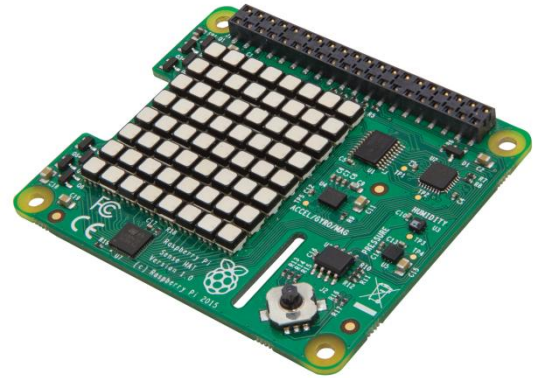


Raspberry Pi Sense HAT

Fact Sheet

Description

The Raspberry Pi Sense HAT is attached on top of the Raspberry Pi via the 40 GPIO pins (which provide the data and power interface) to create an 'Astro Pi'. The Sense HAT has several integrated circuit based sensors that you can use for many different types of experiments, applications, and even games.



Technical Specification

- Gyroscope – angular rate sensor: $\pm 245/500/2000$ dps
- Accelerometer - Linear acceleration sensor: $\pm 2/4/8/16$ g
- Magnetometer - Magnetic Sensor: $\pm 4/8/12/16$ gauss
- Barometer: 260 – 1260 hPa absolute range (accuracy depends on the temperature and pressure, ± 0.1 hPa under normal conditions)
- Temperature sensor (Temperature accurate to ± 2 °C in the 0-65 °C range)
- Relative Humidity sensor (accurate to $\pm 4.5\%$ in the 20-80%rH range, accurate to ± 0.5 °C in 15-40 °C range)
- 8x8 LED matrix display
- Small 5 button joystick

SENSE HAT

INSTALLATION

In order to work correctly, the Sense HAT requires an up-to-date kernel, I2C to be enabled, and a few libraries to get started.

1. Ensure your APT package list is up-to-date:

```
sudo apt-get update
```

2. Next, install the sense-hat package which will ensure the kernel is up-to-date, enable I2C, and install the necessary libraries and programs:

```
sudo apt-get install sense-hat
```

3. Finally, a reboot may be required if I2C was disabled or the kernel was not up-to-date prior to the install:

```
sudo reboot
```

HARDWARE

The schematics can be found [here](#).

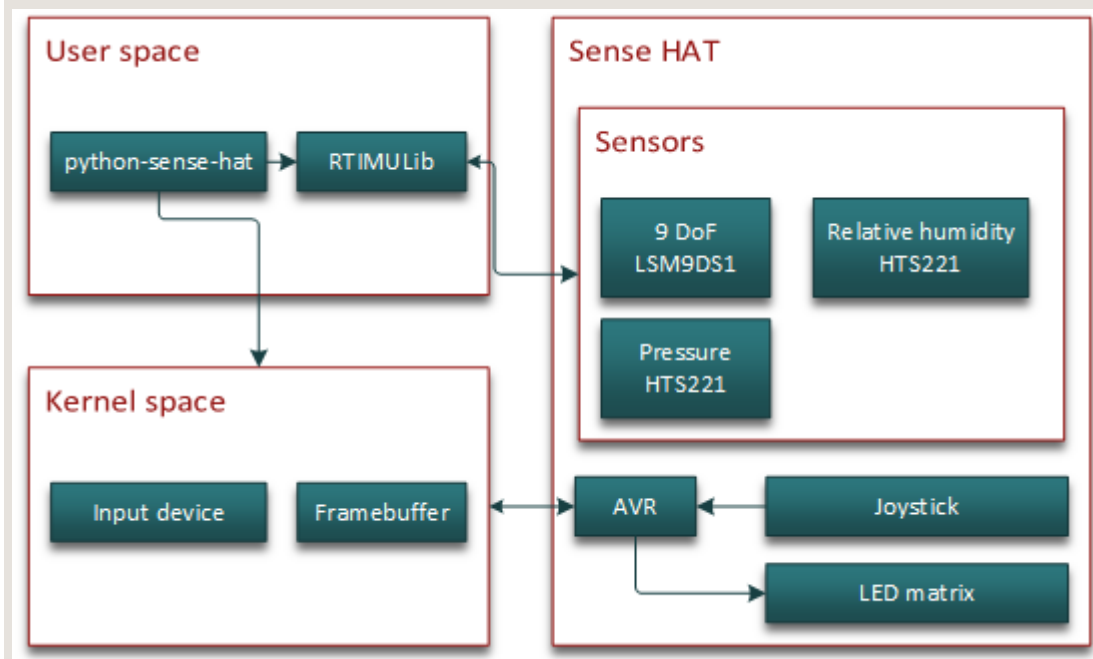
SOFTWARE OVERVIEW

After installation, example code can be found under `/usr/src/sense-hat/examples`.

These can be copied to the user's home directory by running `cp /usr/src/sense-hat/examples ~/ -a`.

The C/C++ examples can be compiled by running `make` in the appropriate directory.

The `RTIMULibDrive11` example comes pre-compiled to help ensure everything works as intended. It can be launched by running `RTIMULibDrive11` and closed by pressing `Ctrl+c`.



PYTHON SENSE-HAT

`sense-hat` is the officially supported library for the Sense HAT; it provides access to all of the on-board sensors and the LED matrix.

Complete documentation can be found at pythonhosted.org/sense-hat.

RTIMULIB

[RTIMULib](#) is a C++ and Python library that makes it easy to use 9-dof and 10-dof IMUs with embedded Linux systems. A pre-calibrated settings file is provided in `/etc/RTIMULib.ini`, which is also copied and used by `sense-hat`. The included examples look

for `RTIMULib.ini` in the current working directory, so you may wish to copy the file there to get more accurate data.

OTHER

LED MATRIX

The LED matrix is an RGB565 [framebuffer](#) with the id "RPi-Sense FB". The appropriate device node can be written to as a standard file or mmap-ed. The included 'snake' example shows how to access the framebuffer.

JOYSTICK

The joystick comes up as an input event device named "Raspberry Pi Sense HAT Joystick", mapped to the arrow keys and `Enter`. It should be supported by any library which is capable of handling inputs, or directly through the [evdev interface](#). Suitable libraries include SDL, [pygame](#) and [python-evdev](#). The included 'snake' example shows how to access the joystick directly.

CALIBRATION

Taken from this [forum post](#).

Install the necessary software and run the calibration program as follows:

```
sudo apt-get update
sudo apt-get install octave -y
cd
cp /usr/share/librtimulib-utils/RTellipsoidFit ./ -a
cd RTellipsoidFit
RTIMULibCal
```

You will then see this menu:

Options are:

```
m - calibrate magnetometer with min/max
e - calibrate magnetometer with ellipsoid (do min/max first)
a - calibrate accelerometers
x - exit
```

Enter option:

Press lowercase **m**. The following message will then show; press any key to start.

```
Magnetometer min/max calibration
-----

Waggle the IMU chip around, ensuring that all six axes
(+x, -x, +y, -y and +z, -z) go through their extrema.
When all extrema have been achieved, enter 's' to save, 'r'
to reset
or 'x' to abort and discard the data.

Press any key to start...
```

After it starts, you will see something similar to this scrolling up the screen:

```
Min x:  51.60  min y:  69.39  min z:  65.91
Max x:  53.15  max y:  70.97  max z:  67.97
```

Focus on the two lines at the very bottom of the screen, as these are the most recently posted measurements from the program. Now you have to move the Astro Pi around in every possible way you can think of. It helps if you unplug all non-essential cables to avoid clutter.

Try and get a complete circle in each of the pitch, roll and yaw axes. Take care not to accidentally eject the SD card while doing this.

Spend a few minutes moving the Astro Pi, and stop when you find that the numbers are not changing anymore.

Now press lowercase `s` then lowercase `x` to exit the program. If you run the `ls` command now, you'll see a new `RTIMULib.ini` file has been created.

In addition to those steps, you can also do the ellipsoid fit by performing the steps above, but pressing `e` instead of `m`.

When you're done, copy the resulting `RTIMULib.ini` to `/etc/` and remove the local copy in `~/.config/sense_hat/`:

```
rm ~/.config/sense_hat/RTIMULib.ini
sudo cp RTIMULib.ini /etc
```

You are now done.

UPDATING THE AVR FIRMWARE

...

EEPROM DATA

These steps may not work on Raspberry Pi 2 Model B Rev 1.0 and Raspberry Pi 3 Model B boards. The firmware will take control of I2C0, causing the ID pins to be configured as inputs.

1. Enable I2C0 and I2C1 by adding the following line to `/boot/config.txt`:

```
2. dtparam=i2c_vc=on
```

```
dtparam=i2c_arm=on
```

3. Enter the following command to reboot:

```
sudo systemctl reboot
```

4. Download and build the flash tool:

```
5. git clone https://github.com/raspberrypi/hats.git  
6. cd hats/epromutils
```

```
make
```

READING

1. EEPROM data can be read with the following command:

```
sudo ./eepflash.sh -f=sense_read.eep -t=24c32 -r
```

WRITING

Please note that this operation is potentially dangerous, and is not needed for the everyday user. The steps below are provided for debugging purposes only. If an error occurs, the HAT may no longer be automatically detected.

1. Download EEPROM settings and build the .eep binary:

```
2. wget  
https://github.com/raspberrypi/rpi-sense/raw/master/eprom/eprom_settings.txt -O sense_eprom.txt
```

```
./eepmake sense_eprom.txt sense.eep  
/boot/overlays/rpi-sense-overlay.dtb
```

3. Disable write protection:

```
i2cset -y -f 1 0x46 0xf3 1
```

4. Write the EEPROM data:

```
5. sudo ./eepflash.sh -f=sense.eep -t=24c32 -w
```

6. Re-enable write protection:

```
i2cset -y -f 1 0x46 0xf3 0
```