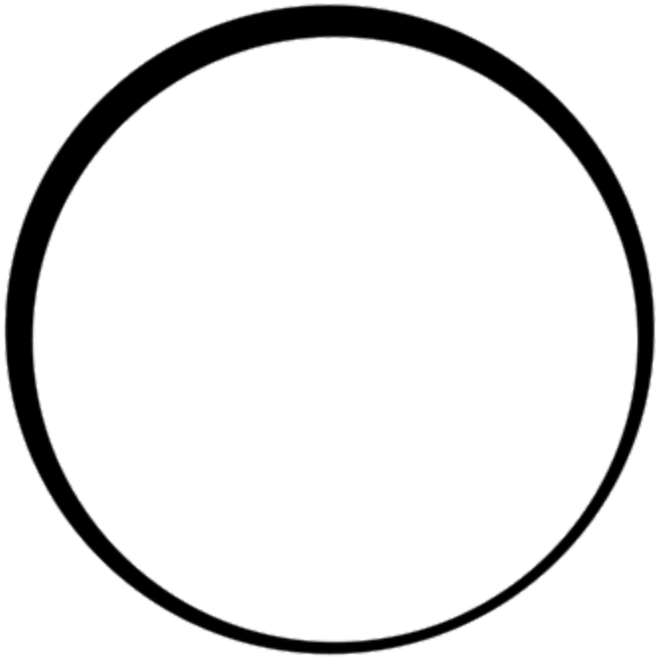

sonnen

Production FW GUI Documentation and Troubleshooting Guide

Revision Table

Changes	Authors	Version
[2021/04/2] Document Created	Heimerl, Justin	0.1
[2021/04/13] Output file feature added/documented.	Heimerl, Justin	0.2
[2021/04/22] Added PWM information	Heimerl, Justin Lin, Alex	0.3

Troubleshooting

Issue: Program crashes when opening COM port

Fix: If the COM port selected in the top box is correct, but the program is still crashing, find the COM port manually by following the steps here. When this is done manually input the COM port into the programming by opening app.py in a text editor. In the set_com_port function go ahead and set the port variable to the string “YOUR_COM_PORT” where YOUR_COM_PORT is the port found in the hardware manager. Other possibilities to find the COM port if this does not work are to use common serial programs such as PUTTY or TeraTerm, and look at what they detect.

Issue: Program crashes after attempting to read/write data over successful COM connection

Fix: Ensure the serial cable is plugged in correctly, and the board is powered. A serial connection will successfully estable with power provided, but data with not be received due to the mismatched serial pins. If the problem persists try again with a different board.

Issue: Program crashes after running script

Fix: Ensure all script arguments are valid. Valid script arguments can be found at the bottom of this document.

Issue: Setting a GPIO and reading it immediately after setting does not provide the expected value

Fix: Confirm the GPIO you are attempting to read is configured as an output, otherwise it will not be readable. Same logic holds true for GPIO’s that are not configured as inputs.

Scripting

The Production GUI gives the user the ability to script inputs to the firmware. This is done by pressing the select script button, and selecting a text file. Commands should be entered in the form: “ Command argument ”, for example a wait command (in seconds), followed by setting a GPIO looks like:

wait 1
set_gpio K7_Relay 1

The valid arguments and their results are listed in the table below.

Command (argument)	Result	Example
wait (argument in seconds)	Program sleeps for the amount of time in seconds.	wait 1
read_gpio (argument in human readable form)	Program returns the GPIO state over the serial connection.	read_gpio Fault_Sig4
set_gpio (argument in human readable form) (State)	Program sets the GPIO.	set_gpio K7_Relay 1
read_adc (argument in human readable form)	Program returns the ADC value.	read_adc I_DC_Leg1
set_pwm_control (argument in human readable form)	Turns PWM on or off.	set _pwm_control PWM1
set_pwm_freq (argument in human readable form)	Sets the frequency of the PWM.	set_pwm_freq 10000
set_pwm_dutycycle (argument in human readable form)	Sets the duty cycle of the PWM.	set_pwm_dutycycle 20

Configuring Output Files

The software allows the user to output all readings to a text file. The text in the file takes the form of:

DD/MM/YYYY HH:MM:SS w/r SIGNAL RETURN

D - Day
M - Month
Y - Year
H - Hour
M - Minute
S - Second
w/r - Write/Read
SIGNAL - Net being interacted with
RETURN - Value returned from the firmware

To begin using this feature first type your desired file path into the appropriate box in the GUI (labeled Output File Path). Click Set File Path. Ensure you receive the response “Output file path set to: __YOUR PATH__”. Now you may both interact with the GUI, and the responses will be sent to the output file, and pass in scripts, the output of which will also be sent to the output file.

IMPORTANT: WHEN THE SCRIPT IS TERMINATED YOU MUST CLICK THE CLOSE FILE BUTTON.

The data is formatted so the responses (say from ADC readings) can be easily plotted over time in Excel, MATLAB, Python, etc. This can also be easily parsed with regular expressions.