MaxAir Technical – Basic Fault Finding

MaxAir relies on a number of services in order function and is built on a LAMP (**L**inux, **A**pache, **M**ySQL, **P**HP) platform. The setup process is performed by executing **php /var/www/setup.php**, which will check that the platform requirements are met before installing the software and configuring the initial database.

Services

pihome_jobs_schedule

This service manages the running of a periodic tasks, to check it is functioning from the command prompt execute:

systemctl status pihome_jobs_schedule.service

The expected response should be similar to the following:

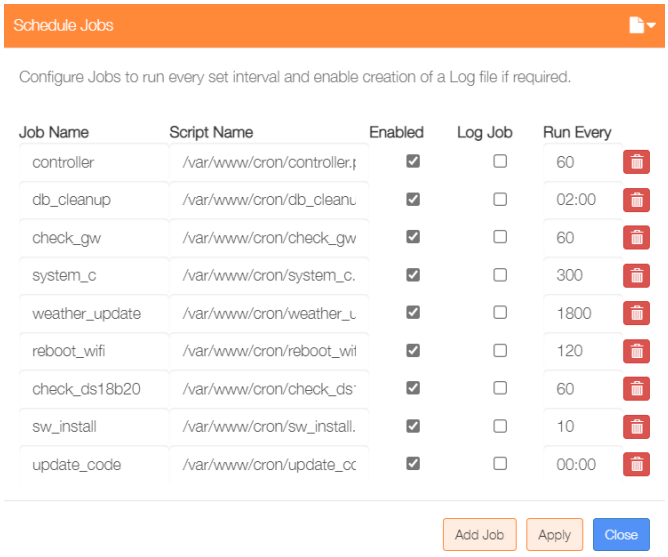
```
● pihome_jobs_schedule.service - Schedule
   Loaded: loaded (/lib/systemd/system/pihome_jobs_schedule.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2021-08-31 21:55:58 BST; 15h ago
 Main PID: 3175 (python3)
    Tasks: 3 (limit: 1126)
   Memory: 85.2M
   CGroup: /system.slice/pihome_jobs_schedule.service
           ├─3175 /usr/bin/python3 /var/www/cron/jobs_schedule.py
           ├─3508 python3 /var/www/cron/gpio_ds18b20.py
           └─8237 python3 /var/www/cron/gateway.py
```

The screen above shows that the service is running and controlling the scheduling of the 'jobs' job_schedule.py, gpio_ds18b20.py and gateway.py.

If the service is not running, the MaxAir setup process has failed in some way. A complete re-install is recommended.

Jobs

The scheduler service executes a number of jobs on a periodic basis, these jobs will be installed as part of the setup process.



The installed jobs can be views from the 'Jobs' menu item in the 'Settings/System Configuration menu.

jobs_schedule

This job manages the running of the other periodic tasks, to check it is functioning from the command prompt execute:

ps -ax | grep jobs_schedule.py | grep -v grep

The expected response should be similar to the following:

3175 ? Ss 0:47 /usr/bin/python3 /var/www/cron/jobs_schedule.py

If nothing is returned, then this indicates that the scheduler function is not operating

gateway

This job manages collection of sensor data and the setting of relay states, to check it is functioning from the command prompt execute:

ps -ax | grep gateway.py | grep -v grep

The expected response should be similar to the following:

8237 ? S 0:09 python3 /var/www/cron/gateway.py

If nothing is returned indicating that the gateway is not operational, try starting it manually using the following command, which may give some clues as to why it has not started:

python3 /var/www/cron/gateway.py

ds18b20

This job manages collection of 1-wire sensor data, to check it is functioning from the command prompt execute:

ps -ax | grep gpio_ds18b20.py | grep -v grep

The expected response should be similar to the following:

3508 ? S 0:00 python3 /var/www/cron/gpio_ds18b20.py

If nothing is returned indicating that the 1-wire sensors are not operational, try starting it manually using the following command, which may give some clues as to why it has not started:

python3 /var/www/cron/ gpio_ds18b20.py

controller

This job is the main engine for MaxAir, it controls the zones by reading and writing to message queues shared with the gateway job and by reacting to 'state' information provided by operating mode and schedule settings.

The jobs schedular is configured to execute controller once ever 60 seconds, it can be executed manually using the following command:

php /var/www/cron/ controller.php

The response shown below is for a 2 zone system, operating in 'Boiler Mode', with no schedule active.

Additional debugging information is available by executing with an additional argument eg:

php /var/www/cron/ controller.php 1

The actual argument is unimportant, any single character will work.

```
         __  __               _    _
        |  \/  |             /\   (_)
        | \  / | __ _ __  __ /  \   _ _ __
        | |\/| |/ _` |\ \/ // /\ \ | | '__|
        | |  | | (_| | >  </ ____ \| | |
        |_|  |_|\__,_|/_/\_\/    \_\_|_|

                 S M A R T   T H E R M O S T A T

*******************************************************************
*   System Controller Script Version 0.01 Build Date 19/10/2020  *
*   Update on 18/08/2021                                          *
*                                     Have Fun - PiHome.eu        *
*******************************************************************

2021-09-01 13:48:28 - Controller Script Started
2021-09-01 13:48:28 - Operating in Boiler Mode
2021-09-01 13:48:28 - Day of the Week: 3
------------------------------------------------------------------------------------------------------
2021-09-01 13:48:28 - Zone: Sensor Reading      18.30
2021-09-01 13:48:28 - Zone: Weather Factor      0.6
2021-09-01 13:48:28 - Zone: DeadBand            0.5
2021-09-01 13:48:28 - Zone: Cut In Temperature         0
2021-09-01 13:48:28 - Zone: Cut Out Temperature        0
2021-09-01 13:48:28 - Zone: Mode        0
2021-09-01 13:48:28 - Zone ID: 38
2021-09-01 13:48:28 - Zone: Central Heating Controller: 0 Controller Child: 13 Zone Status: 0
2021-09-01 13:48:28 - Zone: Central Heating Stop Cause: No Schedule - Target C:0 Zone C:18.30
------------------------------------------------------------------------------------------------------
2021-09-01 13:48:28 - Zone: Sensor Reading      27.80
2021-09-01 13:48:28 - Zone: Weather Factor      0.6
2021-09-01 13:48:28 - Zone: DeadBand            0.5
2021-09-01 13:48:28 - Zone: Cut In Temperature         0
2021-09-01 13:48:28 - Zone: Cut Out Temperature        0
2021-09-01 13:48:28 - Zone: Mode        0
2021-09-01 13:48:28 - Zone ID: 39
2021-09-01 13:48:28 - Zone: Hot Water Controller: 0 Controller Child: 11 Zone Status: 0
2021-09-01 13:48:28 - Zone: Hot Water Stop Cause: No Schedule - Target C:0 Zone C:27.80
------------------------------------------------------------------------------------------------------
2021-09-01 13:48:28 - System Controller Active Status: 0
2021-09-01 13:48:28 - System Controller Hysteresis Status: 0
------------------------------------------------------------------------------------------------------
2021-09-01 13:48:28 - Purging marked records.
------------------------------------------------------------------------------------------------------
2021-09-01 13:48:28 - Controller Script Ended
******************************************************************************************************
```

Checking GPIO Relay Operation

Relays are controlled by the gateway job, which reserves any configured relays and hence they are not available for any other use.

A test program is available, which temporarily stops the gateway job to enable setting relays either ON or OFF. For example if a relay is connected to pin 13 of the GPIO header and is switched ON by applying a HIGH level trigger, then to switch the relay ON, from the command line execute:

python3 /var/www/cron/ gpio_relay_test.py 13 1

to switch the relay OFF, from the command line execute:

python3 /var/www/cron/ gpio_relay_test.py 13 0

If the relay is switched ON by applying a LOW level trigger, then to switch the relay ON, from the command line execute:

python3 /var/www/cron/ gpio_relay_test.py 13 0

to switch the relay OFF, from the command line execute:

python3 /var/www/cron/ gpio_relay_test.py 13 1