BioController

Copyright (C) 2013 Jeff Davies

This program is free software: you can redistribute it and/or modify

it under the terms of the GNU General Public License as published by

the Free Software Foundation, either version 3 of the License, or

(at your option) any later version.

This program is distributed in the hope that it will be useful,

but WITHOUT ANY WARRANTY; without even the implied warranty of

MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the

GNU General Public License for more details.

You should have received a copy of the GNU General Public License

along with this program. If not, see <http://www.gnu.org/licenses/>.

# Table of Contents

Contents

[1) Table of Contents 1](#_Toc365188569)

[2) Description 1](#_Toc365188570)

[3) General System Architecture 3](#_Toc365188571)

[4) Naming Convention 4](#_Toc365188572)

[Android Java Classes 4](#_Toc365188573)

[Android Layout XML objects 4](#_Toc365188574)

[5) Bill of Materials (physical items) 4](#_Toc365188575)

[6) Testing and Compliance 4](#_Toc365188576)

# Description

The BioController low cost Bioreactor controller project comprises of the following:

1. Android application (compatible with Android 2.1 through 4.0)
2. Microcontroller Program
3. Microcontroller hardware including I/O (valves, pumps, sensors)



BlueTooth

The Controller (the Microcontroller and associated IO) in general operation runs a recipie which, using feedback loops, controls the pH, DO, Temperature and various other feeds/impeller/sparges (supply of gas) to the bioreactors for the intention of growing cells such as “Cell Culture” generally using programmed Cell lines like CHO or HeLa to manufacture proteins for use in pharmaceutical drugs.

Additional functionality is Calibration by an Engineer, and Recipe creation by a Recipe Manager.

Apart from in Simulator mode, the Android device does not local control system state. It retains some system state variables in Preferences, but the actual security when not in Simulator mode is managed by the Controller, as is data storage, and recipe-following.

If an Operator or Recipe Manager or Engineer has started a Batch using a Recipe, then the MicroController looks at the recipe every minute and loads the appropriate setpoints for feedback loops and pumps/valves/motor-speed at the appropriate time, meanwhile recording the sensor readings and any Event Marker comment made by the user.

# General System Architecture

Local Microcontroller VGA and PS/2 Keyboard Console (optional)

SD Card Storage Binary files (on Microcontroller Controller device)

BlueTooth

~~MIPS R4K~~ ARM Microcontroller control application

Android HMI application

(set of Activity objects with helper class objects instantiated as required)

SQLLite database to persist Service state (running on Android device)

Android Service

Simulated Controller Service

(running on Android device)

HMI state stored in Android Preferences file

# Naming Convention

## Android Java Classes

Note: This breaks java convention of all classes starting with

a capital, but it makes it easier to see the classes that are activities

activity classnames start with "a".

all classes for utility start with "c".

all data only classes start with "d".

all interfaces start with "i"

all services start with "s"

## Android Layout XML objects

all button names should start with a "b".

all text view names start with "tv"

all EditText names start with "et"

# Bill of Materials (physical items)

PCB design for the Sensor board & MUX in Eagle CAD

Viable cell density probe – optical or impedance.

atlas scientific DO board + sensor

atlas scientific pH board + sensor

~~CGCOLORMAX1 microcontroller board (PIC32 = MIPS R4K core 32 bit RISK) running MMBASIC.~~ Arduino Due

Relay shield

Motor Control shield

TODO later: hardened versions of all boards? (based on MTBF calculations)

# Testing and Compliance

CE Compliance, self heating calculations, EMC testing and making of a technical file.

Validation and the generation of a validation pack with records of protein production using a sample programmed CHO cell line plus offline methods comparison data, analysed with a statistically sound correlation.