Remote Environmental Monitoring Using HopeRF RFM69HW and Arduino

Alexander Davis
info@matrixwide.com

The Project

Monitor the following:
- Water level in a creek
- Air temperature
- Humidity

Send the data wirelessly to a computer for logging and uploading to the “cloud”.

Why Arduino?

- Plenty of well-tested libraries
- Easily understood MCU hardware
- Inexpensive
- Easy to obtain low-power operation

Why HopeRF RFM69HW?

- Inexpensive
- Available
- Good range – up to +20 dBm - 100 mW
- Choice of frequencies (315,433,868 and 915 Mhz)
- Good library support via RadioHead

Hardware

- Anarduino MiniWireless-HW Arduino Uno-compatible boards
- RFM69HW radio modules
- HTU21D I2C temperature & humidity sensor

“Ping” ultrasonic rangefinder module (from RadioShack going out of business sale)

Home Depot solar-powered “flag light” (uses a LiFePo4 battery; nominal 3.2v up to 3.8v during charge)

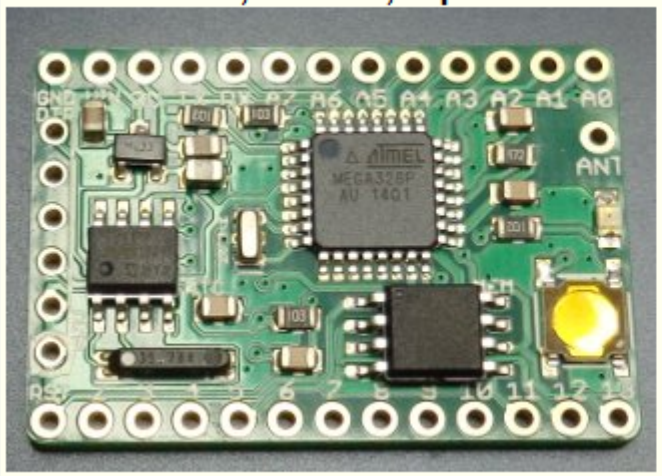
- Tape-measure yagi antennas for 433 MHz

Anarduino

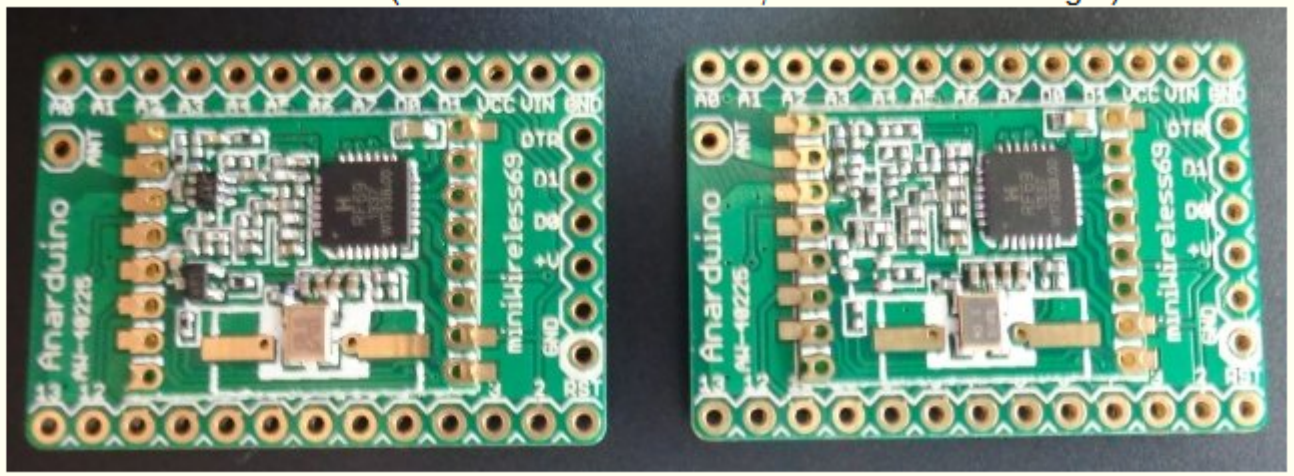
- ATmega328, 16Mhz; a bit of an overclock at 3.3v but it’s stable
- LDO Voltage Regulator (seems to drop less than 0.1v)
- 128 Mbit flash memory
- Real Time Clock (RTC) (MCP7940 – mostly DS1307 compatible)
- 64B RTC Static Ram
- Designed to mount an RFM69HW; can be ordered with or without board attached

Anarduino and RFM69HW Modules

MiniWireless-W, and -HW, Top View

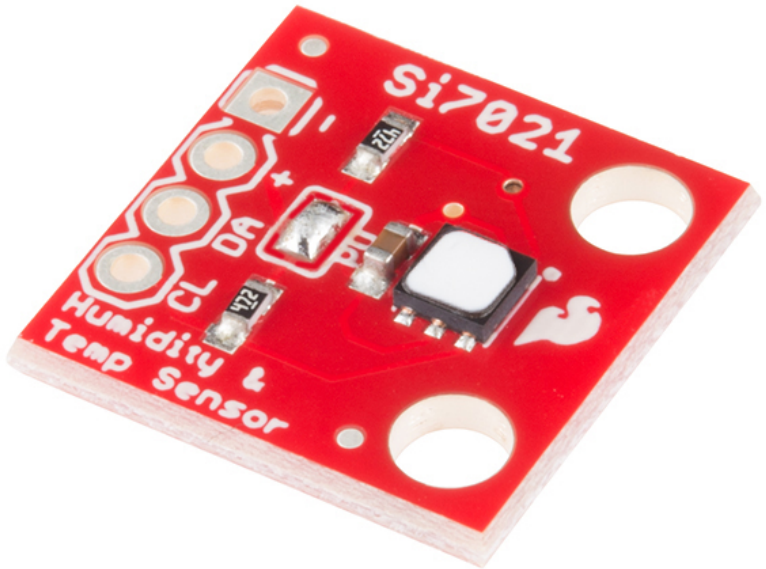


MiniWireless-HW and -W (-HW w/RFM69HW on left, -W w/RFM69W on right)



HTU21D

- https://cdn-shop.adafruit.com/datasheets/1899_HTU21D.pdf
-



Ultrasonic Ping Sensor

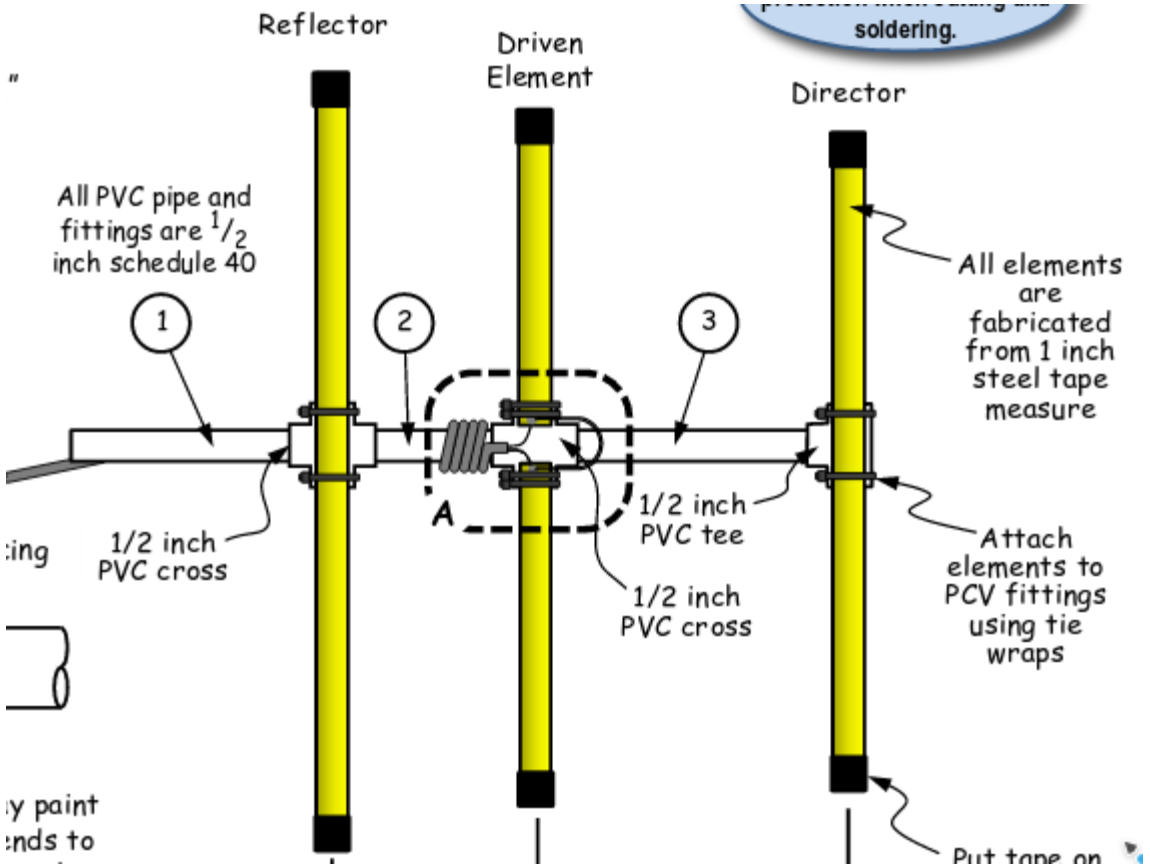
- https://www.parallax.com/product/28015#downloads
-

Power Supply

- http://www.homedepot.com/p/Hampton-Bay-Outdoor-Solar-Powered-LED-Black-Flag-Light-99940/205846412
- A123-size LiFePo4 battery
- ~6v open voltage solar panel, probably 1W or less
- Appears to charge via just a diode – obviously not the best way to do it but it seems to work
- Bypassed dark/light detection circuitry
- If it doesn’t work out (charge not maintained) I’ll go with an Energizer 528 lantern cell (25 Ah!)

Tape Measure Yagi Antenna

- http://www.nrharc.org/UHF%20Tape%20Measure%20Yagi.pdf
-



Code Basics: Sender

- Uses the RTC interrupt pin connected to Arduino pin 3 HW interrupt 1 to wake MCU; uses the RTC alarm function to enable long sleep times without resorting to WDT + cycle counting
- Controls power to ping sensor via FET
- Uses ‘union’ to pack sensor data into bytes for transmission
- Uses simple handshake to detect successful transmission

Code Basics: Receiver

- Simply waits for incoming bytes on the radio
- Acknowledges data received to sender to allow the sender to know transmission was successful
- Uses ‘union’ to convert bytes back into original sensor values
- Formats data as comma-separated and sends it out the hardware serial port

Code Basics: Server-Side

- Python
- Polls serial port for data, assigns to variables, and sends to xively.com via eeml library
- Not obvious how to use values which are not SI or SI-derived, but we'll look at the code to see how.

Github Repo

- https://github.com/quarterturn/xively