neural network; there are no feedback connections and the activation signals always propagate forward to deeper layers. The accumulation of signals will be tested against a threshold (or bias value, denoted **b**), to generate an output according to a certain kernel function **f**. In most neurons, a sigmoid function is used.

$$f(x) = \frac{1}{1 + e^{-x}}$$



$$x_j^l = f\left(b_j^l + \sum_k w_{kj}^{l-1} x_k^{l-1}\right)$$

Figure 17. The calculation of output value of one neuron in a neural network

A neural network's output can be thought of as a transformation of feature (input) vector, which is a member of feature space, into a member of output space. After the system has been tested with a training set (a set of pairs of input and expected output), the weight parameters (w) can be adjusted accordingly using training algorithms such as backpropagation [10] coupled with gradient descent or genetic evolution so that future outputs of the system will match more closely to the expected output.

In this application, we prepare the training data by attaching carbon piece rigidly to the accelerometer and test on paper so that the trajectory of the drawing is left on the paper. The physical measurement of the displacement can be done easily from the carbon trace. Then, the physical data (expected output) is paired manually to the instantaneous parameter estimates (x,y,z,roll,pitch,yaw) collected digitally while the carbon trace was being drawn. The network will have one input layer for the digital measurements and their differential change, one hidden layer, and outputs corrections for the differential changes. The initial layer sizes will be **12x6x3**, from input layer to output layer respectively. However, the results of this structure may present underfit or overfit problems, which suggests the restructuring of the neural network layers and parameter space for the input values.

2.5.4 Force Calculation

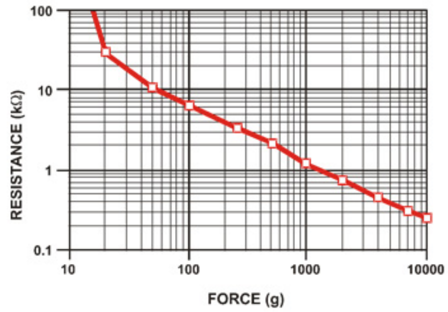


Figure 18. Resistance of Force Sensitive Resistor (FSR) with respect to force applied

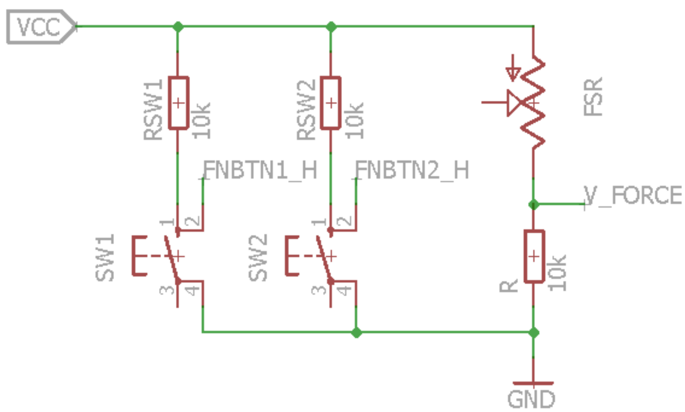


Figure 19. FSR configuration

From Figure 18, the correlation between applied force and FSR's resistance is piecewise exponential, that is for each interval of resistance **[a,b]**, and force applied **F**, FSR follows the relationship

$$ln(FSR) = A_{[a,b]} ln(F) + B_{[a,b]}$$

From the configuration shown in Figure 19, the voltage divider rule on FSR and R (10kOhm) gives

$$V_{FORCE} = \frac{V_{cc}R}{R + FSR}$$

$$FSR = \frac{(V_{cc} - V_{FORCE})R}{V_{FORCE}} = 10k\left(\frac{3 - V_{FORCE}}{V_{FORCE}}\right)\ Ohm$$

$$ln(FSR) = 2.3025 + ln\left(\frac{3 - V_{FORCE}}{V_{FORCE}}\right)$$

so we can approximate the force applied on the sensor; the logarithmic value of the force is sufficient for this purpose:

$$ln(F) = \frac{2.3025 + ln\left(\frac{3 - V_{FORCE}}{V_{FORCE}}\right) - B_{[a,b]}}{A_{[a,b]}}$$

Force (N)	FSR Resistance	(FSR + R) Ω	Current thru FSR+R (mA)	Voltage across R(V)
None	Infinite	Infinite	0	0
0.2	30KΩ	40 KΩ	0.0825	0.86
1	6 KΩ	16 KΩ	0.2062	2.05
10	1 KΩ	11 KΩ	0.2999	2.99
100	250 Ω	10.25 KΩ	0.3252	3.23

Table 6. Example Voltage output at known FSR force and resistance

Constants for F-FSR Relationship			
a (kΩ)	**b (kΩ)**	**A**	**B**
0.1	30	-0.78377	2.1397
30	infinity	-5.1918	-4.9546

Table 7. Regression results for ln(F)-ln(FSR) relationship

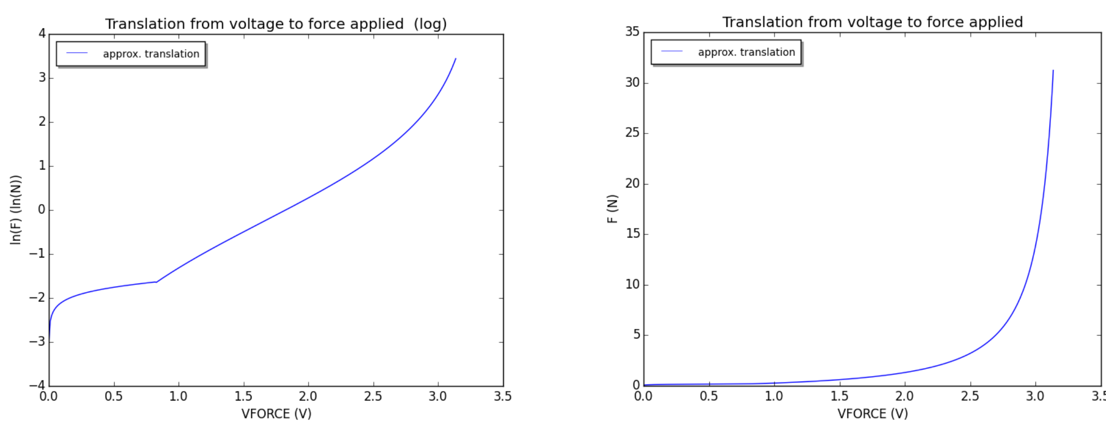


Figure 20. FSR Voltage to Force translation

Break out ALL verification blocks into much more detail, as if you were writing instructions for someone else to do the testing, i.e. "...probe at __, measure width of square pulse at __V crossings with oscilloscope."

3 Requirements and Verification

More Reqs needed for micro. What about processing delay? What maximum latency will you tolerate before the user notices? How much onboard memory do you need? How much external mem?

3.1 Requirements & Verification

3.1.1 Microcontroller [20 Points]

Requirements	Verifications
1) When mounted to PCB, the microcontroller should program the Bluetooth module to transmit data at a baud rate of 115200.	1) Run two codes: ArduinoISP and Pen27 (code containing actual functions of microcontroller), via Arduino onto microcontroller and burn bootloader. Connect the microcontroller to the soldered PCB. The host program should be able to read the data at a rate of 115200 (can be displayed on host program).
2) The data format should be defined as: byte 0: STX byte 1: command character byte 2 to 2+(N-1): command parameters byte 2+N: ETX	2) Verify with host program by examining whether the format of data packet arrives accordingly to the format defined.

3.1.2 Pressure Sensor [15 Points]

Requirements	Verifications
1) Pressure level must be shown on screen to vary accordingly with different amount of force applied. Pressure measurements must be aligned with force measured by different displacements of an elastic spring with an accuracy of 10%.	1) Place a wood block between FSR and elastic spring to measure displacement vs. force (hence pressure). Measure the voltage at each level of force then plot a graph of Force vs. Voltage from data obtained. A trend similar to graph shown in Figure 20 should be present, with error values less than 5%.

How do you know the "right" pressure level and how accurate your pressure sensor readings are?

3.1.3 Motion Sensing Unit: Accelerometer & Gyroscope [15 Points]

Requirements	Verifications
1) Detects the motion according to the movement of the pen. Canvas must be displayed with software and the stylus should be assigned to (x,y,z) coordinates, which depend on the relative position of the user-controlled pen location with an accuracy of 10%.	1) Motion of the pen tip should be displayed on software program to see if it moves correspondingly to the actual movement. Compare the displayed coordinates of the whole pen relative to movement. The corners of canvas must be correctly defined according to 3 buttons pressed.
2) Detects rotational movement with at most 360 degree per second with an accuracy of 10%	2) Compare measurements obtained from software relative to sensor data acquired and angles measured using a protractor.
3) Detects linear acceleration of ±2g with an accuracy of 10%.	3) Compare measurements obtained from software relative to sensor data acquired.

Should be verifying debounced signal vs. jumpy input signal with a oscope.

3.1.4 Buttons [10 Points]

Requirements	Verifications
Button: 1) Buttons should be debounced and allows only one register input per press. 2) Check to make sure that pen function button presses triggers microcontroller finite state machine transitions.	Button: 1) Voltage measured from the pin connected to the button behaves as digital active-low signal when buttons are pressed or released (pressed=LOW, released=HIGH). 2) Check on the host device display whether a line is being drawn accordingly when the pen function button is pressed up to the Collect state (3 times).

3.1.5 Power Supply Unit [15 Points]

Requirements	Verifications
Li-Poly Battery: 1) Supply +3.7V±5% power at 200uA draw on the PCB. 2) Lifetime of the pen should be approximately 1 hour 37 minutes assuming full-time usage and 2 hours during standby.	Li-Poly Battery: 1) A multimeter will be used to check if the voltage and current outputs are equal to specified values. Use an oscilloscope to check if the voltage signal is steady. 2) Measure the lifetime of the pen during active usage and during standby. See if the measurements are close or last longer than the calculated values. (The longer it lasts, the better).
Linear Regulator: 1) Voltage must be regulated to +3.3V±5% at 170uA draw as required for specific components.	Linear Regulator: 1) Connect multimeter and oscilloscope across the linear regulator to measure if the potential difference is within specified range. It is also a good practice to check the current whether it is close to 170uA as well.

3.1.6 Bluetooth Transceiver [15 Points]

Requirements	Verifications
1) Module needs to be implemented with orange LED as an indicator. 2) Use the algorithms we set up to transfer data to the host device.	1) If data is successfully being transmitted to the host device, the orange LED close to crystal oscillator is correctly lit up. 2) The receiving end receives data in the correct format.

3.1.7 Status LEDs [10 Points]

Requirements	Verifications
Indicates a pressed button: 1) Microcontroller, button, switch, and FSR must all be integrated in order to enable functionality of status LEDs. Indicates Bluetooth on: 2) Microcontroller, button, switch, and Bluetooth must all be integrated in order to enable functionality of status LEDs.	Indicates a pressed button: 1) Orange LED (right) is lit up when the button is pressed by a user. Indicates Bluetooth on: 2) Another orange LED (left - closer to crystal oscillator) is lit up when the data is successfully being transferred via Bluetooth.

3.2 Tolerance Analysis

The critical part of the design is the accelerometer sensor module. If the I2C accelerometer sensor fails to store immediate data to the register interface correctly for the microcontroller to fetch, then errors will propagate to the error correction algorithm. Let **p** be the probability that there will be a bit error in a single bit, then **P**, the probability that the 14-bit x 3 acceleration values will contain error will be (from packet error ratio calculation [11] with n=42)

$$P = 1-(1-p)^{42}$$

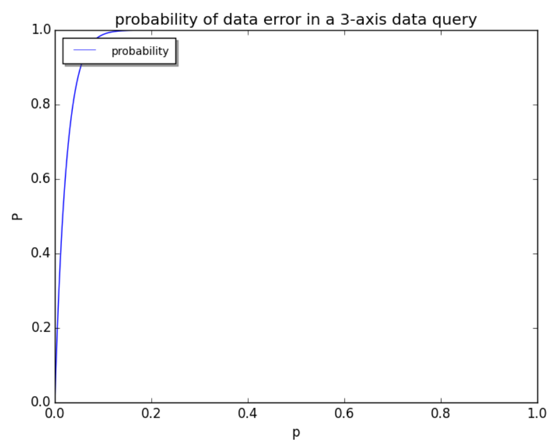


Figure 21. data error of one dataset

The probability **P** grows in high order with respect to **p**, but **p** in reality is very low, therefore at **p**~0 we can estimate that

$$P = 42p$$

Therefore the expected value for number of errors per second is KP^{-1} where **K** is the number of samples taken per second. The error may occur in any of the 14 bits of each axis, so the expected value of the error once it occurs is, assuming independence of the bits,

$$E[a_{x,err}]=E[a_{y,err}]=E[a_{z,err}]= sum(2^i, i=0 \text{ to } 13) /14 = (2^{14}-1)/14 = 1170$$
$$\%E[a] = 1170/2^{14} \sim 7\%error$$

Since the number of errors that cannot be avoided by correction software is **KP^{-1}** per second, the average drift due to error is **$0.07KP^{-1}$** per second at and the maximum accumulated error after **T** seconds is expected to be **$[(1+0.07KP^{-1})^{T}/\ln(T)]-1$**. From the design, we expect this value to be within 5% bound (0.05). After some time **kT** the percent error will exceed 5% assuming we use the data from a big time window to correct current data, therefore a timeout is needed for any data used for processing and should be discarded as the time expires. Note that the actual spatial error is lower than the stated bounds because the interpreted value for the acceleration readings are signed values, not unsigned, meaning that the net drift error accumulated over a usage on the same surface of a random user with random movement would result in a very small accumulated error (accumulated error amplify future error is low, but error still exists as derived in this section).

3.3 Safety

The invisible canvas digital pen has low power consumption overall, hence there are only few parts that can be harmful. However, our project relies heavily on Li-Poly battery and we will be charging/discharging the battery many times. Therefore, those following factors are important and must be considered at all times:

- Do not charge batteries above their maximum safe voltage (+4.2V)
- Do not discharge batteries below their minimum safe voltage (+3.0V)
- Do not draw more current than the battery can provide (1-2C)
- Do not charge them with more current than the battery can take (~1C)
- Do not charge batteries above or below certain temperatures (usually 0-50 degrees C)
- Do not short circuit the battery (avoid connecting positive and negative terminals to each other with conductive materials)
- Do not immerse the battery in water or liquids. Keep or store it in a cool and dry place
- Make sure the battery is not installed in reverse polarity
- Do not leave battery in a high temperature environment. It could lead to overheating of battery and/or explosion or fire.
- Do not use battery in an environment with high static-electricity or magnetic fields
- Prior to usage, the batteries should be inspected for deformities and potential failure
- NiMH/NiCad/lead-acid charger should not be used

Furthermore, all members of the team have completed the lab safety training before engaging with any lab work. The trainings completed include both general laboratory safety training and electrical safety training.

3.4 Ethical Issues

Our project follows IEEE codes of ethics as following:

[1] To accept responsibility in making decisions consistent with the safety, health, and welfare of the public, and to disclose promptly factors that might endanger the public or the environment.

[3] To be honest and realistic in stating claims or estimates based on available data.

[5] To improve the understanding of technology; its appropriate application, and potential consequences.

[6] To maintain and improve our technical competence and to undertake technological tasks for others only if qualified by training or experience, or after full disclosure of pertinent limitations.

[7] To seek, accept, and offer honest criticism of technical work, to acknowledge and correct errors, and to credit properly the contributions of others.

[9] To avoid injuring others, their property, reputation, or employment by false or malicious action.

[10] To assist colleagues and co-workers in their professional development and to support them in following this code of ethics.

4 Cost and Schedule

4.1 Cost Analysis

4.1.1 Labor

Name	Hourly Rate	Hours	Total	Total x 2.5
Pichamon Meteveravong	$30	300	$9000	$22500
Perut Boribalburephan	$30	300	$9000	$22500
Total				$45000

Table 8. Labor Costs

4.1.2 Parts

Description	Quantity	Manufacturer	Vendor	Cost/unit	Total Cost
ATmega328P-PU with arduino bootloader	2	Atmel	Amazon	$6.35	$12.70
USB LiPoly Charger	1	Microchip	Adafruit	$6.95	$6.95
Li-Poly Battery with JST-PH Cable	1	Hunan Sounddon New Energy Co. Ltd.	Adafruit	$9.95	$9.95
30ft Wireless Bluetooth RF Transceiver Module Serial RS232 TTL HC-05 for Arduino	1	Sunkey	Amazon	$8.89	$8.89
MPU-6050 3-Axis Gyroscope + Accelerometer	1	InvenSense	Amazon	$5.00	$5.00
Round Force-Sensitive Resistor (FSR) - Interlink 402	2	Interlink Electronics	Adafruit	$7.00	$14.00
Stylus Hardware Container	1				$10.00
Mini Push Button Switch	2	TE Connectivity	SparkFun	$0.35	$0.70
LP2953 Adjustable Micropower Low-Dropout Voltage Regulator	1	Texas Instruments	Texas Instruments	$11.58	$11.58
Crystal Oscillator	3	Vishay Dale	Mouser	$0.49	$0.49
Total					$70.16

Table 9. Component Costs

4.1.3 Grand Total

Section	Total
Labor	$45000
Parts	$70.16
Grand Total	$45070.16

Table 10. Grand Total Cost (Labor + Parts)

4.2 Schedule

Week	Task	Responsibility
09/14/15	**Project Proposals Due**	All
	Finalize project proposal	Pichamon
	Prepare mock design review	Perut
09/21/15	**Mock Design Review (Wed) Eagle Assignment (Fri)**	All
	Research and select all hardware modules	Pichamon
	Prepare design review	Perut
09/28/15	**Design Reviews & Complete Safety Training (Wed)**	All
	Purchase hardware & all parts, research on neural network	Pichamon
	Collect preliminary sensor data for neural network training	Perut
10/5/15	**Soldering Assignment (Fri)**	All
	Study datasheet for sensor, microcontroller, bluetooth modules	Pichamon
	Program microcontroller	Perut
10/12/15	Assemble location and pressure sensors module	Pichamon
	Test microcontroller	Perut
10/19/15	Test sensor accuracy and design PCB	Pichamon
	Set up host service	Perut
10/26/15	**Individual Progress Reports Due**	All
	Test PCB	Pichamon
	Test host service and implement front-end application	Perut
11/2/15	**Mock Demo**	All
	Prepare mock demonstration	Pichamon
	Black-box testing on individual components	Perut
11/9/15	Assemble all components	Pichamon
	Run tests on final project as a whole	Perut
11/16/15	Ensure functionality	Pichamon
	Fix remaining issues	Perut
11/23/15	Prepare presentation	Pichamon
	Prepare demonstration	Perut
11/30/15	Prepare final paper	Pichamon
	Finalize project demonstration	Perut
12/7/15	**Final Papers Due**	All
	Finalize presentation	Pichamon
	Lab checkout and finalize final paper	Perut

Table 11. Project Schedule and Task Allocation

5 References

[1] *IEEE Code of Ethics.* Retrieved February, 2015. Available:
http://www.ieee.org/about/corporate/governance/p7-8.html

[2] *Polymer Lithium-Ion Battery Product Specification* [Online]. Available:
http://www.adafruit.com/datasheets/503562%201200mah.pdf

[3] *Bit error rate* [Online]. Available:
https://en.wikipedia.org/wiki/Bit_error_rate

[4] *Li-Ion and LiPoly Batteries* [Online]. Available:
https://learn.adafruit.com/li-ion-and-lipoly-batteries/conclusion

[5] *Datasheet Bluetooth to Serial Port Module HC-05* [Online]. Available:
http://www.electronica60norte.com/mwfls/pdf/newBluetooth.pdf

[6] *Linear and Switching Voltage Regulator Fundamental Part 1* [Online]. Available:
http://www.ti.com/lit/an/snva558/snva558.pdf

[7] *PerfBoard Hackduino* [Online]. Available:
http://www.instructables.com/id/Perfboard-Hackduino-Arduino-compatible-circuit/

[8] *Atmel 8-Bit Microcontroller with 4/8/16/32KBytes In-System Programmable Flash Datasheet Summary* [Online]. Available:
http://www.atmel.com/Images/Atmel-8271-8-bit-AVR-Microcontroller-ATmega48A-48PA-88A-88PA-168A-168PA-328-328P_datasheet_Summary.pdf

[9] *SensorEvent* [Online]. Available:
http://developer.android.com/reference/android/hardware/SensorEvent.html#values

[10] *Fast Learning Algorithms* [Online]. Available:
http://page.mi.fu-berlin.de/rojas/neural/chapter/K8.pdf