

Results

Sawaiz Syed

May 2, 2014

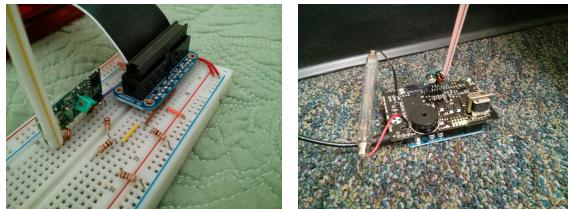
1 Results

The device, although still under development, has reached a late prototype stage. Many of the design goals set out were reached. Circuits and schematics still need to be finalized therefore testing was completed with the prototypes, Figure 1.

Power The device design has not reached the phase to focus on power consumption. The power regulators on the board consume the majority of power and therefore even an estimation of the final designs power consumption would not be possible. The unit is not able to run for elongated periods of time on batteries, although because the average current consumption is less than 100mA and is dependent on the amount of radiation it could be run from a solar panel.

Cost Although the price per module was not reduced to the expected final product's cost, the device's cost per sensor is lower than commercially available sensors. The cost per module was reduced to about \$130 per transmitter mostly representing the cost of the Arduino Geiger Shield, with the Microcontroller unit and radio transmitter as the rest of the cost.

Testing Range Range tests were conducted of the 433 MHz modules with a 173mm quarter wave whip antenna, with an indoor range of approximately



(a) Raspberry Pi RX (b) Arduino Transmitter

Figure 1: Prototype devices

10m, it can be used in close range, but would fare much better outdoors. With a longer range due to less obstruction, approximately 30m is enough to mount receivers on lamp posts in urban areas, and attach the sensing transmitters to nearby buildings. Testing of multiple devices was also done, the receiver successfully took data from two sensors and differentiated counts based on their address. Through testing, of the 13588 rows transmitted, only 14 were not received properly.

Server The server was built on two platforms based on usage scenarios. The Raspberry Pi (a small, inexpensive ARM based computer), and a Arduino. The Raspberry Pi also has the advantage of not needing anything but a 700mA power connector and an Ethernet port. This server is able to provide live data through a web page and internally through a SQL query-able database shown in Figure 2. The Arduino based implementation (Figure 3) is great for debugging and testing transmitter technologies before they need to be ported.

Timestamp	Transmitter ID	CPM
2014-04-12 15:09:05	81	30
2014-04-12 15:09:15	81	6
2014-04-12 15:09:25	81	12
2014-04-12 15:09:35	81	24
2014-04-12 15:09:45	81	42
2014-04-12 15:09:55	81	12
2014-04-12 15:10:05	81	6
2014-04-12 15:10:15	81	24

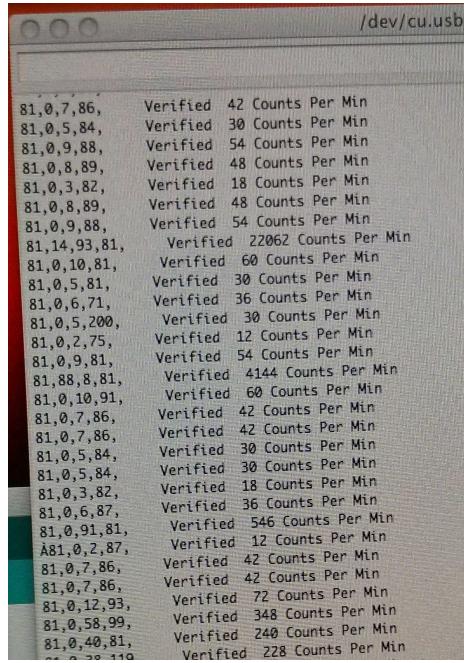
Figure 2: Web Page hosted on Raspberry Pi

Documentation All the hardware and software specifications to this point are hosted on-line under open source licenses. There is also documentation of hardware setups and code hosted on an internal Wiki for parallel testing. This documentation allows upkeep of the code-base and devices by others.

Future Prospects

The device still has a lot to be completed before it reaches the production quality needed. The prototype has shown significant promise and therefore steps to the final product should be few.

Production Preparing the device for production, with custom made circuit boards, and reducing connectors, and part counts is the best way to reduce cost. For this to take place, components need to be ordered and assembled in prototypes of circuit boards and casing.



The screenshot shows a terminal window with the title bar reading "/dev/cu.usb0". The window displays a list of data entries, each consisting of a six-digit code followed by a status and a count value. The data is as follows:

Code	Status	Count
81,0,7,86,	Verified	42 Counts Per Min
81,0,5,84,	Verified	30 Counts Per Min
81,0,9,88,	Verified	54 Counts Per Min
81,0,8,89,	Verified	48 Counts Per Min
81,0,3,82,	Verified	18 Counts Per Min
81,0,8,89,	Verified	48 Counts Per Min
81,0,9,88,	Verified	54 Counts Per Min
81,14,93,81,	Verified	22062 Counts Per Min
81,0,10,81,	Verified	60 Counts Per Min
81,0,5,81,	Verified	30 Counts Per Min
81,0,6,71,	Verified	36 Counts Per Min
81,0,5,200,	Verified	30 Counts Per Min
81,0,2,75,	Verified	12 Counts Per Min
81,0,9,81,	Verified	54 Counts Per Min
81,88,8,81,	Verified	4144 Counts Per Min
81,0,10,91,	Verified	60 Counts Per Min
81,0,7,86,	Verified	42 Counts Per Min
81,0,7,86,	Verified	42 Counts Per Min
81,0,5,84,	Verified	30 Counts Per Min
81,0,5,84,	Verified	30 Counts Per Min
81,0,3,82,	Verified	18 Counts Per Min
81,0,6,87,	Verified	36 Counts Per Min
81,0,91,81,	Verified	546 Counts Per Min
A81,0,2,87,	Verified	12 Counts Per Min
81,0,7,86,	Verified	42 Counts Per Min
81,0,7,86,	Verified	42 Counts Per Min
81,0,12,93,	Verified	72 Counts Per Min
81,0,58,99,	Verified	348 Counts Per Min
81,0,40,81,	Verified	240 Counts Per Min
81,0,36,119	Verified	228 Counts Per Min

Figure 3: Arduino Receiving Data

Radio Transmitter For longer range modules using the NRF24L01+ chip based modules, are available and suggest a more concrete hardware link, and with their engineered antennas, provide longer range.