Tombola Controller

Software Functional Requirements

By **Jesse Zondervan and Gary Twinn**

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |

# Contents

[Tombola Controller 1](#_Toc149235053)

[Contents 2](#_Toc149235054)

[List of Figures 3](#_Toc149235055)

[List of Tables 4](#_Toc149235056)

[Overview 5](#_Toc149235057)

[Web interface 6](#_Toc149235058)

[Interfaces 7](#_Toc149235059)

[RS485 Specification 7](#_Toc149235060)

[Operating System Installation 8](#_Toc149235061)

[Operating system 8](#_Toc149235062)

[Software installation 8](#_Toc149235063)

[Run Operating System Updates 8](#_Toc149235064)

[Install PIP3 for Python 3.x installation 8](#_Toc149235065)

[Install Flask installation 8](#_Toc149235066)

[Download Tombola Controller Application 8](#_Toc149235067)

[Copy the python files to the Pi 9](#_Toc149235068)

[Nginx installation 10](#_Toc149235069)

[Gunicorn for Python 3.x installation 10](#_Toc149235070)

## List of Figures

[Figure 1: Web status page 6](#_Toc149235071)

## List of Tables

**No table of figures entries found.**

# Overview

The Tombola Controller software consists of a python ([Python Software Foundation, 2020](#_ENREF_1)) application to control a 3 phase electric motor via a Siemens V20 single phase to three phase invertor.

The computer that controls the valves is a Raspberry Pi 4B ([Raspberry PI Foundation, 2020](#_ENREF_2)), running the latest Debian operating system (Debian 1:6.1.54 “bookworm”).

The Raspberry Pi Communicates with the Siemens V20 using an RS485 controller connected to /dev/ttyUSB0 at 9600 baud. The command are MODBUS commands which consist of setting registers to set values for the motor such as rotational speed or direction and reading registers such as frequency and voltage.

RPM Interface

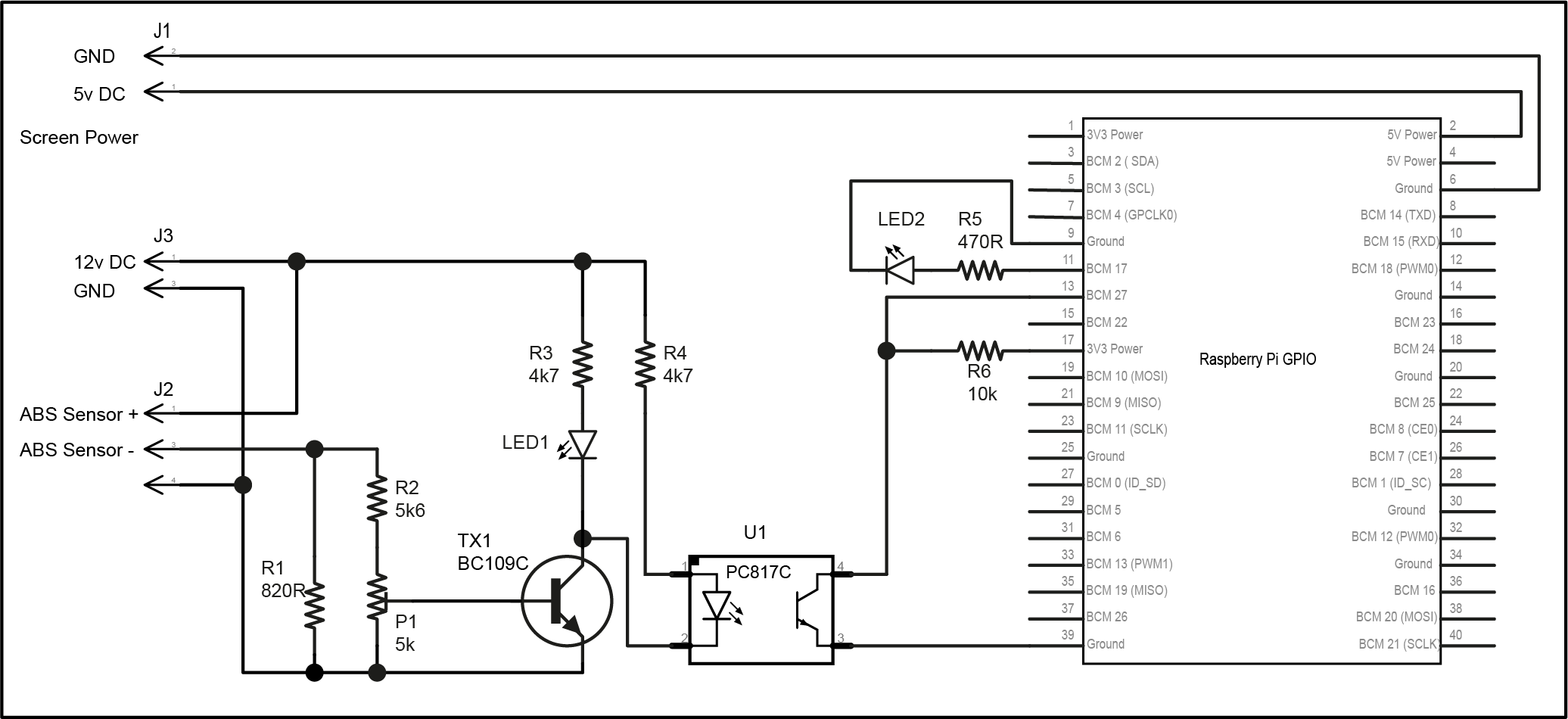


Figure : Schematic of RPM ABS adapter

Dd

# Web interface

Access to the controller is via a web page which shows the status of the invertor and allows a user to set the rotational speed of the motor and stop the motor.

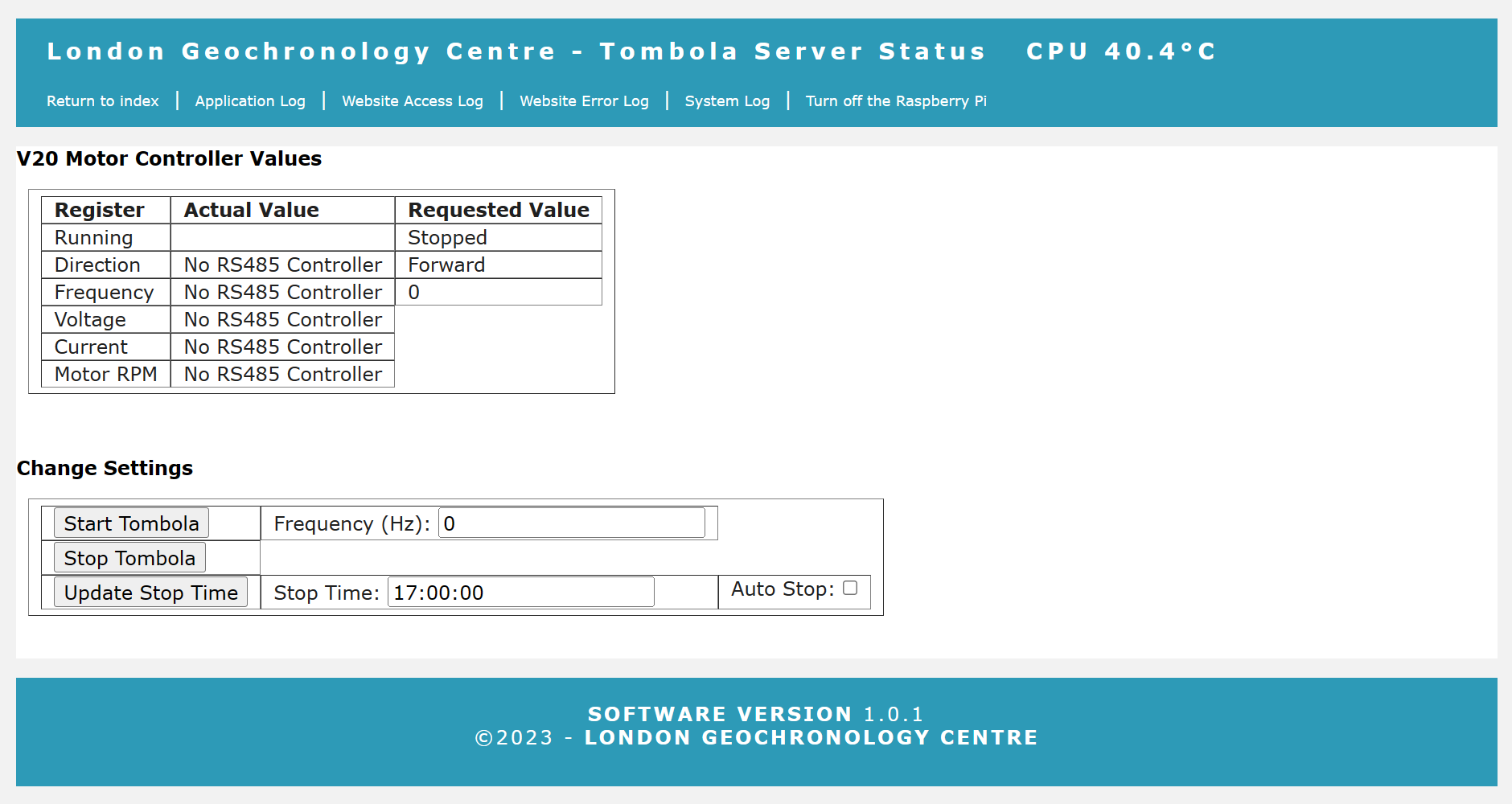


Figure : Web status page

The page has a “Start Tombola” button that required the frequency to be entered. It will then send a forward(frequency) command to the V20

The “Stop Tombola” button will stop the Tombola Motor.

The “Update Stop Time” button will allow you to change the stop time, the check box sets if the autostop is enabled. If auto stop is enabled and you the restart the Tombola it will immediately auto stop!

# Interfaces

## RS485 Specification

Need to add something about RS485 and Modbus

Uses the mimimalmodbus library to interface python

# Operating System Installation

## Operating system

Use BalenaEtcher to install the latest version of the bookworm-lite operating system onto a 32Gb MicroMMC card.

Connect via a USB keyboard and monitor and boot up the Raspberry Pi

Set the new username to tompi

Set a secure password

Run the sudo raspi-config command to:

enable ssh

disable Serial

disable 12C bus

change the GPU memory to 16Gb

set the hostname to byron-corelab

# Software installation

## Run Operating System Updates

Run sudo apt update

Run sudo apt upgrade

## Install PIP3 for Python 3.x installation

Run sudo apt install python3-pip

## Install Flask Libraries

Run sudo pip install flask --break-system-packages

## Install Minimal Modbus Package

Run sudo pip install minimalmodbus --break-system-packages

## Download Tombola Controller Application

Download from <https://github.com/westerlymerlin/UCL-tombola.git>

## Copy the python files to the Pi

Copy the files from GitHub to /home/tompi/

Copy the folder templates to /home/tompi/templates

Copy the folder static to /home/tompi/static

Create a folder /home/pi/logs

Copy the files /raspberry-pi/home/tompi to /home/tompi

Copy the file /raspberry-pi/home/tompi/etc/resolv.conf to /etc/resolv.conf

## Nginx installation

Run sudo apt install nginx

Copy the Github file

\raspberry-pi\home\pi\etc\nginx\nginx.conf

to

/etc/nginx/nginx.conf

Copy the GitHub file

\raspberry-pi\home\tompi\etc\nginx\sites-available\tombola

to

/etc/nginx/sites-available/tombola

Change directory to /etc/nginx/sites-enabled/

Run sudo rm default

Run sudo ln -s /etc/nginx/sites-available/tombola

## Gunicorn for Python 3.x installation

Run sudo apt install gunicorn3

Copy the GitHub file

\raspberry-pi\home\tompi\etc\systemd\system\gunicorn.service

to

/etc/systemd/system/gunicorn.service

Run sudo systemctl enable gunicorn

Run sudo systemctl start gunicorn

If flask is installed, the python files are in the /home/tompi directory, gunicorn3 is installed and configured and nginx is installed and configured the web service should be running and the site will be accessible on http://*ip address of the server*

References

Python Software Foundation (2020) *Python 3 Programming Language*. Online. Available online: <https://www.python.org> [Accessed July 2020].

Raspberry PI Foundation (2020) *Raspberry PI Model 4B Reference*. Available online: <https://www.raspberrypi.org/products/raspberry-pi-4-model-b/> [Accessed July 2020].