Final project – Micro-greenhouse Microcontroller project.

CPE185 Lab, Spring 2018

Sean Kennedy

Eric Telles

Lab Days (Tuesday Morning, Wednesday Nights, & Thursday Nights)

Jeremy Shaw – Wednesday Night (http://bitbucket.org/jeremyshaw/jshaw)

Vadim Babiy – Wednesday Night (http://bitbucket.org/vadimbabiy/vadimcpe185)

Wesley Nguyen – Tuesday Morning (http://bitbucket.org/wnguyen370/wnguyen370)

Daniel Bracamontes – Thursday Night (http://bitbucket.org/smoothrhythms/micro-greenhouse)

Shaw! Babiy! Nguyen! Bracamontes!

We are Team Valor!

1 Micro - Greenhouse control system

2 Project Description

This project aims to create a control system for a tiny, plant-based ecosystem. It will be a system that monitors the relative health of the plants (humidity, water levels, light levels), disperse necessary materials to support plant life (water), mechanically controls the environment (fans and heaters, basically), and reports on all taken actions (present/upload data). This is a comprehensive project which utilizes microcontrollers, microcontroller peripherals, and microcontroller code to observe and care for a small ecosystem. A Central Controller is a device which is capable of unifying various microcontrollers in this project by logging their sensor data and sending commands for the microcontrollers to execute.. The Central Controller will also provide an user interface to interact with the various microcontrollers.

3 Main Sub-Functions

3.1 Ecosystem monitoring – Bracamontes

This is an overall system of sensors which monitor the temperature, humidity, and light levels.

<u>Requires:</u>

1) Temperature sensor

2) Light Sensor

3) Microcontroller to read and report on sensor data (e.g, Arduino, MAX32, Propeller, etc)

4) Other potential sensors

   a) Humidity

<u>What needs to be done:</u>

Develop a program to interface with the sensors, process the data, and send a concise, formatted data output via I2C to the Central Controller

<u>Outcome:</u>

An Ecosystem monitoring system capable of reliably reporting data via I2C to the Central Controller.

3.2 Plant care – Shaw

This is an electromechanical system which adjusts the amount of light and water to the plant(s), based on information gathered from the ecosystem monitoring functions.

Requires:

1) Water container and servo controlled valve/pump to disperse water

2) Lights

3) Relay to power lights and servo/pump

4) Microcontroller to control servo/pump and servo (e.g, Arduino, MAX32, Propeller, etc)

What needs to be done:

Develop a program to use an electrical relay to control a water pump (to disperse water) and a light (to provide necessary light), using instructions received via I2C from a Central Controller.

Outcome:

A Plant Care system capable receiving and executing commands from the Central Controller.

3.3 Environmental care – Nguyen

This is an electromechanical system which attenuates the environmental factors of the ecosystem, adjusting humidity, heat, airflow, etc via means of controlling heating elements, fans, and vents.

Requires:

1) Fan

2) Heat lamp / heat source

3) Relay to power fans and heat source

4) Servo to control vent

5) Microcontroller to control fan and heat source (e.g, Arduino, MAX32, Propeller, etc)

What needs to be done:

Develop a program to use an electrical relay to control a fan (to provide air/humidity control) and a heat source (to provide heat/humidity control), using instructions received via I2C from a Central Controller.

Outcome:

An Environmental Care system capable of adjusting the general environment of a greenhouse, based on commands received from the Central Controller.

3.4 (Central Controller) Data gathering and reporting – Babiy

A system which collects and reports on the overall health of the ecosystem, and individual plants. Also monitors the overall level of consumable supplies (water, electricity?). This may use sensors from the ecosystem monitoring function, along with other sensors and means of collecting, formatting, and reporting data.

Requires:

1) Device with I2C interface, data storage, and data processing abilities (e.g, Raspberry Pi)

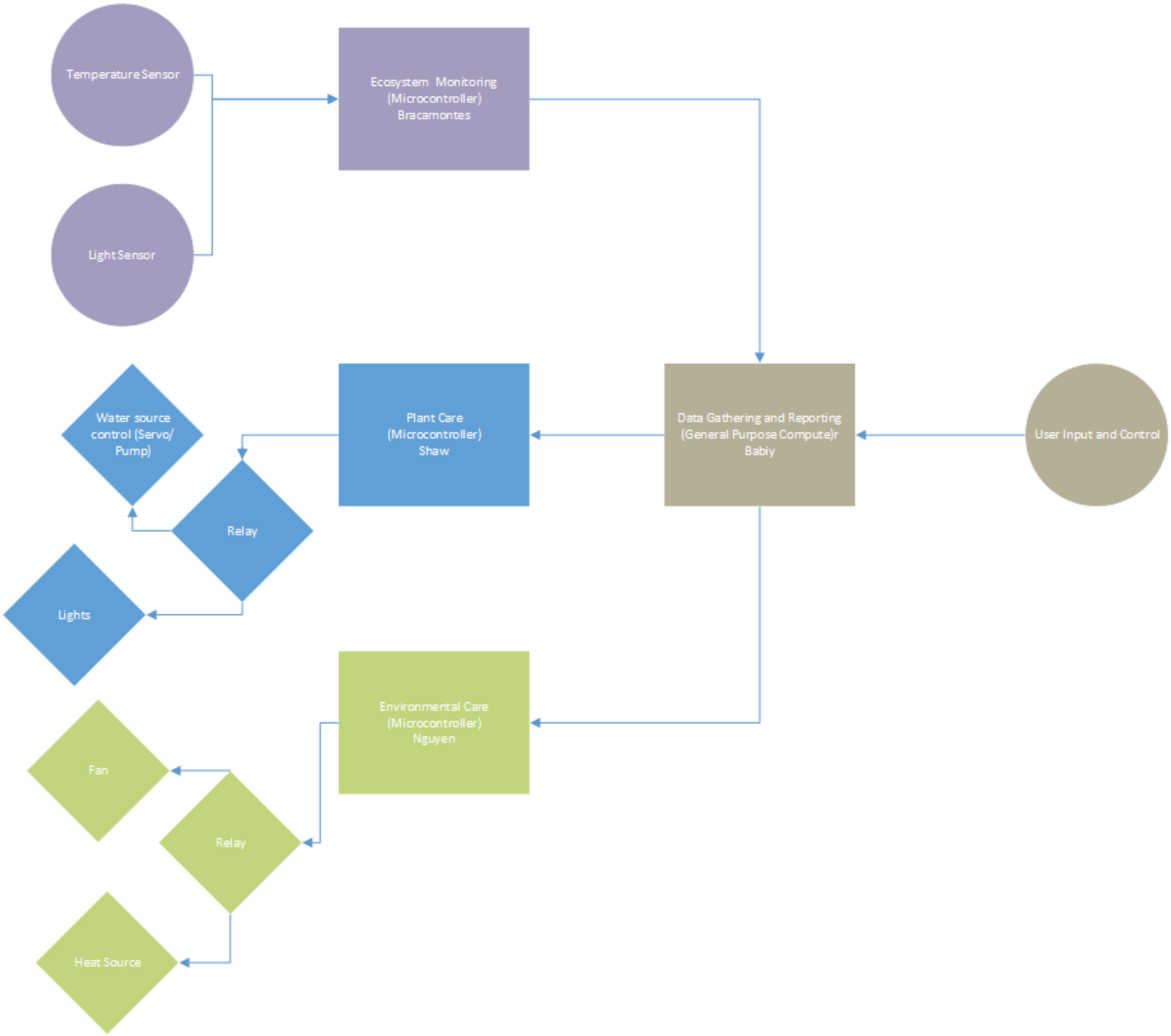
What needs to be done:

Develop a Central Controller, which receives formatted sensor data from I2C devices and puts it in a formatted log for user consumption. User is also able to give commands to Central Controller, to direct the controller to activate certain elements of the Plant Care and Environmental Care systems. The Central Controller will also be capable of automatically directing the two Care systems, in the event sensor data received from the Environmental Monitoring system exceeded safety bounds (bounds to be determined).

Outcome:

A Central Controller capable of providing user interaction with various Greenhouse systems, providing semi-autonomous safety limits, and providing a data log.

4 Block Diagram of Project



5 Recommended Labs

Bracamontes

Lab 6) Create a microcontroller program to interface with w Temperature Sensor and report the values via I2C. This will require a temperature sensor, a microcontroller, and a device to receive and display the I2C data. The microcontroller must be setup as a slave transmitter.

Lab 7) Create a microcontroller program to interface with a Light Sensor and report the values via I2C. This will require a Light sensor, a microcontroller, and a device to receive and display the I2C data. This receiving device must be different from the device used in Lab 6. The microcontroller must be setup as a slave transmitter.

Labs 8-10) Student choice.

Nguyen

Lab 6) Use a microcontroller to control a servo and a fan via a relay. This will require a microcontroller, a relay with two outputs, a servo, and a fan.

Lab 7) Configure Nguyen Lab6 to have the servo control a vent, used to control airflow to the fan. The microcontroller determine what to do with the devices, based on input from I2C. The microcontroller must be setup as an I2C Slave Receiver and Slave Transmitter (if possible), to report the status of the servo and fan.

Lab 8) Create a logical, simulated PLC which ensures the fan will never activate when the vent is closed. Use either sensor data or control data to determine the status of the vent, and the fan.

Lab 9) Add to the microcontroller project in Nguyen Lab 7, a relay controlled heat source, which is integrated into the microcontroller's I2C (Slave).

Lab 10) Student Choice

Shaw

Lab 6) Use a microcontroller to control a servo/pump and a light via a relay.

Lab 7) Configure Shaw Lab 6 to allow the microcontroller to receive commands from I2C (microcontroller must be setup as I2C Slave, with receive capability). These commands will control actuation of the servo/pump and the state of the light.

Lab 8-10) Student Choice

Babiy

Lab 6) Using a device capable of interfacing with I2C, generating a video output, and accepting keyboard commands (“Control Center”), create an I2C Master that polls a target I2C Slave for information, and saves the data to a file.

Lab 7) Using Babiy Lab 6 and Shaw Lab 7, modify Babiy Lab 6 to enable the “Control Center” to issue commands to Shaw Lab 7’s device. The commands will be interpreted from keyboard input, and sent via I2C Master.

Lab 8 - Optional) Create an user interface to integrate all aspects of Babiy Lab 6 and Babiy Lab 7.

Labs 9-10) Student Choice