NoteOn Smartpen Design Document

Nick Ames (nick@fetchmodus.org)

2014-08-19

Figure 1: Assembled Prototype

Project Goals

Current smartpens are not very convenient to use. Most are quite bulky, and all require either special paper[1] or a base station that clips on to the page.[2] NoteOn aims to be a better smartpen; one that is slim, wireless, and self-contained.

- **Slim**. Most smartpens are about 12mm in diameter, more akin to a marker than a pen. Ideally, a production version of NoteOn would be 8mm in diameter, but this would require a custom battery and four layer PCB. The current prototype is 10.0mm in diameter using an off-the-shelf battery and two layer PCB.
- **Wireless**. NoteOn incorporates a Bluetooth 4.0 transceiver to transfer data to a mobile device or computer, where it can be uploaded to an cloud service such as Evernote, Onenote, or Google Drive. A micro USB connector is used for charging and wired data transfer.
- **Self-Contained**. NoteOn uses inertial data (provided by two accelerometers, a gyroscope, and a magnetometer) to track the movement of the pen, allowing it to be used with any notebook, post-it, or napkin. Pen strokes can be stored to the internal memory and downloaded later.

Component Highlights

(A complete Bill of Materials is available in the file package, which can be found on the project page.)

- ST STM32F302K8U6 *ARM Cortex-4F microcontroller with 64kB ROM/16kB RAM.* System microcontroller.
- ST LSM9DS0TR *9-axis (accelerometer, gyroscope, and magnetometer) IMU.* Primary IMU.
- ST LIS3DSHTR *3-axis accelerometer.* Auxiliary accelerometer (used to improve the accuracy of motion sensing).
- Nordic nRF8001-R2Q32-T *Bluetooth 4.0 LE transceiver.* Bluetooth interface.
- Micron N25Q512A13GF840E *512Mbit SPI NOR Flash Memory.* Pen stroke data storage.
- Skyworks AAT3693IDH-AA-T1 *Li-Ion Battery Charger.* Battery charger.

[1] A camera in the pen tracks a pattern of dots on the page. Used by Livescribe and others based on Anotto technology.
[2] The base station tracks the pen using infrared (Wacom Inkling) or ultrasonic (Equill Jot) emissions.

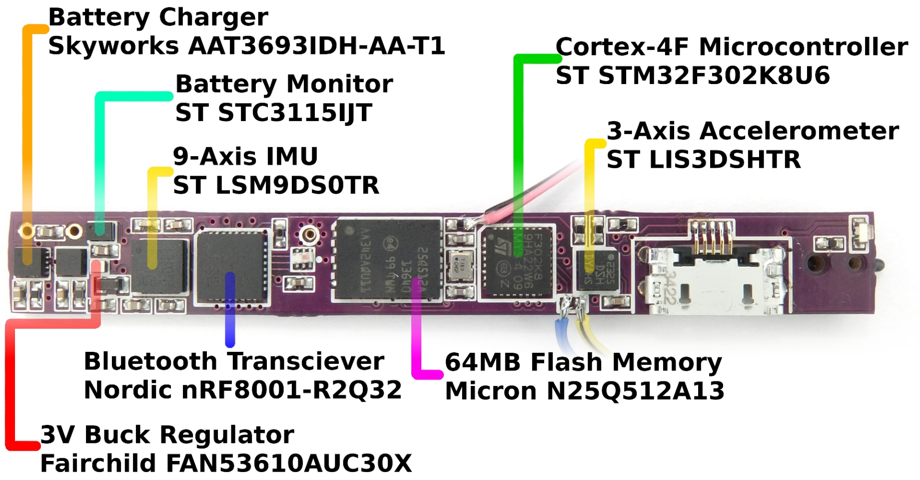

Figure 2: Component Highlights

- ST STC3115IJT *Li-Ion Gas Gauge.* Battery monitor.
- Fairchild FAN53610AUC30X *3Mhz 3V Synchronous Buck Regulator.* System voltage regulator.
- GoldPeak GP0836L17 *170mAh Li-Ion Battery.* Battery.

Status

The electronic and mechanical design is complete. Two prototype PCBs and one case have been manufactured. Work on the firmware is ongoing.

The first assembled PCB, prototype 1, was damaged by an ESD event which disabled the microcontroller's crystal oscillator circuit[3]. Prototype 2 was assembled in response, but its auxiliary accelerometer does not work[4]. I've purchased a replacement accelerometer, and it will be swapped in as soon as I can acquire a hot-air rework tool. All other parts of prototype 2 appear functional.

Development work on the firmware is proceeding using an open-source toolchain (gcc-arm-embedded with chip support provided by libopencm3). The board has been brought up, and drivers for the internal peripherals (except for USB), battery monitor, and external flash have been completed and tested. A driver for the IMU is almost complete. The next step will be to bring up the Bluetooth transceiver and demonstrate connectivity with an Android mobile phone.

System Overview

The microcontroller communicates with the peripherals using I^2C and SPI buses. The IMU, auxiliary accelerometer, and battery monitor are connected to the I^2C bus. The Bluetooth transceiver and external flash connect over SPI. The board uses a single 3V power domain, provided by a buck regulator from the battery or USB V_{bus}. The 3V rail is always on, so the off state is provided by putting the peripherals and microcontroller into their low-power modes.

The device firmware is structured around a single main task interrupted by device driver tasks. Interrupts are caused by on-chip peripheral events or timers. The IMU driver fetches data from the IMU and aux. accelerometer at a 100Hz rate and delivers it to the main task, which will perform motion calculations. (The IMU and aux. accelerometer store their data in internal FIFOs between reads.) Bluetooth and USB communication will be handled by device drivers.

The pen's user interface is simple: a button on top of the pen (connected to a momentary switch) brings it in and out of sleep mode, and a blue LED illuminates when the pen is active. Device configuration will be performed over Bluetooth or USB using a control application.

[3] I was probing the crystal oscillator when a spark jumped between the probe ground spring and the crystal capacitor. At the time, the prototype was powered by isolated power supply, which allowed a charge to build up.

[4] Checking against prototype 1's working aux. accelerometer indicates that the software is not at fault.

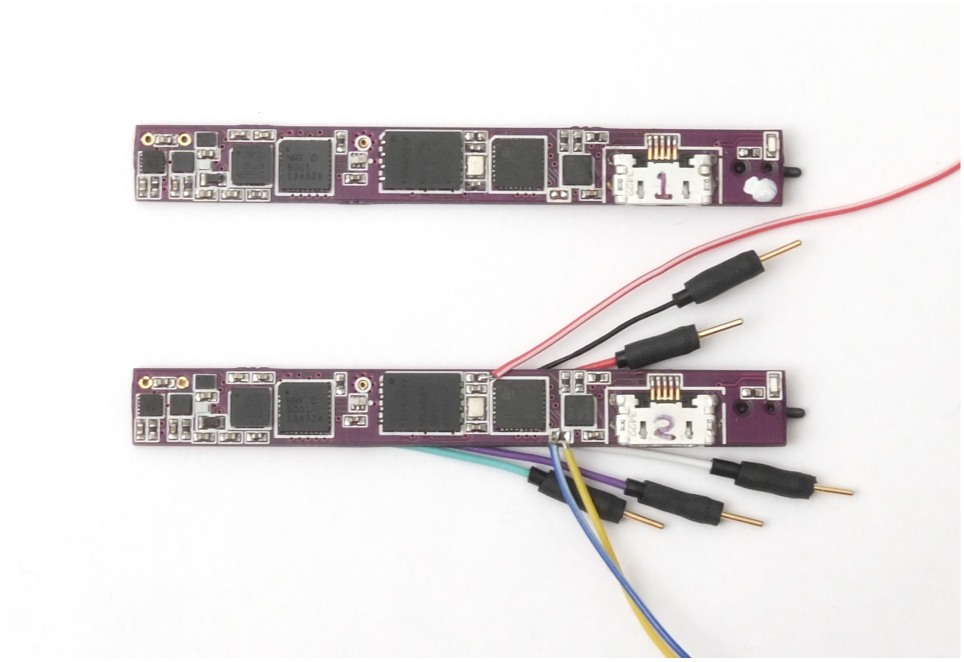

Figure 3: Prototype PCBs. Prototype 2 has wires attached to connect to the microcontroller's bootloader and probe the SPI bus.

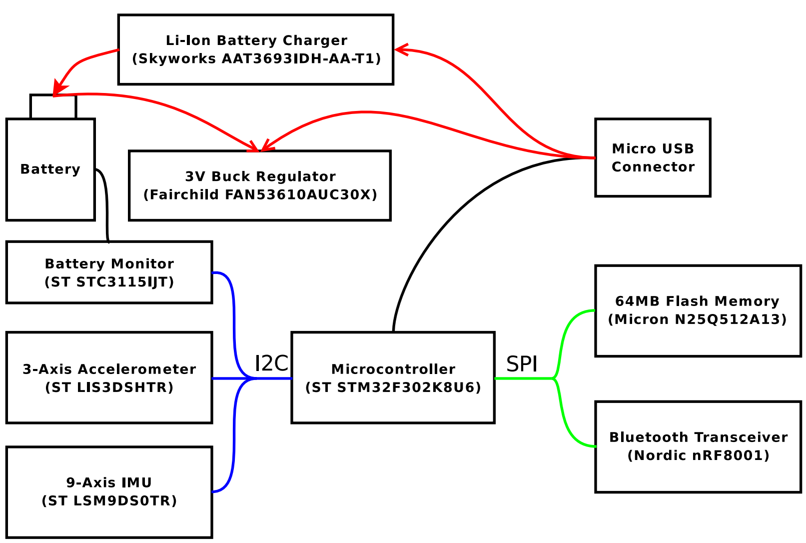

Figure 4: System Block Diagram

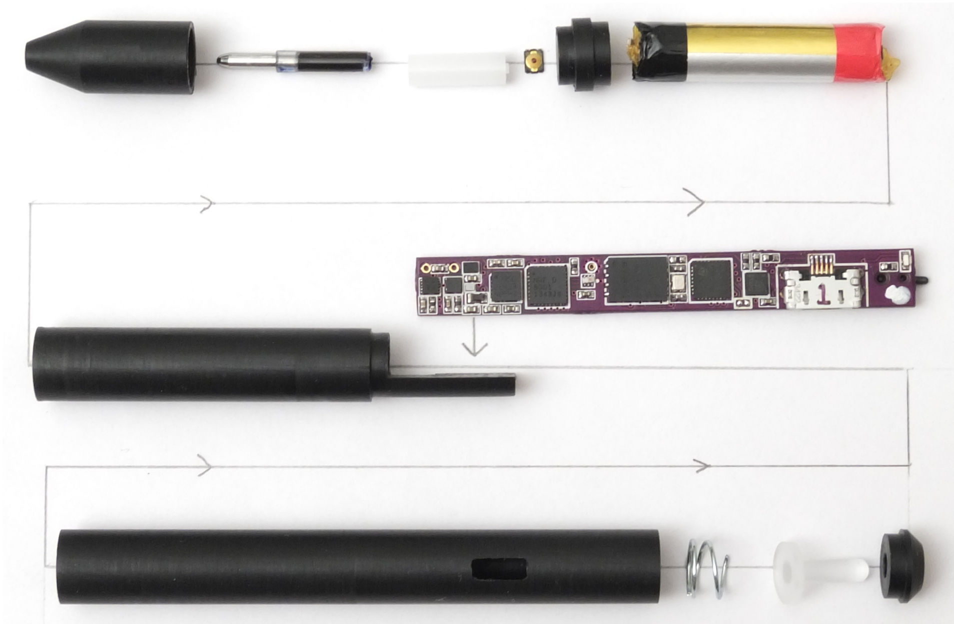

Figure 5: Prototype Case Assembly

Design Process

Making the pen as thin in diameter as possible was my main goal during design, so I chose the smallest Li-Ion battery I could find: the GoldPeak GP0836L17 (8.3mm dia. by 36mm, 170mAh). (Unfortunately, this battery is not available from GoldPeak, but clones are marketed[5] as replacements for the Sony MH100 headset.) To save space, leadless ICs and 0402 components are used. Both prototypes were assembled by hand and reflowed in a toaster oven after laying down lead-free solder paste with a Kapton stencil.

The pen casing is assembled from several pieces made of black polycarbonate, clear acrylic, and delrin. All parts were machined on my small hobbyist-grade lathe. (A milling attachment was used to cut steps and slots.) The case is divided into three compartments, which hold the ink cartridge, battery, and PCB. Refer to the photos and mechanical drawings in the file package for more details.

Prototype Flaws

There are a number of flaws in the current design. Some are due to oversights and mistakes, while others were design decisions based on limited board space and routing complexity. Although this list is long, I don't believe any of these problems will prevent the prototype from meeting the project goals.

- **High Power Consumption** The design uses a single 3V power domain, and no provision is made to turn it off when the device is in sleep. Most of the components can operate at lower voltages, so several switchable domains would be beneficial. Additionally, the regulator continues to draw power until it reaches its 2.2V UVLO level, which may damage the Li-Ion battery over time.

- **Suboptimal Trace Routing** Traces are routed underneath the IMU and aux. accelerometer against datasheet recommendations. The central pads are omitted on the memory and microcontroller footprints. While these pads are not electrically necessary, they increase mechanical strength.

- **No Matching Network on Antenna** As I don't have the knowledge or equipment to design an RF matching network, the output of the balun feeds directly into the antenna. The RF performance will probably be very poor.

- **Oscillator Output Marginal** The Bluetooth transceiver's 16Mhz input is fed from the output of the microcontroller's crystal oscillator, but the amplitude of this signal is less than the transceiver datasheet recommendation.

[5] BatteryBob #201747 and several sources on Ebay.

- **Battery Monitor Not Configured for Current Measurement** On-board current measurement would be very useful for optimizing power consumption.
- **No Real-Time Clock** The date/time of writings and sketches will not be available.
- **No Transient Voltage Suppressor on USB Port** May make the device more susceptible to ESD damage.
- **Status Light Isn't Bright Enough** While the LED brightness is adequate, the light isn't properly directed in the case.

The next design revision will use a four-layer board and smaller BGA packages for the microcontroller and Bluetooth transceiver, freeing up space for additional components.

Risks

The only substantial project risk is that current MEMS technology cannot provide enough sensitivity. This could be caused by inadequate sensor resolution or a high noise/drift level. If necessary, the pen's capabilities could be reduced to simple text transcription. In that case, character recognition would be built into the pen, and the inertial measurement system's task would be simplified to distinguishing individual letters.

Licenses

Links

Hackaday.io Project Page: http://hackaday.io/project/2678-NoteOn-Smartpen
Project File Package: http://www.fetchmodus.org/projects/noteon/NoteOn-Files-2014-08-20.zip
Firmware Repository: https://github.com/NickAmes/NoteOn-Firmware