This document describes the installation procedure of Husdata's MQTT heatpump gateway for Home Assistant, home automations system.

The instruction is based on the following assumptions:

- StatLink and the H1 interface is installed and run on the same Raspberry Pi as where Home Assistant is installed. These can run on separate devices but then the Broker IP setting in mqtt.py has then to be changed.

- Home Assistant is based on the Hassbian image.

- The built in HA mqtt broker is to be used. So that means that you should not install the Mosqitto broker.

Screenshot from Home Assistant connected to and IVT Greenline Rego 600

H1 interface basics

The H1 interface from Husdata.se is a universal platform that is compatible with most popular heat pumps form Bosch, IVT, Thermia and NIBE. The communication is serial using a clear text protocol described in "H1 Interface development manual" downloadable from Husdata.se.

The interface is continuously monitoring changes of sensors and devices on the heatpump. As soon as for example temperature is changed the new value will be sent over the serial line.
That data string could loo like this: XR0001002A
XR is an instruction telling that data has been updated
0001 means "Radiator Return temp"
002A means 2A hex = 42 divided by 10 = 4,2 degrees

Sample output
```
XR8105FA59    Degree min/integral (-144.7d)
XR000100F2    Radiator Return (24.2c)
XR000200A4    Radiator Forward (16.4c)
XR0003018B    Heat carrier Return (39.5c)
XR00040114    Heat carrier Forwrd (27.6c)
```

So this means the some pump models will have the 0001 sensor and others are lacking this sensor. You have to add / remove sensors in the HA configuration.yaml to adapt to your specific heat pump. But the Sensor ID:s are always the same for all heat pumps. For example 0001 is always Rad Return and 0003 is always Heat carrier Returns despite model.

Installation procedure

1. Install interface
   a. Plug in the H1 interface to the GPIO connector to the outmost row.
   b. Connect to Heatpump according to instructions supplied.

2. Install Hasbian
   a. Download the Hasbian image and load into your SD card
   b. Install Home Assistant according to Home Assistant instructions.
   c. Append to Configuration.yaml
   ```
   mqtt:
      username: homeassistant
      password: password
      keepalive: 60
   ```

   d. Append Sensors as described last in this document to the "Configuration.yaml" file.
      Note that these definitions has to be adjusted depending on heat pump model.

3. Copy program files to Raspberry
   a. `mkdir husdata` (optional. You can place the files on directory of choice)
   b. Using suitable SFTP client, copy supplied files to the "py/husdata" directory
   ```
   statlink.py
   s_io.py
   mqtt.py
   get-pip.py
   ```

4. Install PAHO Mqtt client
   a. Run: `sudo python get-pip.py`
   b. Run: `sudo pip install paho-mqtt`

5. Install Python serial library
   a. Run: `sudo apt-get install python-serial`

6. Setup Serial GPIO port
   a. Run: `sudo nano /boot/cmdline.txt`
      Remove text: *console=serial0,115200*
   b. Run: `sudo nano /boot/config.txt`
      Add following rows at the end: enable_uart=1
      In the case of Rapsberry Pi 3 this line has to be added:
      `dtoverlay=pi3-disable-bt`

   Save and exit, Reboot the Pi after update!

Functionality test

To run StatLink, use the following command: `python statlink.py`

Optional step to make StatLink start up at boot.

Run: `sudo nano /etc/rc.local`
Add before last "exit 0" row
```
cd /home/pi
python statlink.py
```

Troubleshooting

- Testing the communication with interface and heatpump
    - Install serial terminal emulator:
      ```
      sudo apt-get install minicom
      ```

    - Start Minicom terminal emulator
      ```
      minicom -D /dev/ttyAMA0 -b 19200
      ```
      (To quit minicom, Ctrl-A Q.)

    - Test communication with interface and heatpump
      Some commands you can send to interface.:
        - XV – Returning H1 forware version
        - XP – Turn onn Clear Text mode showing objects and data in clear text.
        - XR – Refresh read all data from heatpump
    - For additional information and command for the interface, download and view the "H1 Interface development manual"

CONFIGURATON.YAML

Depending on type, brand or model of heat pump you are having the sensor definitions has to be adjusted to match the specific heat pump.

So what you need to do is to add, remove and change the sensor definitions to match the correct Sensor ID:s (0001 in this case) with the correct description in HA, "Rad Return"
The available and usable Sensor ID:s for different heatpumps can be found in the "H1 Development manual" downloadable on Husdata.se

```
- platform: mqtt
  state_topic: "heatpump/0001"
  name: "Rad Return"
  unit_of_measurement: "C"
```

Values to be adjusted according to heat pump model highlighted.

Samle config for Rego 600.

```
sensor:
  - platform: mqtt
    state_topic: "heatpump/0001"
    name: "Rad Return"
    unit_of_measurement: "C"

  - platform: mqtt
    state_topic: "heatpump/0002"
    name: "Rad Forward"
    unit_of_measurement: "C"

  - platform: mqtt
    state_topic: "heatpump/0003"
    name: "Heat Carrier Return"
    unit_of_measurement: "C"

  - platform: mqtt
    state_topic: "heatpump/0004"
    name: "Heat Carrier Forward"
    unit_of_measurement: "C"

  - platform: mqtt
    state_topic: "heatpump/0005"
    name: "Brine In"
    unit_of_measurement: "C"

  - platform: mqtt
    state_topic: "heatpump/0006"
    name: "Brine Out"
    unit_of_measurement: "C"

  - platform: mqtt
    state_topic: "heatpump/0007"
    name: "Outdoor"
    unit_of_measurement: "C"

  - platform: mqtt
    state_topic: "heatpump/0008"
    name: "Indoor"
    unit_of_measurement: "C"

  - platform: mqtt
    state_topic: "heatpump/0009"
    name: "Warm water"
    unit_of_measurement: "C"

  - platform: mqtt
    state_topic: "heatpump/000B"
    name: "Hot gas"
```

```
      unit_of_measurement: "C"

  - platform: mqtt
    state_topic: "heatpump/3104"
    name: "Electric heater"
    unit_of_measurement: "kW"


binary_sensor:
  - platform: mqtt
    state_topic: "heatpump/1A01"
    name: "Compressor"
    payload_on: "1"
    payload_off: "0"

  - platform: mqtt
    state_topic: "heatpump/1A07"
    name: "Switch valve"
    payload_on: "1"
    payload_off: "0"

  - platform: mqtt
    state_topic: "heatpump/1A06"
    name: "Radiator pump"
    payload_on: "1"
    payload_off: "0"

  - platform: mqtt
    state_topic: "heatpump/1A04"
    name: "Brine pump"
    payload_on: "1"
    payload_off: "0"

  - platform: mqtt
    state_topic: "heatpump/1A05"
    name: "Heat carrier pump"
    payload_on: "1"
    payload_off: "0"

  - platform: mqtt
    state_topic: "heatpump/1A20"
    name: "Alarm"
    payload_on: "1"
    payload_off: "0"
    device_class: power
```