Remote Water Monitor Project

Chad A. Woitas

December 4, 2018

Remote Watering Project

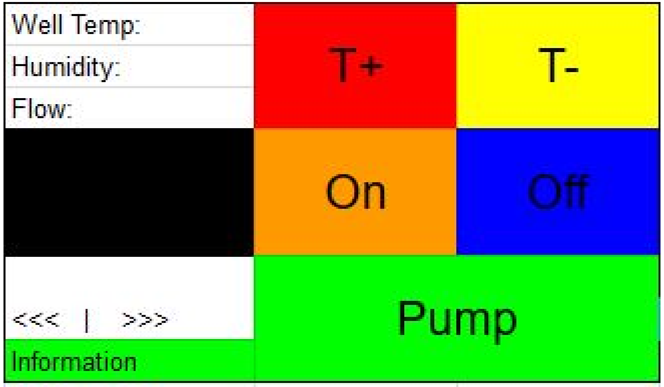
Abstract

There is a need in rural Saskatchewan to monitor and control water supply systems from deep wells. This project explores the use of LoRa radios with the CoAP protocol not only to monitor water supplies, but also other remote applications such a environmental conditions in a barn, shop, or freezer. This is being modelled as a star point-endpoints, in future with more nodes, mesh networking may be incorporated. As this project is being submitted for CMPT 436, the hardware will be described in brief while focusing mainly on the protocols and software.

0.1 Hardware

The project is prototyped on the AdaFruit Feather 32u4 micro controllers. The feather was chosen for availability of stacking headers and on board options. Each controller has an on board 915 MHz radio with a 73mm 1/4 whip antenna.

Server On the Server board a TFT screen is attached for data display and user interaction. This was intended to be powered via a USB jack. A battery supply is possible, however due to the on board Display the battery life is limited to 12 hours with a 2000mAh usb battery.



Interface on the server

Client On the node/client boards stacking headers can include servos, sensors, and relay boards. For initial testing sensors are connected to the main board, while control is maintained via a Pulse Width Modulation (PWM) header. For this prototype, one environment sensor, one flow sensor, and one control servo are attached. Battery life has yet to be determined.

0.2 Signals

This project packs data packets into 4 byte packets in the form aa00. Here the . are to denote the bytes.

The first 2 bytes denote the data information:

- c.c - Connection packets
- c.n - Node connection packets
- s.t - Temperature sensor packets
- s.h - Humidity sensor packets
- s.f - Flow sensor packets
- s.0 - Control ”0” packets

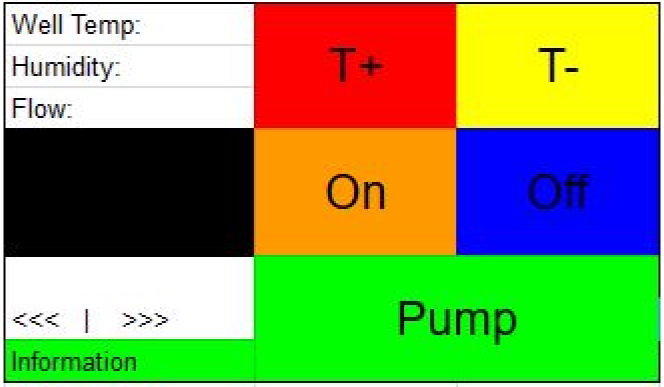
The third byte is an id of the signal. For example if there is more than 1 temperature sensor, this will distinguish between sensors and controls at the same node. This could be improved by changing them to a hexadecimal value for larger ranges.

The forth byte is the actual data.

- c.c.0.0 - Server acknowledge connection received
- c.c.0.1 - Server acknowledge data received
- c.c.0.2 - Redacted
- c.c.0.3 - Bad request
- c.c.0.4 - Reset to initial settings and clear the node memory
- c.n.0.1 - Sensor 0 moved to position 1
- s.t.0.22 - Temperature data, Sensor 0, 22degrees
- s.1.0.1 - Control data, Control 1, Data specific to the control (turn on, activate)
- s.1.0.2 - Control data, Control 1, Data specific to the control (turn off)
- s.1.0.3-255 - Control data, Control 1, move to position

0.3 User Interface

The Server node features a Touch Screen with both Sensor information and Controls.



- Data from sensors
- Buttons to send controls
- Connection information

- empty space to add elements for switch between nodes

The nodes save power by having no display. Some Micro Controllers can last for years off of a small battery using this method. To have a sense of feedback, The on board LED will flash varying intervals to indicate status.

- 3 quick blinks - error in function
- 3 erratic blinks - error in transmission
- 4 slow blinks - successful packet receive

0.4 Protocols

The original intention of this project was to analyze the CoAP protocols. However it seems the CoAP protocols have yet to make it to the realm of Arduino and LoRa. This project attempted at mimicking signals from CoAP on the RadioHead library. TODO INSERT LINK

Packets Each packet specified in this has: an 1 octet preamble, 4 octet Header with To, From, ID, Flags,0-251 octets data, and CRC checking that is handled by the hardware. From preamble, the data packets have been chosen to be 4 bytes as specified in signals. In particular GET and POST.

POST The clients pushes data to the server when it is available. Each data packet with sensor information has been denoted with an "sa" character. s- Sensor a- can be replaced with the type of sensor. The full list is provided in sensors. The server can also push control signals with acknowledgement each of these is "s00" followed by a data. 00 refers to the control id.

GET The only get requests currently are for connection tests and acknowledgement of packets received. As this is intended in rural Saskatchewan internet connectivity is not easily found there and thus most web page requests would be redundant with the on board displays.

0.5 Going Forward

Going forward as there is a lack of libraries available for CoAP on LoRa boards, I'm looking into writing these, either as an extension to RadioHead, or establishing a universal lightweight for P2P applications. This project will be fully installed into a functioning system in the spring time, with a high probability of expanding to multiple monitoring nodes across the farm.