Setting up the Raspberry Pi based field sensor monitor
August 2016

Contents

1 Introduction

This document describes how to set up and use the field monitoring system based on a Raspberry Pi base station and Moteino/Arduino based loggers. It provides a description of the hardware used and how to set up specific parameters in software, such as logging frequency.

2 What you need

Arduino IDE

3 Description of hardware

3.1 Base station (Raspberry Pi)

3.2 Logger (Moteino)

4 Using the software

4.1 Base station (Raspberry Pi)

The SD card image provided has everything already set up, but there are some more details here if you need to change anything. If you just want to set up a new Pi with the provided image, go to **Setting up the Pi: Simple**.

The Raspberry Pi is set up so that it automatically runs a script when it starts. This script listens for any input from the controller and then captures and saves an image using the camera. Then it shuts down the pi. This script is located in */home/pi/scripts* and it runs from */etc/rc.local*. Scripts run from *rc.local* run as the "root" user (similar to Administrators in Windows). You can look at this file by typing *cat /etc/rc.local*. The Raspberry Pi is turned off and off by the controller; it turns the Pi on, waits for it to complete its tasks, then turns it off.

4.1.1 Setting up the Pi: Simple

4.1.2 Setting up the Pi: Complicated

4.2 Logger (Moteino)

The logger is the Moteino with the light sensor array attached. It runs from 2 x AA batteries and has a low power radio to send data back to the controller. The controller that the logger is sending data back to must have the same *NET-WORK_ID* as the logger, or they can't communicate.

When setting up a new logger, there are a few things that you need to change, and a few things that you can change if you want to. As a minimum, you need to:

- Check that the logger is on the right network, so that it can communicate with the right controller

- Check that the logger's *LOGGER_ID* is unique on it's network

- Check that the logging frequency is correct

4.2.1 Radio parameters

LOGGER_ID The *LOGGER_ID* is the unique number of the logger. All loggers with the same *NETWORK_ID* need to have different *LOGGER_ID*'s. This is partly so that when you look at the resulting data, you will know what logger it came from. It also allows the radios to identify each other, and to prevent clashes between   loggers.

The logger ID is transmitted to the controller along with the data from the logger. Because there is a limit to how much data can be transmitted at once, there is a limit to how big the *LOGGER_ID* can be, because it is only allowed to take up one byte, or eight bits. The highest number you can represent with eight bits is 255. So, including zero, we can have up to 256 unique *LOGGER_ID*'s on one network (networks work the same way). One of these numbers (the number 1) is actually taken by the controller, and the number 255 is used as a "broadcast", meaning that all loggers on the network can see data transmitted to 255.

So, when you are setting up a new logger, set the logger ID to soemthing greater than 1 and less than 255, and make sure that no other loggers with the same *NETWORK_ID* have the same *LOGGER_ID*.

NETWORK_ID

GATEWAY_ID

FREQUENCY

ENCRYPTKEY

ENABLE_ATC

4.2.2 Logging parameters

NUM_PHOTODIODES