# Instructions for Running the Current Stable Build of Aardvark

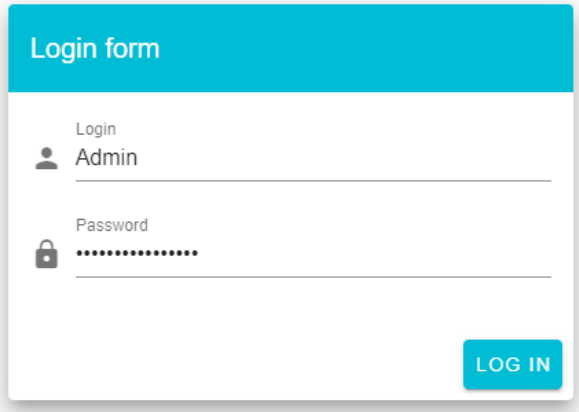
These slides are using the files from the stable Main branch of https://github.com/TriEmbed/edger

It is assumed that your wifi info has been entered, ant should is running, and aardvark has been started.

See http://edger.app for details on how to get to this point.

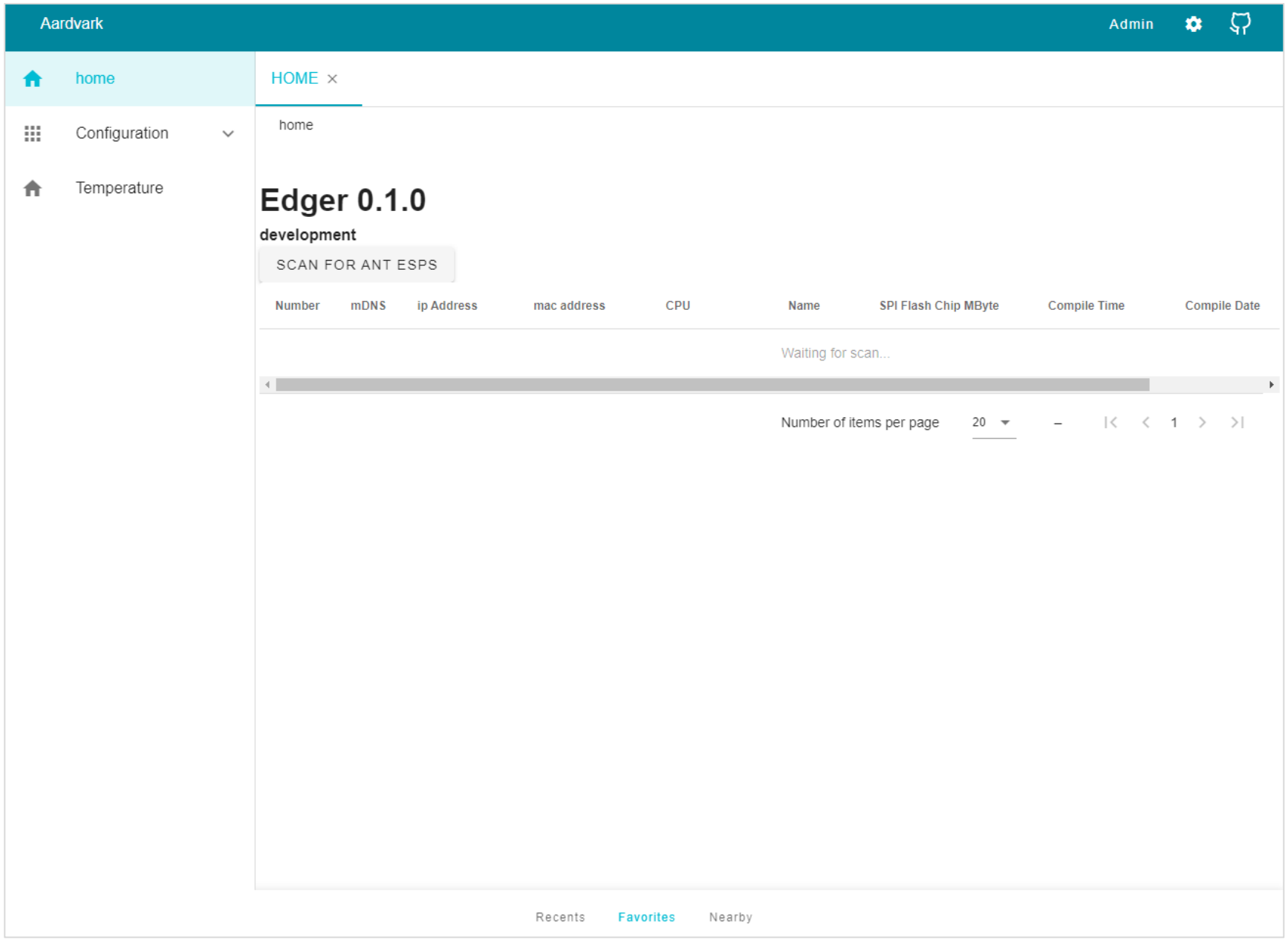
# Login

- Point your browser to http://localhost:8080 and this screen should appear
- This demo uses Chrome. Firefox also seems to work.
- Click LOG IN button



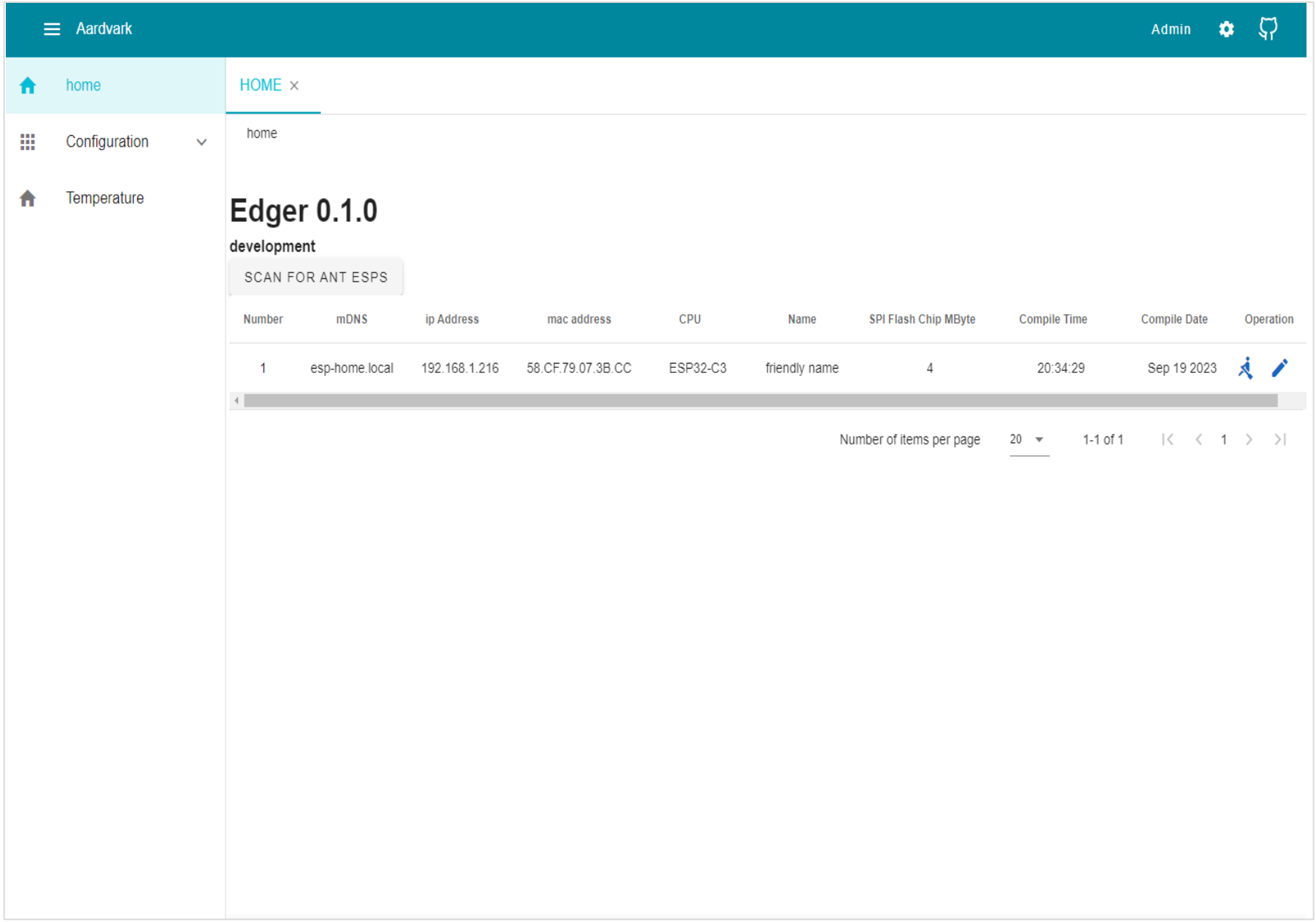
# Home

- After a successful login, this Home screen should appear
- Click the SCAN FOR ANT ESPS button to find available devices
- If at least one device doesn't appear after the initial scan, you may need to repeat the scan a few times



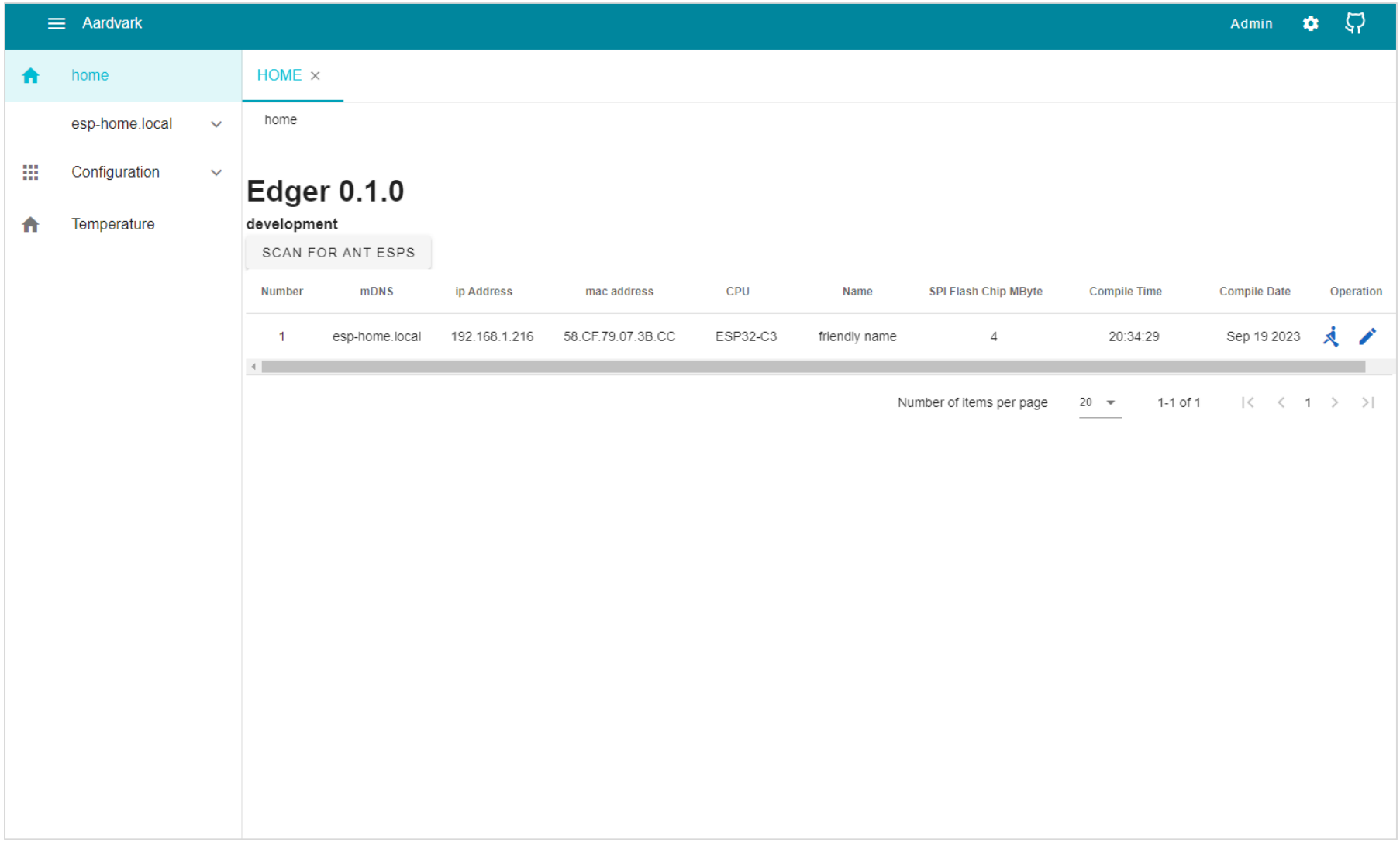
# Scan for Devices

- After a successful scan, the available devices will be listed
- Click on the ‘Witches Broom’ icon at the right side of the device row to connect to the device.



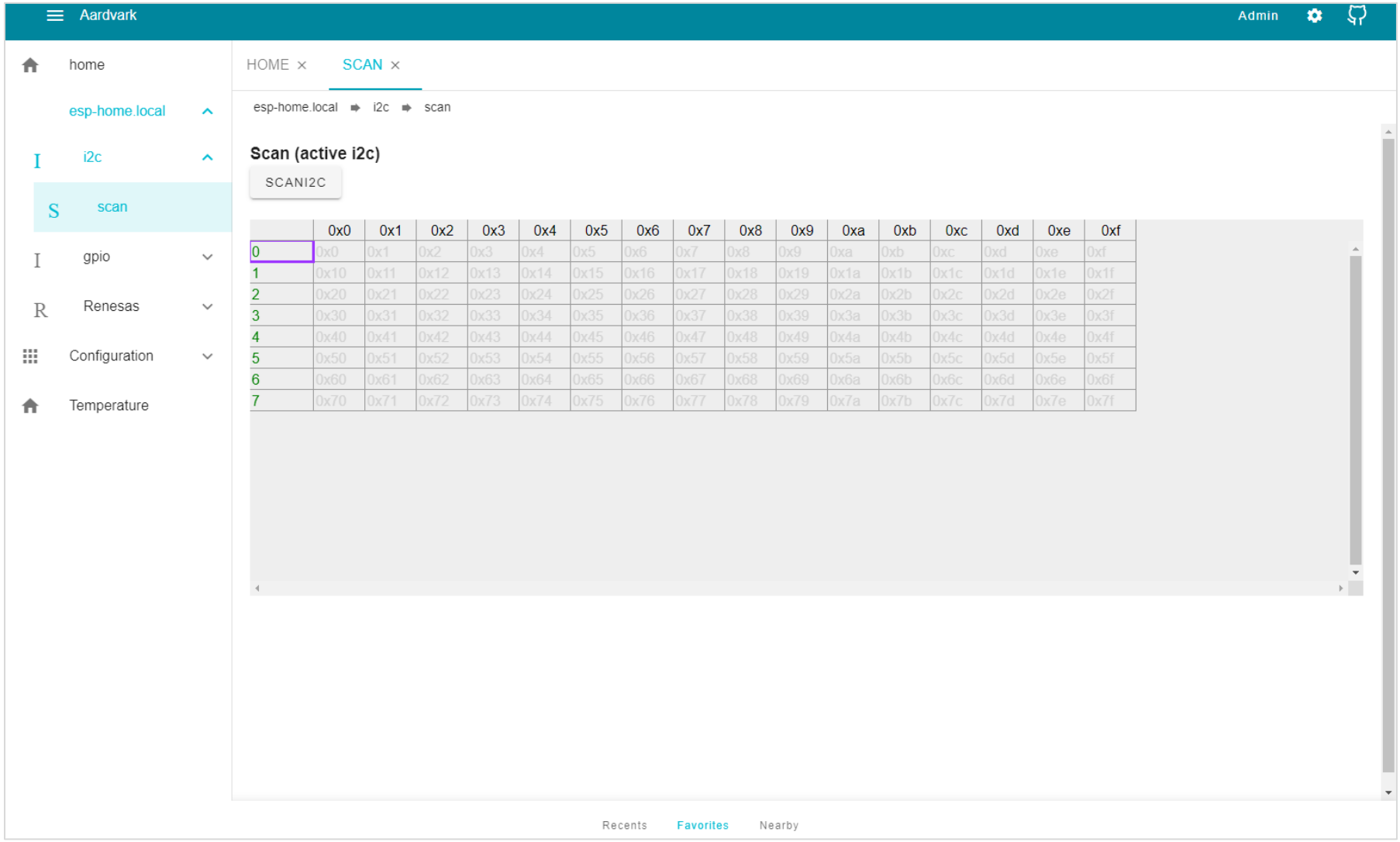
# Connect to ESP

- Once you are connected, esp-home.local will appear as an option in the left sidebar



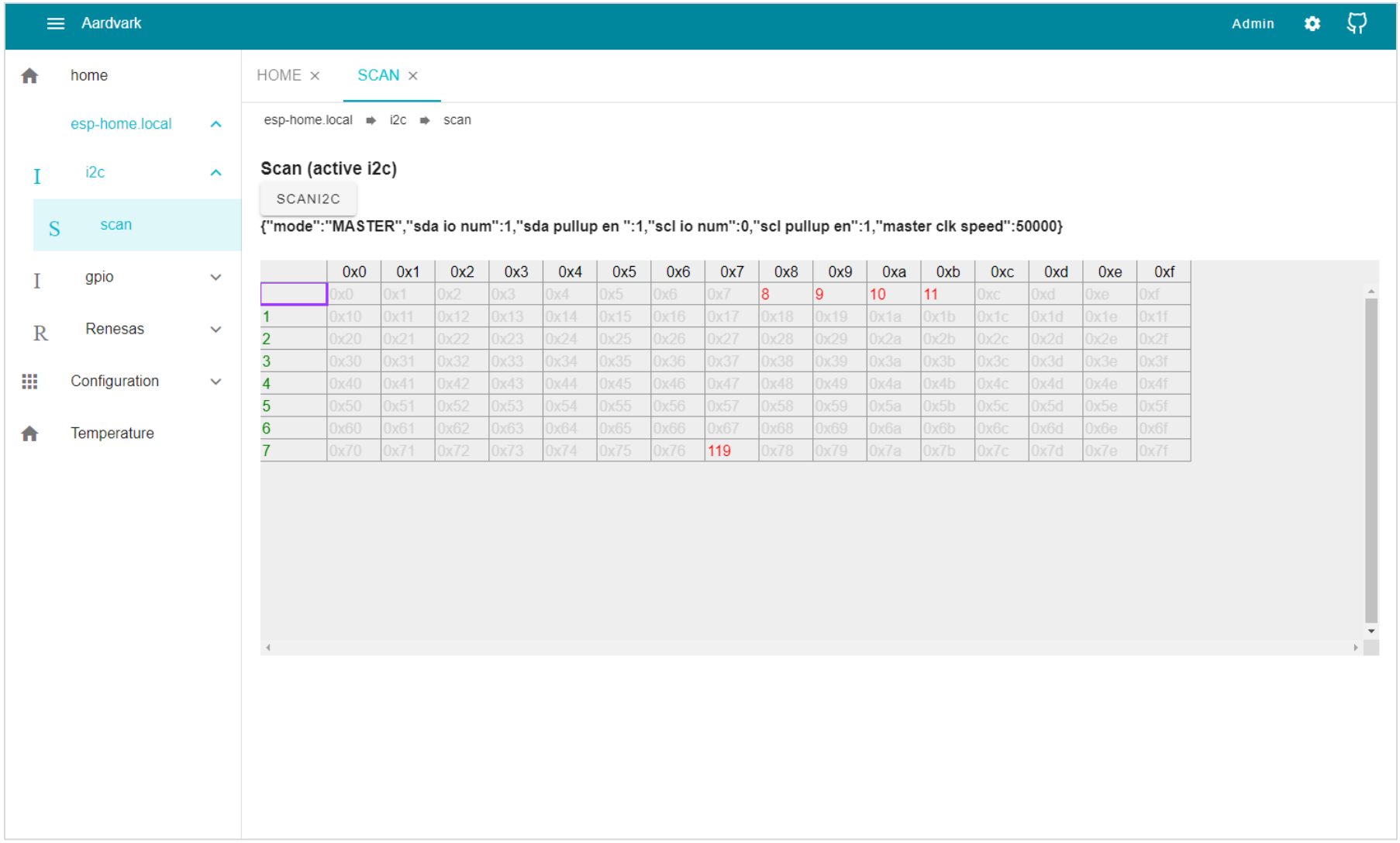
# Scan i2c connections

- To see the i2c connections, expand esp-home.local
- Then click on i2c->scan in the left sidebar
- This array of memory locations should appear



# Perform i2c Scan

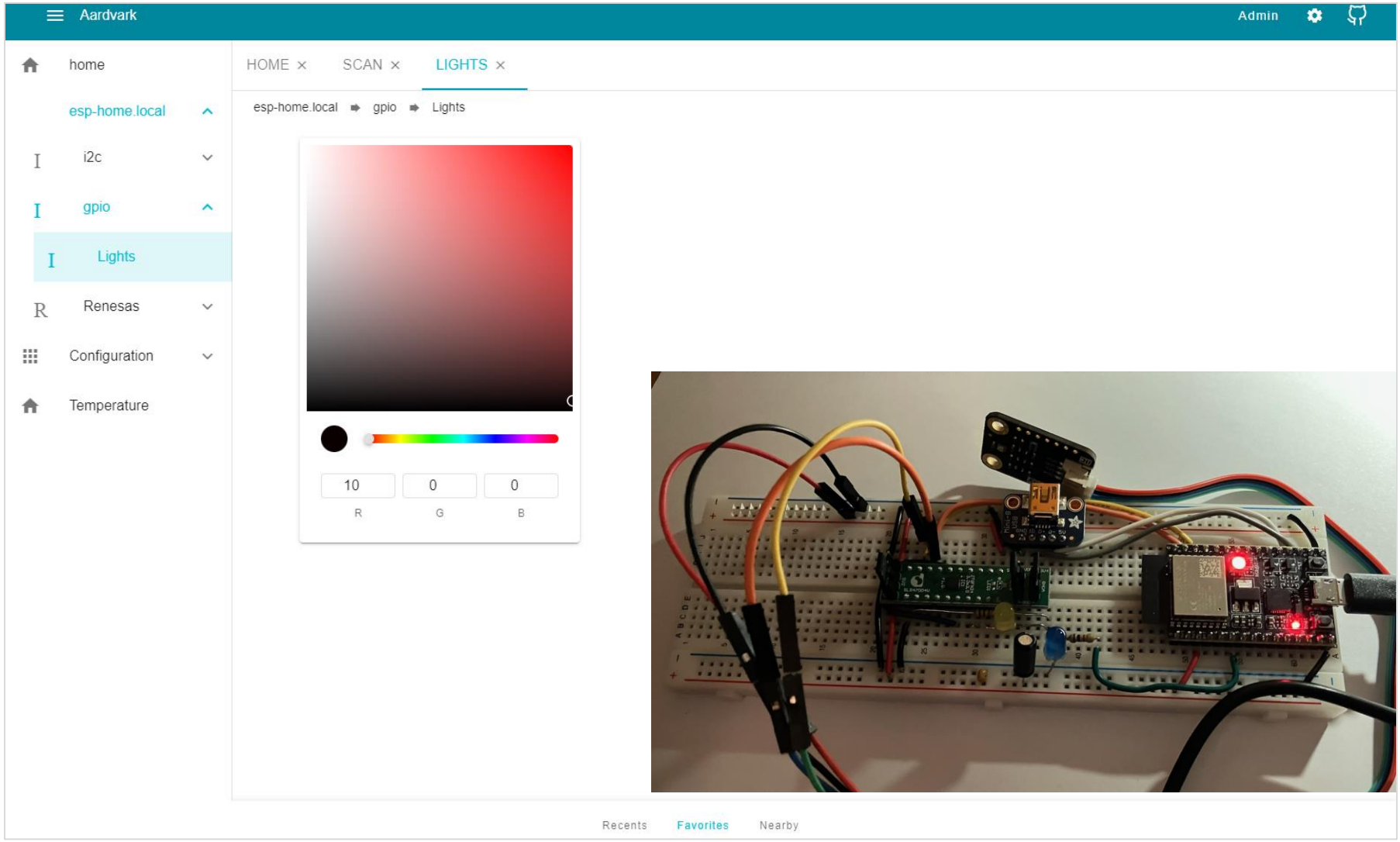
- Once the i2c array is visible, click on the SCANI2C button
- After a successful scan, some squares in the array will turn **red** and their values will change from hex -> decimal
- In this example, **119** indicates the i2c address of the attached DFRobot temperature sensor, and **8**,**9**,**10**,**11** indicate the i2c addresses of the FPGA




| | 0x0 | 0x1 | 0x2 | 0x3 | 0x4 | 0x5 | 0x6 | 0x7 | 0x8 | 0x9 | 0xa | 0xb | 0xc | 0xd | 0xe | 0xf |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 0x0 | 0x1 | 0x2 | 0x3 | 0x4 | 0x5 | 0x6 | 0x7 | 8 | 9 | 10 | 11 | 0xc | 0xd | 0xe | 0xf |
| 1 | 0x10 | 0x11 | 0x12 | 0x13 | 0x14 | 0x15 | 0x16 | 0x17 | 0x18 | 0x19 | 0x1a | 0x1b | 0x1c | 0x1d | 0x1e | 0x1f |
| 2 | 0x20 | 0x21 | 0x22 | 0x23 | 0x24 | 0x25 | 0x26 | 0x27 | 0x28 | 0x29 | 0x2a | 0x2b | 0x2c | 0x2d | 0x2e | 0x2f |
| 3 | 0x30 | 0x31 | 0x32 | 0x33 | 0x34 | 0x35 | 0x36 | 0x37 | 0x38 | 0x39 | 0x3a | 0x3b | 0x3c | 0x3d | 0x3e | 0x3f |
| 4 | 0x40 | 0x41 | 0x42 | 0x43 | 0x44 | 0x45 | 0x46 | 0x47 | 0x48 | 0x49 | 0x4a | 0x4b | 0x4c | 0x4d | 0x4e | 0x4f |
| 5 | 0x50 | 0x51 | 0x52 | 0x53 | 0x54 | 0x55 | 0x56 | 0x57 | 0x58 | 0x59 | 0x5a | 0x5b | 0x5c | 0x5d | 0x5e | 0x5f |
| 6 | 0x60 | 0x61 | 0x62 | 0x63 | 0x64 | 0x65 | 0x66 | 0x67 | 0x68 | 0x69 | 0x6a | 0x6b | 0x6c | 0x6d | 0x6e | 0x6f |
| 7 | 0x70 | 0x71 | 0x72 | 0x73 | 0x74 | 0x75 | 0x76 | 119 | 0x78 | 0x79 | 0x7a | 0x7b | 0x7c | 0x7d | 0x7e | 0x7f |

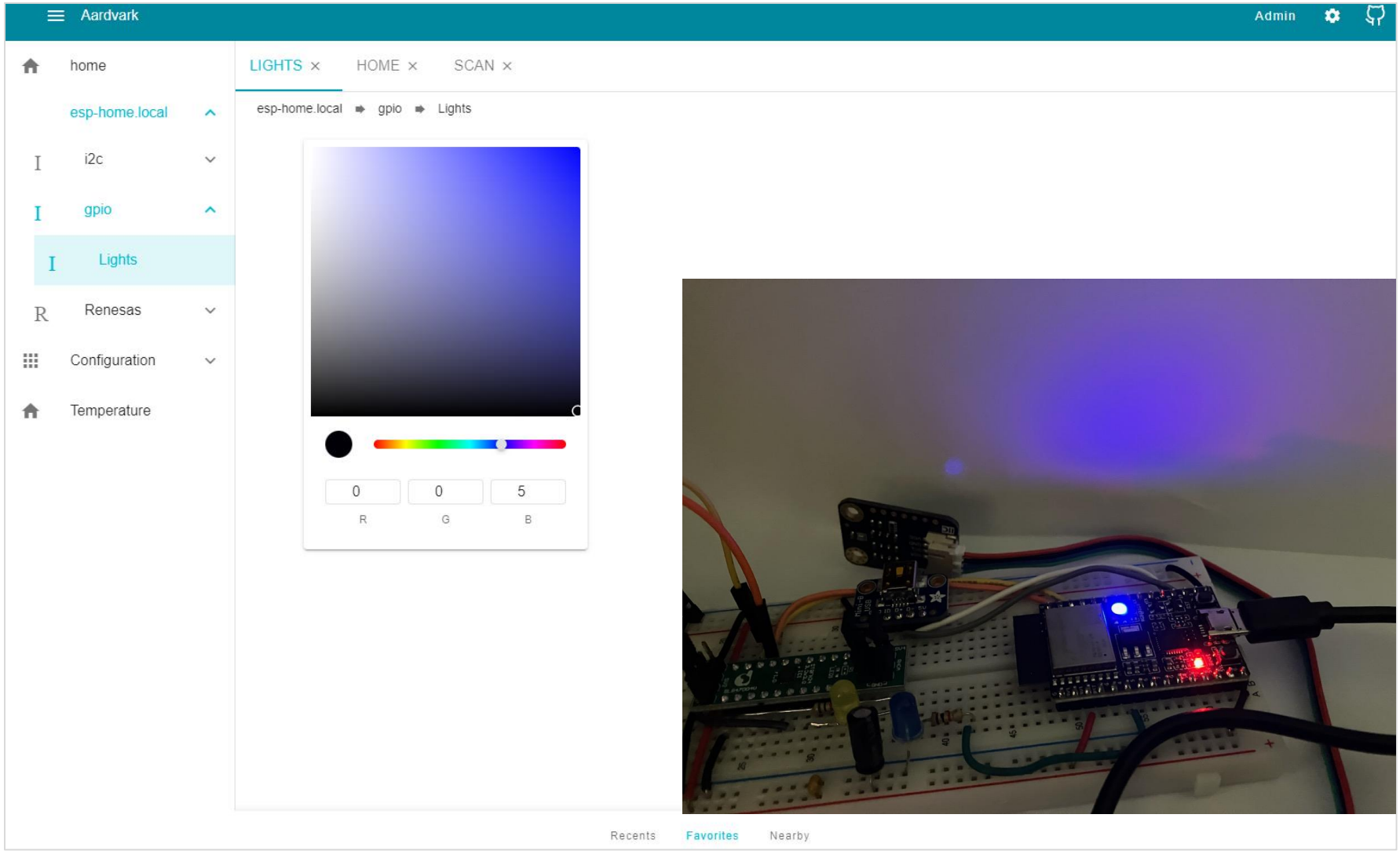
# Access GPIO Pins

- Expand gpio->Lights in the left sidebar
- The color picker will appear, preset to deep red, as indicated by the location of the small white circle.
- This color is communicated to the LED attached to the GPIO pins of the ESP32, as shown in the photo (uppermost of the two LEDs)



# Access GPIO Pins

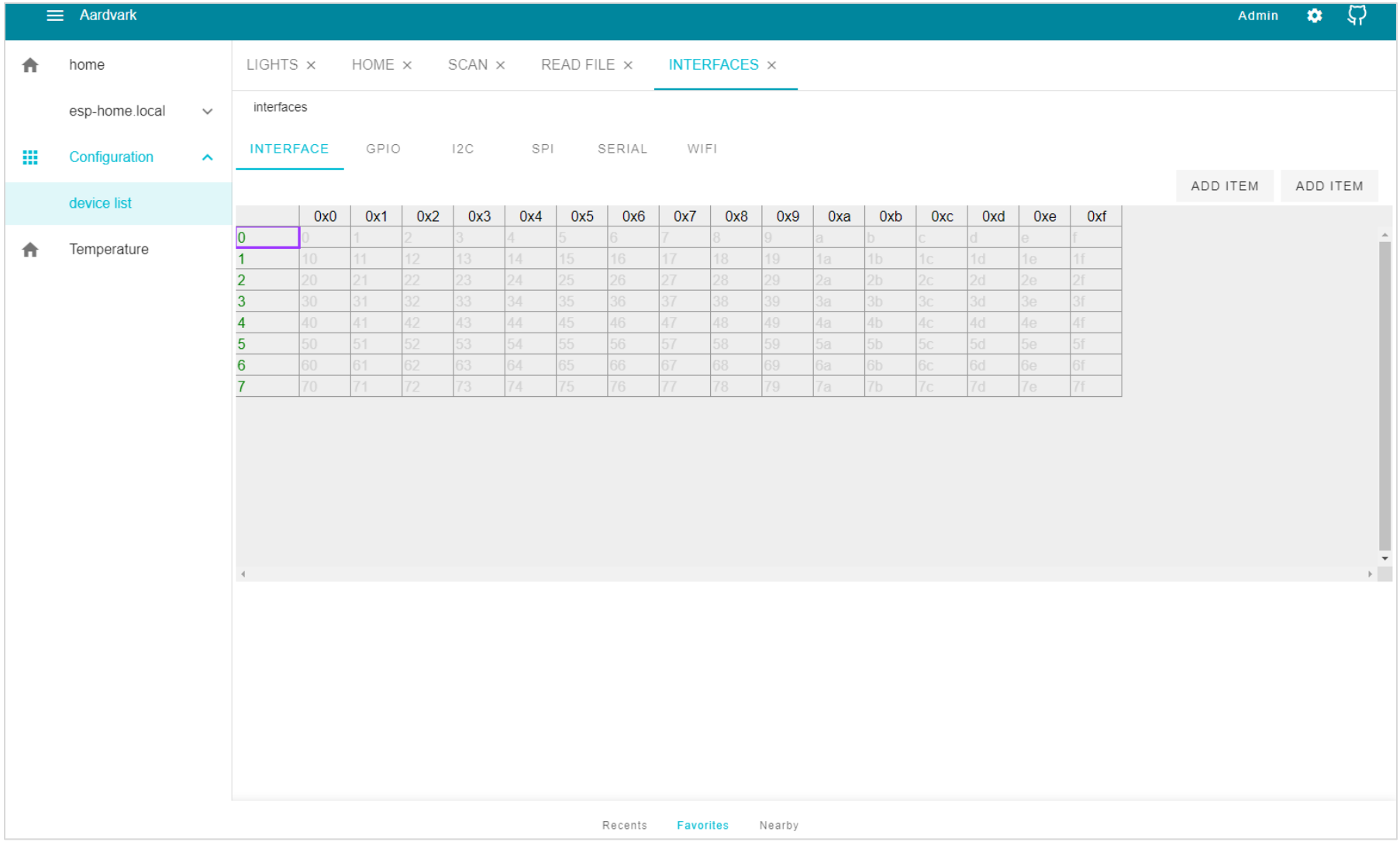
- Changing the slider position and/or the location of the small white circle using the color picker will change the color of the LED.
- The corresponding color change is shown in the photo.
- Communication between Aardvark and the ESP32/ant is occurring via the GPIO interface.



# Device Interfaces

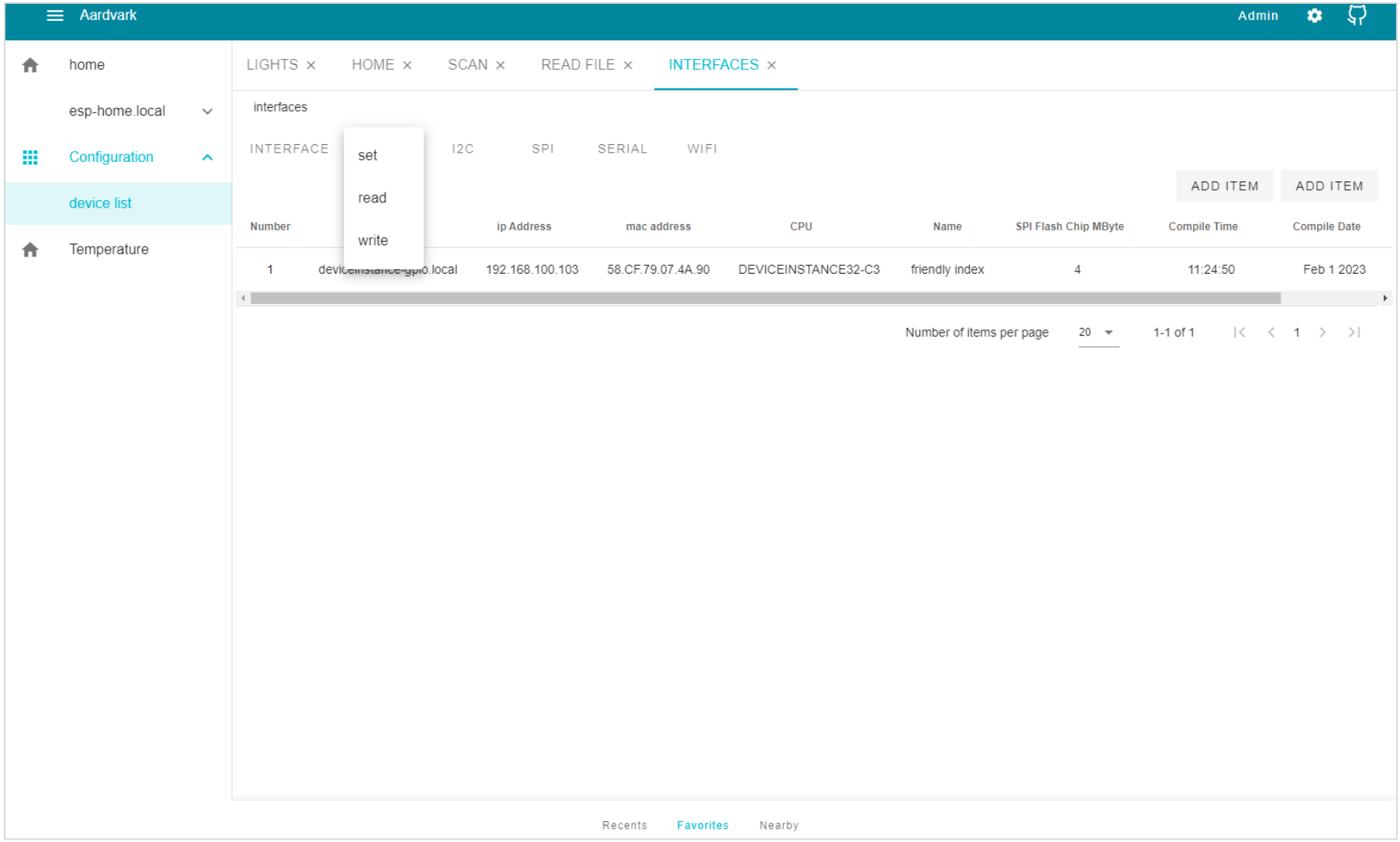
- Expanding Configuration->device list will allow for direct communication with devices connected to the ESP via several different protocols.

- This is not yet fully implemented in the stable release used for this demo, but the next few slides give an idea of what is to come.



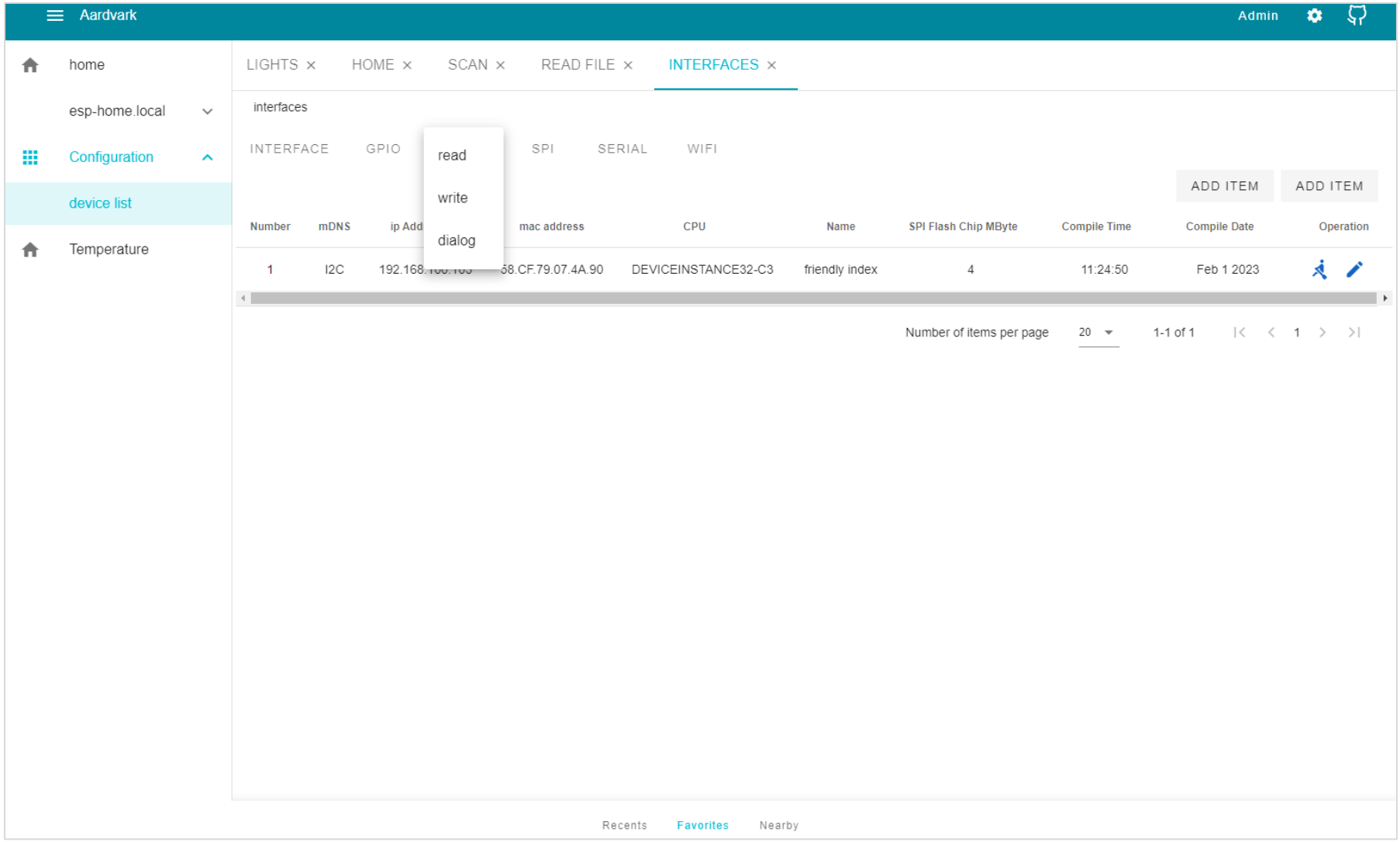
# GPIO Interface

- Clicking on the GPIO heading gives an example of the GPIO commands to be available from Aardvark.



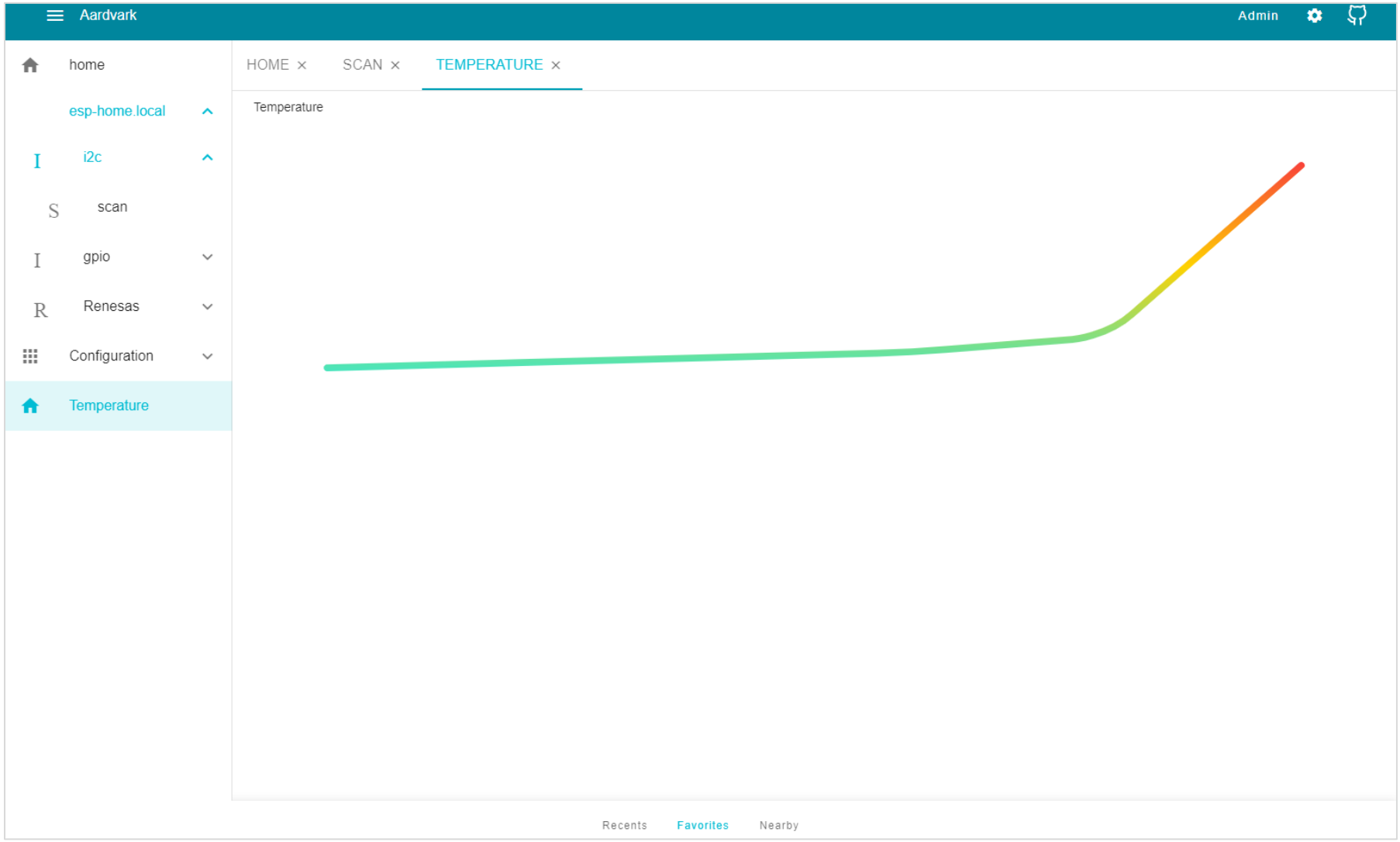
# I2C Interface

- Clicking on the I2C heading gives an example of the I2C commands to be available from Aardvark.
- SPI, SERIAL and WIFI commands are not available with the configuration used for this demo.



# Temperature - Start

- Clicking on Temperature in the left sidebar brings up a graph of the data currently being collected by the DFRobot Temperature Sensor.



# Temperature - Changes

- The Temperature graph changes in realtime as the sensor readings change.
- Though currently limited to the predefined DFRobot temperature sensor, the fully implemented graphing function will be available to any connected device.