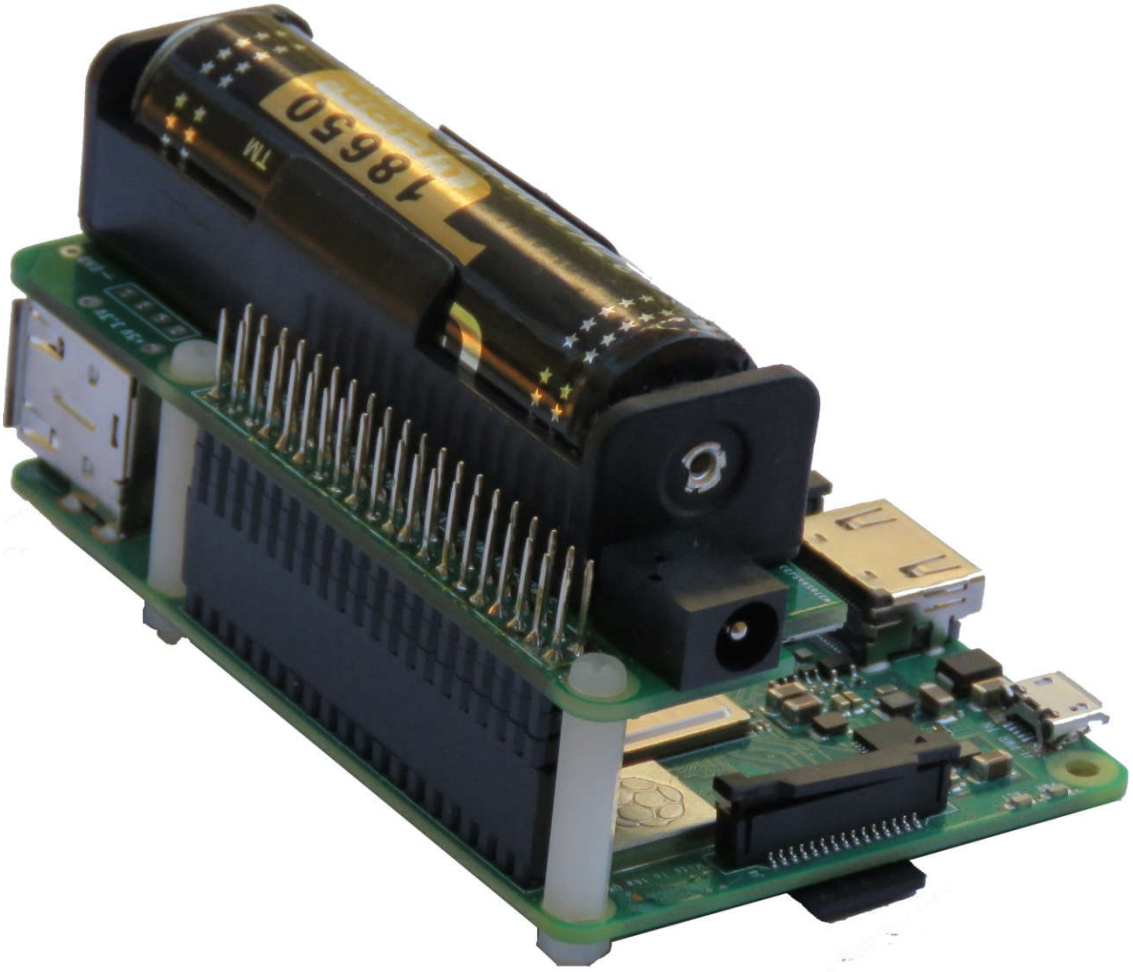
SUPER-WATCHDOG WITH BATTERY BACKUP FOR RASPBERRY PI

www.sequentmicrosystems.com

USER'S GUIDE VERSION 2.0

GENERAL DESCRIPTION



The Super-Watchdog for Raspberry Pi B+, 2, 3, 4 and Zero provides the safety required by mission critical projects, implementing two major functions. It power-cycle the Raspberry Pi in case of software lock-up, resetting not only the Pi but also any peripheral attached to it, and it prevents SD-Card failure in case of power loss, by allowing the Raspberry Pi to shut down itself safely. The 18650 Li-Ion battery can keep the Raspberry Pi running for hours, assuring continuous operation during power outages.

The watchdog has on board hardware to self check all the major functions. Self check is performed at each power up. The on-board LED is flashed rapidly (5 times per second) if a hardware error is detected, or slowly (1 time per second) to indicate normal functionality. The LED is turned on if the watchdog resets the Raspberry Pi, and off is the Raspberry Pi is powered down for a specific time interval.

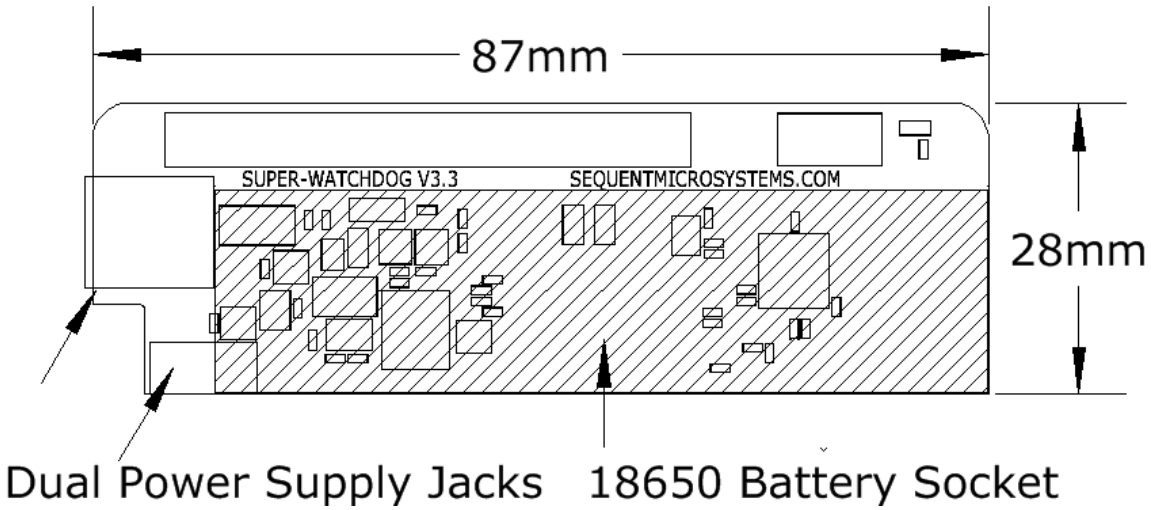
The card has a default timeout of 120 seconds. Once installed, if it does not receive a reset from Raspberry Pi within 2 minutes, it cuts the power and restores it after 10 seconds.

Raspberry Pi needs to issue a reset command before the timer on the watchdog expires. The command can be sent either on the I2C port , or by toggling GPIO11 (Pin 23 on the GPIO connector). The timer period after power up and the active timer period can be set from the command line. The number of resets is stored in flash and can be accessed or cleared from the command line.

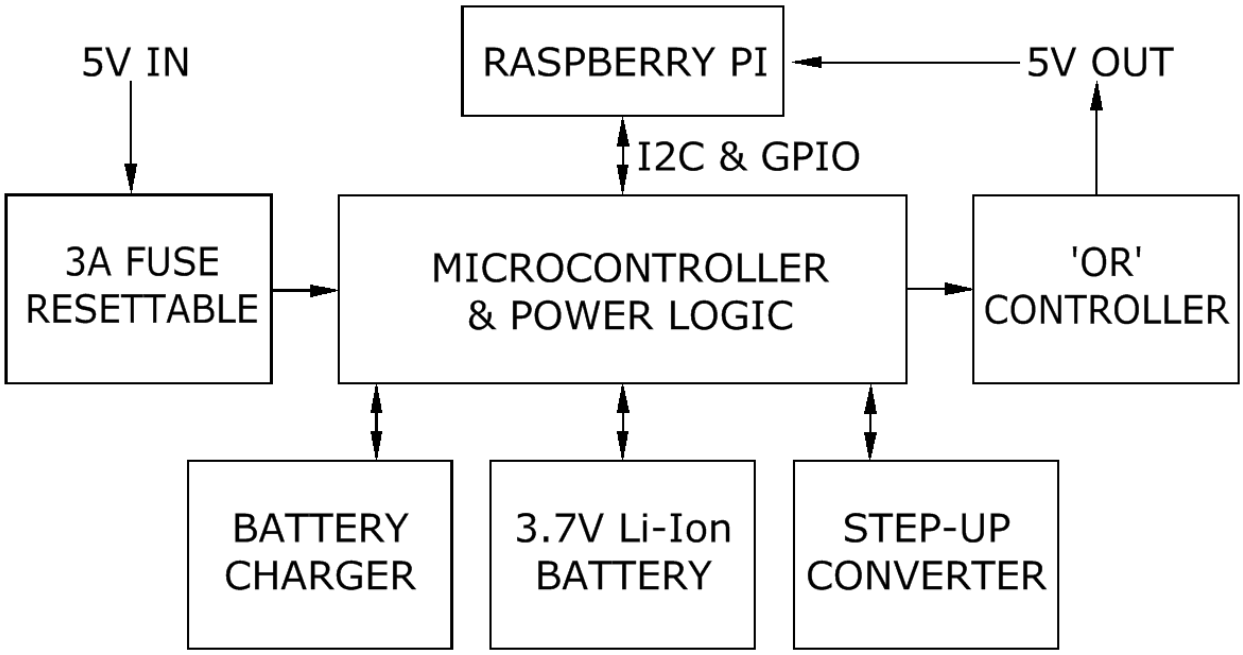
When running on battery power, the watchdog processor monitors the battery voltage. To prevent permanent damage to the battery due to over-discharging, the processor will cut off the power to Raspberry Pi when the battery drops below 2.8V. A software command is available to the user to also monitor the battery voltage. It is strongly recommended that Raspberry Pi performs a self-shutdown then the battery voltage drops to 3.0V.

Command line software and Python drivers can be downloaded from Github.

BOARD LAYOUT



BLOCK DIAGRAM



POWER REQUIREMENTS

In order for the watchdog to cycle the power to Raspberry Pi, +5V power needs to be supplied only to the watchdog. The watchdog has two power connectors: USB connector on bottom and 2.1mm barrel connector on top. The USB connector can supply up to 1.5A, while the barrel connector up to 2A. If your Raspberry Pi system requires more than 2A, you need to supply power to both connectors.

If the 5V power supply fails, the watchdog switches the power supply to battery. When running on battery, the current supplied to Raspberry Pi is limited to 1.5A. Actual current drawn from battery will be about twice the current supplied to Raspberry Pi.

SPECIFICATIONS

Watchdog current consumption:	10 mA @ +5V
Maximum switched power to Raspberry Pi:	3.5A
Battery charging current:	400mA
Maximum Raspberry Pi current supply:	3.5A
Raspberry Pi current supplied from battery:	2.0A (up to 10 seconds)
Raspberry Pi current until battery discharge:	1.5A

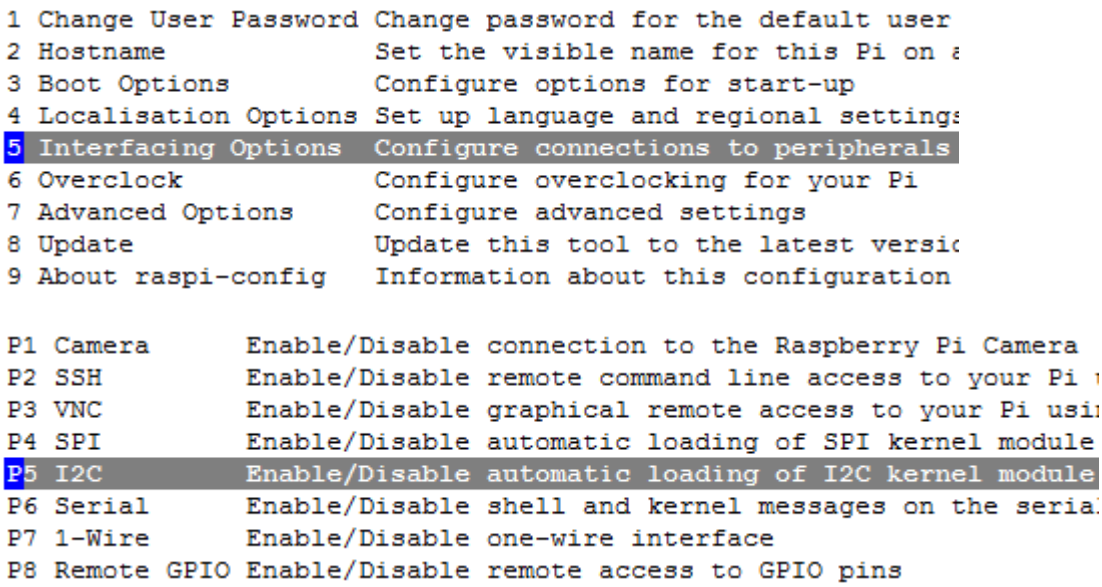
SOFTWARE SETUP

The Pi-DOG board occupies the I2C address 0x30.

1. Have your Raspberry Pi ready with the latest OS.

2. Enable I2C communication:

~$ sudo raspi-config



3. Install the wdt software from github.com:

~$ git clone https://github.com/SequentMicrosystems/wdt-rpi.git

~$ cd /home/pi/wdt-rpi

~/wdt-rpi$ sudo make install

~/wdt-rpi$ wdt

The program will respond with a list of available commands.

Type "wdt -h" for online help.

After installing the software, you can update it to the latest version with the commands:

~$ cd /home/pi/wdt-rpi

~/wdt-rpi$ git pull

~/wdt-rpi$ sudo make install

After installing the software, you can address the watchdog with the command "wdt". The watchdog will respond with a list of available commands. Any command send to the watchdog will reload the watchdog timer.

wdt -r[eload]: reload watchdog timer, prevent Raspberry power cycle.

wdt - d[efault]: set the default watchdog timeout (seconds). The watchdog loads this value after a power cycle. This value needs to be long enough for the boot process to complete.

wdt - p[eriod]: set the current watchdog period (seconds). Raspberry must address the watchdog faster than this value to prevent a power cycle. The period will be overwritten after a Raspberry power cycle with the default value.

wdt -c[clear]: clear the number of power cycles performed by the watchdog.

wdt -off[interval]: set the time the watchdog will keep the Raspberry power off (seconds). The default value is 10 seconds. After each power cycle, the off interval is reset to 10 seconds. Set this parameter to a large value to power down Raspberry Pi for a specific time interval.

wdt -g[et] d[efault]: get watchdog default parameter.

wdt -g[et] p[eriod]: get the watchdog period parameter.

wdt -g[et] r[esets]: get the number of power cycles performed by the watchdog.

wdt -g[et] v[in]: get the input voltage (mV).

wdt -g[et] off[interval]: get the current off interval (seconds).

wdt -g[et] vr[asp]: get the voltage applied to Raspberry Pi (mV).

wdt -g[et] vb[at]: get the battery voltage (mV).

wdt -g[et] c[harge]: read the battery charging status (0=off, 1=charged, 2=charging, 3=fault).

wdt -g[et] t[emp]: get the watchdog processor temperature.

PROGRAMING THE WATCHDOG

After the first power up, the watchdog cuts off the power to Raspberry Pi, performs a self-check and tries to turn on the power. The watchdog monitors the input power, the battery power and the power supplied to Raspberry Pi. If a hardware error is detected, the watchdog flashes rapidly (5 times per second) the on-board LED. If no error is detected, the LED flashes 1 time per second and the watchdog enters the stand-by state, waiting to be activated.

The watchdog is activated when is addressed first time on the I2C interface. The default timeout is set to 120 sec. If the timeout expires, the watchdog will turn off the Raspberry Pi power for 10 seconds, then turn it on again.

To disable the watchdog while setting it up, set the period to zero:

wdt -p 0

The watchdog can be reset either by writing a reset command from the command line ("wdt -r"), or by toggling pin 23 (GPIO 11) on the Raspberry Pi. The commands can be issued from a memory resident script or from cron. The script can have a resolution of 1 second, while cron can have a resolution of 1 minute.

1. RESET THE WATCHDOG FROM A SCRIPT USING THE COMMAND LINE

a. Use your favorite editor to create a script called wdt_reload with the following content:

```
#!/bin/sh
while :
do
   wdt -r
  sleep 60
 done
```

b. Make the script executable:

~/wdt-rpi$ chmod +x wdt_reload

c. Launch the script memory resident:

~/wdt-rpi$ wdt_reload&

d. Set the default watchdog timer period to your desired value. In this example we use 120 sec. The timer period needs to be longer than the "sleep" command in the wdt_reload file.

~/wdt-rpi$ wdt -p 120

2. RESET THE WATCHDOG FROM A SCRIPT USING THE GPIO INTERFACE

Same as above, with the following wdt_reload file content:

```
#!/bin/sh
```

```
gpio -g mode 11 out
while :
do
   gpio -g toggle 11
  sleep 60
 done
```

3. RESET THE WATCHDOG FROM CRON

Create the wdt_reload with the following content:

```
#!/bin/sh
/home/pi/wdt-rpi/wdt -r
```

Make it executable, then edit the crontab file:

```
~/wdt-rpi$ crontab -e
```

Add the following line at the end:

```
* * * * * /home/pi/wdt-rpi/wdt_reload
```

Save the crontab file. To check that the cron is running, add at the end of the wdt_reload the command

```
echo `date` >> /home/pi/wdt-rpi/logfile
```

After 1 minute you should see the logfile with the timestamp when it was called by cron:

```
~/wdt-rpi$ cat logfile
Fri Jun 28 18:04:02 PDT 2525
```

Remove the timestamp from the wdt_reload file. Set the watchdog period to anything shorter than 60 seconds, and wait for the watchdog to cycle the power on Raspberry Pi. Ten seconds before reboot the watchdog will accelerate flashing it's LED. During the 10 seconds reboot, the LED will stay on.

4. SCHEDULED POWER DOWN

To schedule a Raspberry Pi power down for a specific time T(seconds), do the following:

a. Set the watchdog off interval:

```
wdt -off T
```

b. Set the watchdog period long enough for Raspberry Pi to shut down (10 sec recommended).

```
wdt -p 10
```

c. Shutdown Raspberry Pi:

```
sudo shutdown now
```