Using Whand on a Raspberry pi

By Alain R Marchand – nov 2020

This document is intended for Raspberry pi (RP) users who want a quick look at the Whand language under Linux. Whand allows you to control events with statements such as “when”, “until” or “since” instead of the usual sequential programming (e.g. under Python), so that programs are easy to read.

Here I describe how to implement Whand on your RP in a few simple steps. More practical details on the implementation can be found in Whand_User_Manual. For an introduction to the language, see Whand_for_beginners or Whand_step_by_step. For a full description of Whand, see Whand_Reference_Manual.

A very good tutorial by Christopher Barnatt on how to plug various devices onto the Raspberry pi can be found at:

https://www.youtube.com/watch?v=NAl-ULEattw

Once plugged, input devices that provide an on/off signal or output devices that require turning on or off can be easily controlled under Whand without the need to get into Python programming (which is not a bad idea anyway). So far, only one RP device may be controlled at the same time under Whand.

Whand is open source, written in Python. Help in improving and developing the language on various platforms is welcome.

Step 1: Check or install Python

Whand works well with Python 3 (especially Python 3.7). If it is not already installed on your RP, type in the following Linux commands:

```
sudo apt update
sudo apt install python3 idle3
```

Step 2: unpacking Whand

Whand is provided as a zip file containing all the necessary modules, the main program whand_V2_7.py, a config.txt file, as well as documentation (see details in Whand_User_Manual).

Unzip the file and copy all parts into a new directory of your RP, e.g. /home/pi/whand while respecting the directory structure, i.e.

```
whand_____
          |_____autotests
          |_____data
          |_____doc
          |_____install
          |_____license
          |_____scripts
          |_____whand_modules
```

Step 3: installing the driver

In directory /home/pi/whand/whand_modules:

- copy or rename whand_driver_raspy.py to whand_driver.py
- copy or rename whand_parameters_raspy.py to whand_parameters.py

Step4: creating a Whand script

In directory /home/pi/whand/scripts:

Edit or create a text file (e.g. new.txt) containing for instance:

```
# This is my first Whand program on Raspberry pi
button: pin(27)                 # define an input (BCM convention on RP: GPIO 2 to 27)
output(20): LED                 # define an output (BCM convention: GPIO 2 to 27)
LED when button                 # link LED to a button press
   until end button             # definition of LED (continued)
exit when 30s since start       # stop/exit condition
controlpanel                    # display a control panel
show button, LED
```

Note: you cannot use the idle editor via a distant interface, but other editors are available, e.g. mousepad.

Step 4: running the Whand script

The script may be run either directly on the RP, or via a distant interface (e.g. from a PC with Windows). The controlpanel and show instructions only work when the RP is controlled directly, i.e. with a keyboard and mouse connected to the RP. When using the distant interface, remove or comment-out with # the last two lines of the script.

The advantage of the controlpanel is that it allows you to visualize the functioning of the Whand script even when no physical input or output is connected. The input can furthermore be manually simulated by clicking on its box in the controlpanel.

To run the script, type in the following command lines:

```
cd whand
sudo python whand_V2_7.py
```

Then you get a message from Whand asking for a file name. Type new.txt if that is the name of your script. The program should then compile and ask you for a start command, either on the keyboard (Enter) or on the controlpanel (Start or Global Start). In any case, the above script will only run for 30s during which you may try manipulating the button and watching the LED.

After the script terminates under controlpanel, please click Global Stop before closing the panel.