

Project Proposal

Intelligent Plant Monitoring System

February 20, 2007

Ya-Shian Li, yashian@gmail.com

Abstract

The objective of the project will be to design, implement, and evaluate a networked house plant monitoring system. The system will alert plant caretakers to water their plants, adjust the temperature/humidity, and light intensity according to specification entered by the user. The web server will also be able to intelligently analyze data and growth trends to suggest and predict optimal tolerance levels based on certain conditions sensed in the specific home. The monitoring system will be based on an embedded Java system running a web server to allow the user to enter the precise moisture, temperature and humidity range preferred by a specific type of plant. When the sensor detects a parameter which is outside of the user's desired tolerance window for a specific plant, the device will automatically alert the user on the web interface and/or audio alerts based on the user preferences. The user can choose to have dynamically refreshed web interface with a graph of the data trends such as temperature and humidity as the data is being collected. The data collected should be able to be downloaded for users to perform future analysis. Possible extensions to the initial project include tracking and correlating growth with environment conditions. The user can also enter growth data to have the system calculate optimal growth conditions.

Strategy

The house plant monitoring system will be based on the Systronix TStik, which uses the TINI embedded Java platform.

The capabilities include a temperature and humidity sensor with the ability to add, modify, and track optimal conditions for a specific type of plant. The sensors will need to be evaluated for their accuracy and cost as well as its ability to interface with the TStik. A web server will be deployed on the TStik. The possible web servers which may be used include Tynamo (<http://tynamo.qindesign.com/docs/tini/index.html>).

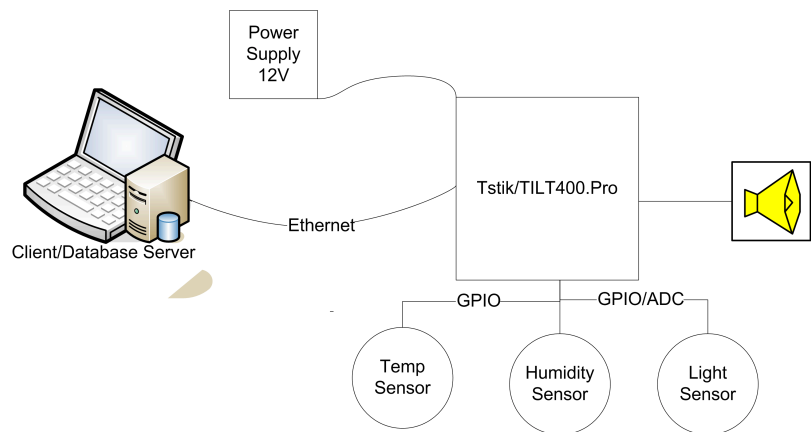
An audio alert will also be used, but can be muted automatically based on the time of day. For example, if the user prefers not to be alerted between 12 a.m. and 7 a.m., the audio alerts will sound only before and after the scheduled silence times. In addition to configuring the plant preferences, the user can choose to configure the frequency and type of data collection from the three different sensors. For example, a user can choose to retrieve a single data point or have the system take the data from the plant's environment as frequently as once a millisecond (just because it is capable) to once a day.

For data collection, the plant monitoring system will use GPIO interrupts and timers to trigger the system to collect data at varying frequencies. Depending on user configuration, the system will also trigger an interrupt to transmit data when the number of data points has reached the data buffer size. The system will also need to be able to handle analog to digital conversion for analog sensors such as the light sensor.

The system will utilize interrupt prioritization to provide a responsive system without compromising data collection quality. For example, while collecting data from sensors at frequent intervals, the system will buffer the data points and wait until collection is completed before transmitting the data to the database. To ensure a more deterministic level of quality of service, the user can choose to disable module to database transmits during data collection. Otherwise, the module will buffer a specified number of data points before transmission to the data base server (e.g. 10 - 100 data points) or dynamically refreshing the web page showing the graph of the buffered data points.

Each data point collected by the sensors will include a real-time time-stamp with at least a 1 ms resolution. The resolution, adjustment and synchronization of the TINI clock

will be configurable by the user through the web interface. The time stamps will be used to correlate trends in temperature and humidity. Users will also be able to chart daily plant growth data, which along with the sensor data, can be correlated and analyzed to provide the system with capabilities to suggest to the user more optimal temperature and humidity tolerance windows if necessary. For example, if the user entered in a tolerance window of 75 to 85 degrees F with 70-80% ambient relative humidity for her orchid plant, the plant monitoring system may suggest a few days later to narrow the tolerance window to 80-85 degrees F with 78-80% relative humidity. It is up to user if she would accept the new tolerances based on the system's suggestions.

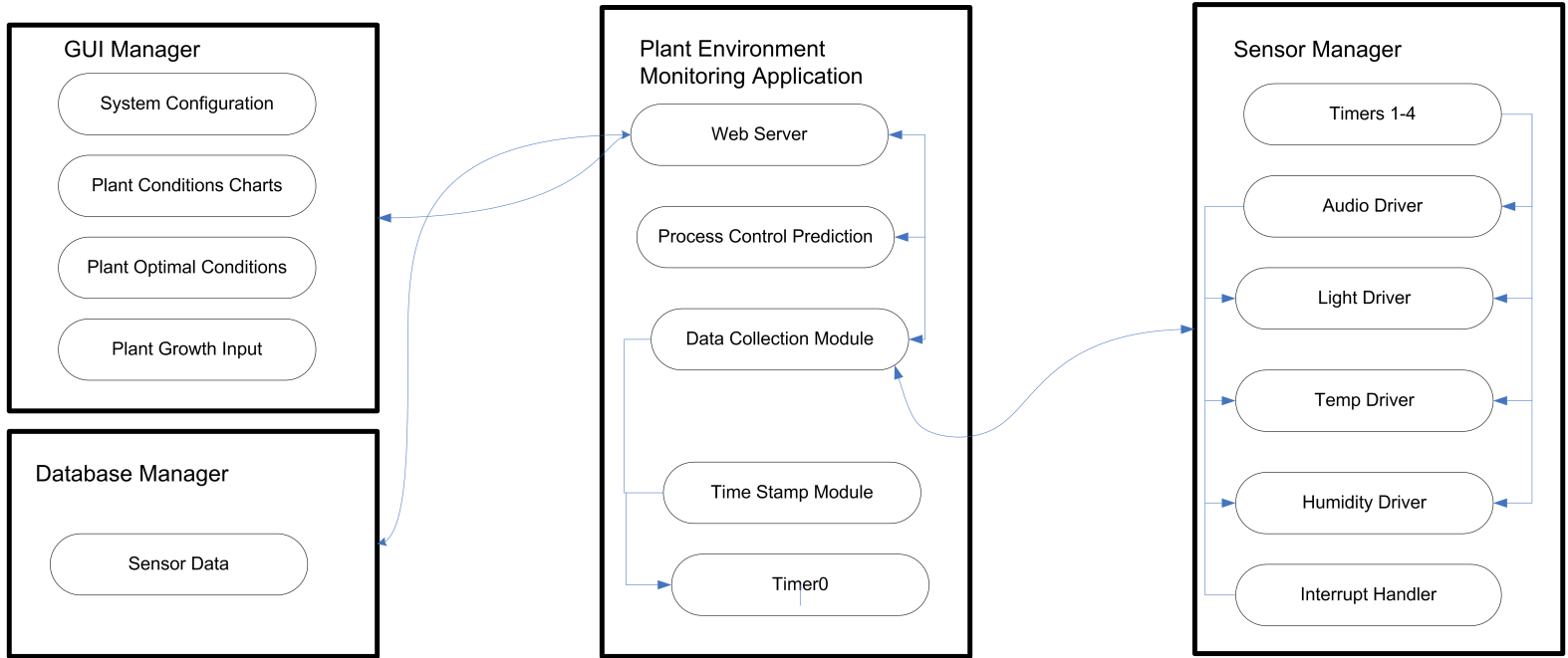


Preliminary Hardware Diagram

A MySQL database server will be deployed on a Macintosh PowerBook G4, which the TStik can access over the network to display graphs of temperature and humidity trends based on data stored in the database. The networked database servers allow more data points to be stored and provides easier access via a relational database and alleviating data storage burdens from the TStik module.

The software components necessary to complete this project will be a web server application, with a web-based user interface for user configuration of the plant monitoring system, entering growth data as well as the GUI for tracking the plant growth parameters. A database connector will need to be written to interface between the web server application and the database, an audio manager to trigger sounds when the sensors detect specific parameters are out of range, a temperature sensor API, a humidity sensor API, and a light sensor API. A plant process control module will also be implement to provide predictive environmental tolerance windows to the user. The plant monitoring system application will need to be extensible to allow future types of data correlation analysis methods and user configuration/input to be readily incorporated. Additionally, other features to be logged by the user, such as fruit growth and flowering can also be

future extensions to the system. Therefore abstraction layers will need to be developed for rapid integration of intelligent correlation components as well as the IO device components for adding new sensor devices, camera, etc.



Preliminary Software Diagram

Unknowns

It is uncertain when several components would be acquired. For example, the TILT board for the TStik Development Kit was sold out at the time of order, however after contacting their sales department, Systronix said the shipment would arrive soon. Additionally, once the development kit arrives, it would help in the decision of which web server to deploy based on ease of use and performance on the TStik TINI platform.

Understanding how to interface the various sensors and audio component will need to be addressed in terms of both hardware and software. How to interface between the TStik will be determined based on the datasheets, documentation on the various sensor interfaces, as well as experimentation when the components arrive. How to evaluate whether the components are working and how to test will also need to be determined. The data acquired by the new sensors will be compared with a regular thermometer and a humidity measurement tool (hygrometer). It is also unknown how the light sensor will measure data, whether it is by light intensity and duration or simply light duration depending on the difficulty of retrieving and converting the analog signals. Most likely, all the sensors will be 1-wire due to ease and robustness.

Several software components are also unknown. At this point, Tyanmo appears to be a viable solution, however other web servers will also need to be researched and tried, if necessary. The method of integrating the sensors using the different protocols will also need to be researched.

Additionally the memory and processing requirements of the web server and data collection on the TINI is currently unknown. This information would help to determine the number of data points the TStik module can buffer before running out of memory as well as the GUI features.

Methods for calculating the optimal tolerance windows for growth will need to be further researched. Initially, the thought would be to utilize a correlation analysis method, as the focus of the project is on embedded systems. Therefore the majority of the system will be focused on developing the sensor and data collection capabilities.

Implementation Plan

In order to construct the project, the following milestones will need to be completed.

Milestones	Tasks
<p>Acquire all hardware and software materials and become familiar with component datasheets and communication protocols. (February 28)</p>	<ul style="list-style-type: none"> • Research availability of appropriate sensors which can interface with TStik using 1-wire, I2C, or CAN • Order sensors. • Understand the different sensor interfaces and their pros and cons, how it works, how to interface by reading data sheets. • Test hardware components as they arrive. • Download software components to become acquainted with the applications. Attempt to find the best web server applications for TINI and for the purpose of the project. • Set up a MySQL RDBMS.
<p>Deploy a web server application on TINI with access to real-time clock capabilities (March 12)</p>	<ul style="list-style-type: none"> • Install a web server on TINI. • Enable network connectivity via Ethernet between laptop and TStik/TILT board. • Test connectivity and web server using laptop. • Enable capability to access real-time clock and set/synchronize the clock via the web.
<p>Integrate the temperature sensor and audio alerts. Prepare project status report. (March 20)</p>	<ul style="list-style-type: none"> • Create a software driver to interface with the temperature sensor. • The Temperature Sensor API should have the following capabilities: <ul style="list-style-type: none"> • Read temperature from the sensor. • Based on user configuration, generate periodic interrupts (GPIO timers) to read temperature data from the sensor. • Temperature data should be handled by the application as follows: <ul style="list-style-type: none"> • Time stamp each incoming data point at the resolution specified by the user. • Update web page of plant status. Issue warnings and alerts if temperature is outside of desired range. • Buffer data and upload to database server based on user configuration. • Connect 8 ohm speaker to the TILT board. • Create an audio output driver. • When temperature is outside of the user-configured tolerance window, play a tune.

Milestones	Tasks
Integrate the humidity sensor. (April 1)	<ul style="list-style-type: none"> • Implement a Humidity Sensor API <ul style="list-style-type: none"> • Read temperature from the sensor. • Based on user configuration, generate periodic interrupts (GPIO timers) to read data from the sensor. • Humidity data should be handled by the application as follows: <ul style="list-style-type: none"> • Time stamp each incoming data point at the resolution specified by the user. • Update web page of plant status. Issue warnings and alerts if humidity is outside of desired range. • Buffer data and upload to database server/dynamically refresh web page based on user configuration.
Integrate a light exposure intensity/duration sensor. (April 8)	<ul style="list-style-type: none"> • Determine how to interface photodiode to the TStik. • Implement a Light Sensor API <ul style="list-style-type: none"> • Read light intensity/duration from the sensor. • The light sensor should only be scheduled for specific hours of the day or should be automatically triggered based on a significant change (between night - morning and dusk to night as well as any weather changes - cloudiness, sunshine) • Light exposure data should be handled by the application as follows: <ul style="list-style-type: none"> • Time stamp each incoming data point at the resolution specified by the user. • Update web page of plant status. Issue warnings and alerts if humidity is outside of desired range. • Buffer data and upload to database server/dynamically refresh web page based on user configuration.
Implement plant process control module for “intelligently” predicting optimal temperature/humidity tolerance windows. (April 17)	<ul style="list-style-type: none"> • Research simple correlation techniques. • Database connector API to transmit and retrieve buffered/historical data. • Implement plant process control module. Add logic to correlate environmental parameters from sensor data with growth data entered by the user.
Test, debug and final evaluation of product. Finalize project paper. (April 22)	<ul style="list-style-type: none"> • Develop evaluation methods and demonstration scenario. • Test system based on evaluation methods and prepare a dry-run for the class demo. Draft paper to detail implementation, changes and issues.

Resources

The following hardware will need to be acquired for the final project (**The items in bold have not yet been acquired**):

- Laptop (Macintosh PowerBook G4)
- **TStik Development Kit** (TStik, Tilt400.pro) (<http://www.tstik.com/documentation.html>)
- **120 VAC Input/12 VDC Output Power Supply**
- Ethernet Cable (**straight** or crossover)
- Serial Cable (borrowed from the University's Z8 Development Kit)
- Audio speaker (borrowed from the University, the one used for Lab 2)
- **Temperature sensor (1-wire ... maybe iButton)**
- **Light sensor (photodiode 1-wire ADC)**
- **Humidity Sensor (1-wire)**

The project will also require the following software:

- TINI SDK
- TStik API
- JavaKit
- Latest Dallas TINI Firmware/OS, Slush
- **Tynamo Web Server for TINI**
- MySQL Relational Database