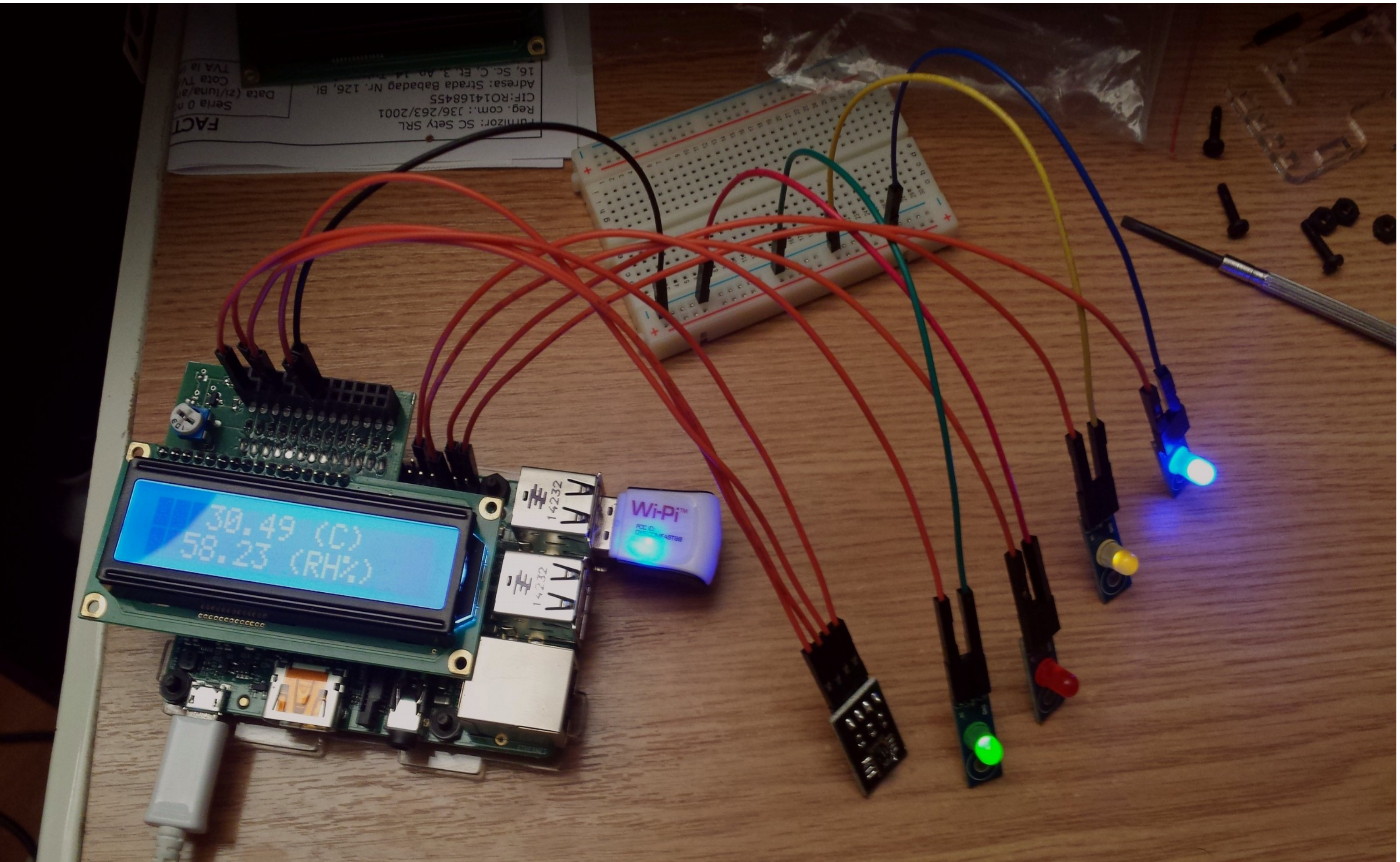
pi + sensor = weather station



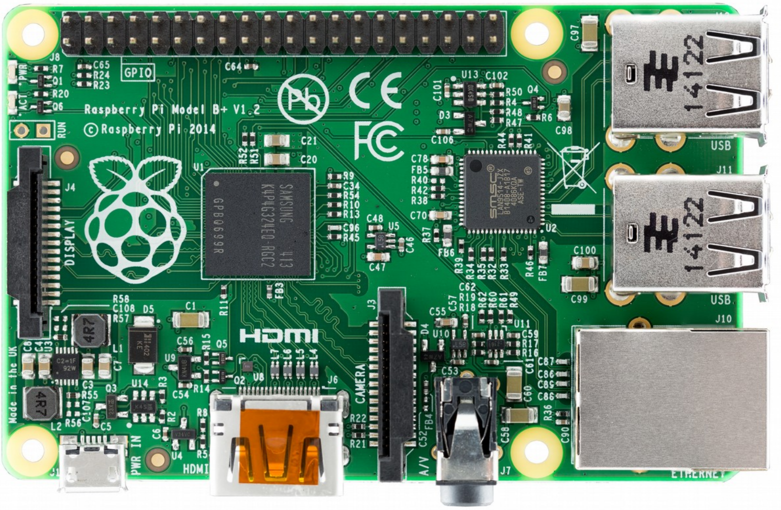
Nashwan Azhari

1.2

Contents:

- about the Raspberry PI
- technologies used for the project
- driving the components
- the weather station as a whole
- short demo
- Q&A
- closing thoughts

Raspberry PI B+

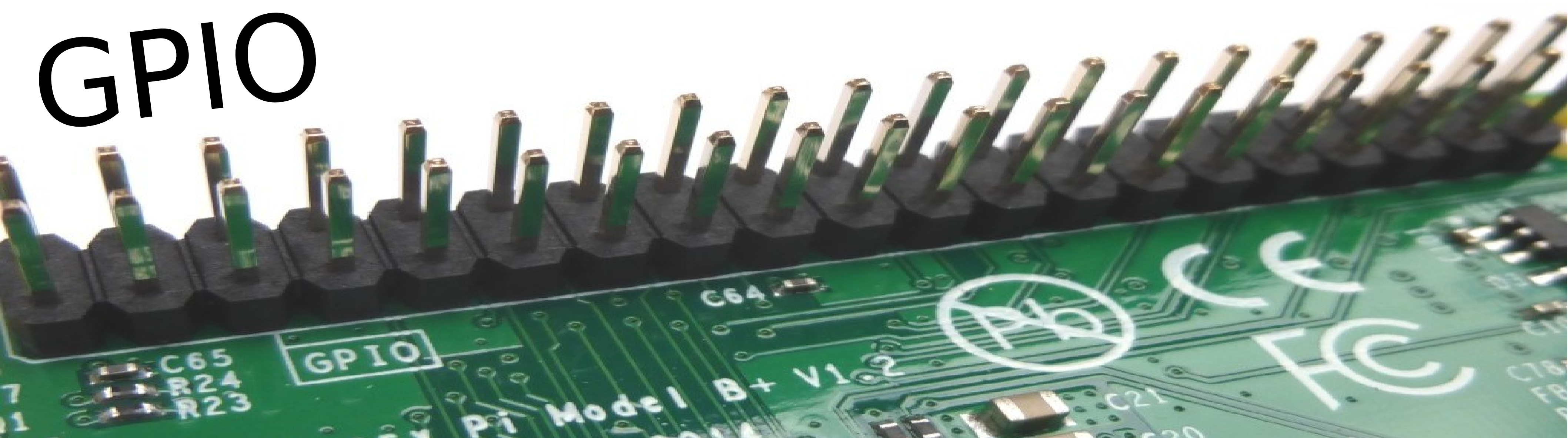


- full OS-capable PC for $35
- Broadcom SoC running at 700MHz with 512 MB RAM
- runs 32-bit ARM builds of Linux or the BSDs
- various peripherals including: 4xUSB, Ethernet, Composite A/V, HDMI
- General Purpose Input/Output header

- 40 in number
- have PWM support, but mostly used digitally
- have kernel drivers on all major OS-es

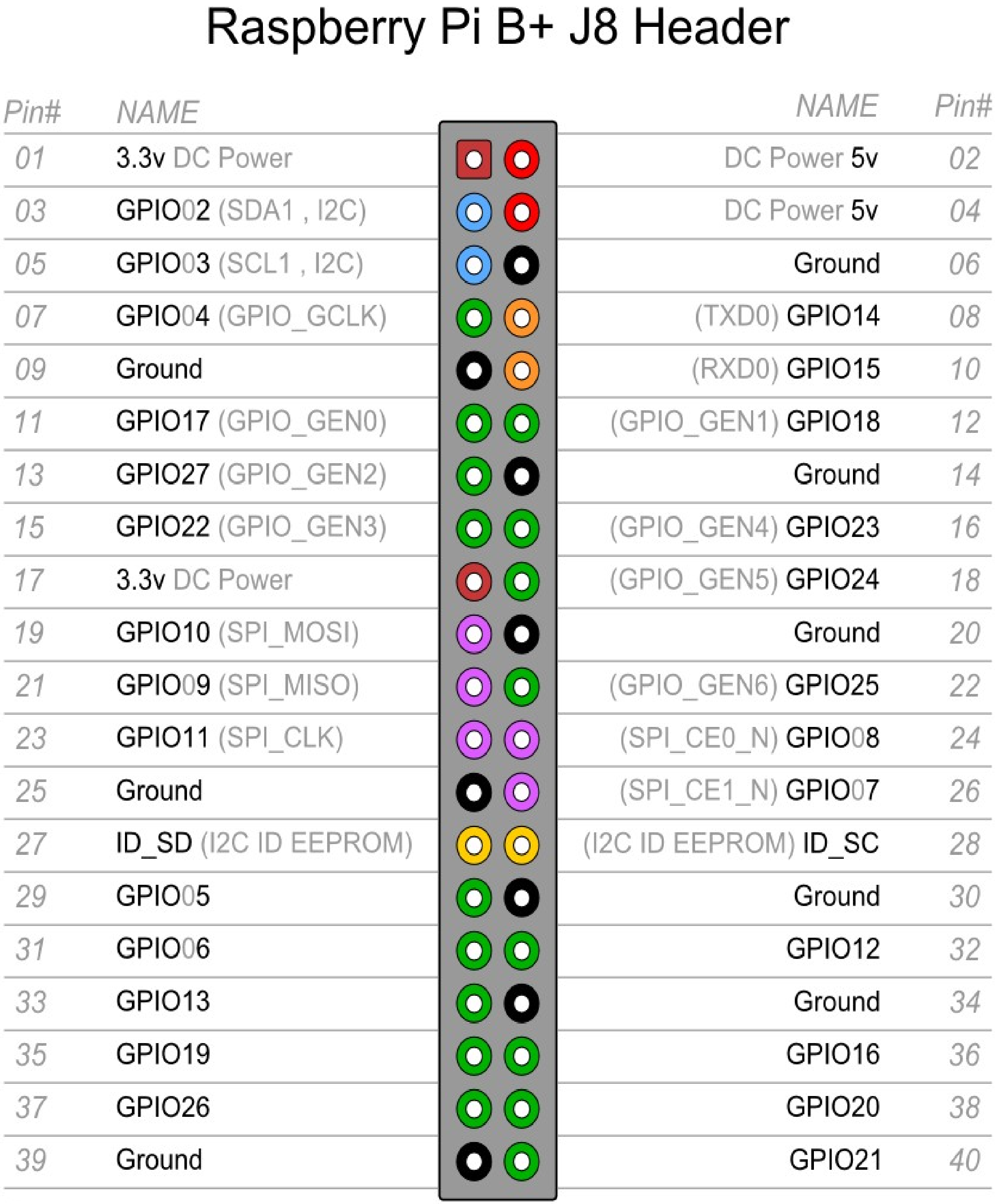
  (file descriptors can be found in /sys/class/gpio)
- have pre-made libraries for BASIC, C, C++, Java, Perl, Python and Lua

GPIO



lua-periphery

- C library
- Has functions for:
  - setting up
  - reading from
  - outputting to
  - cleaning up
- the GPIO pins

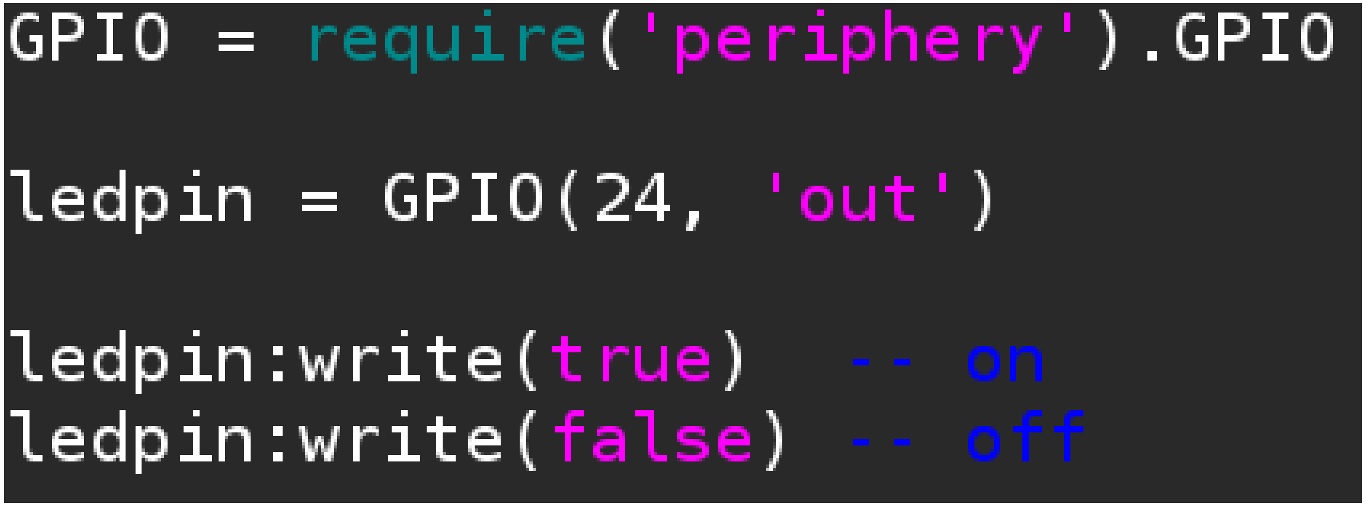


Raspberry Pi B+ J8 Header

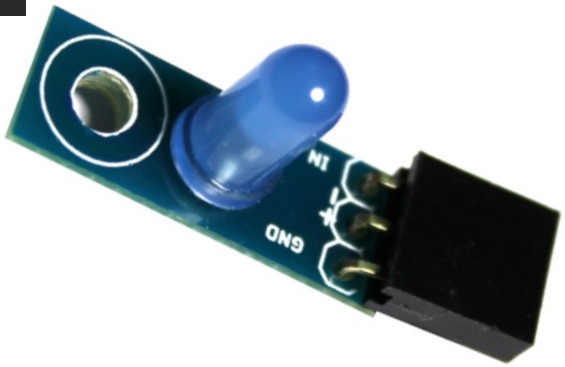
Pin#	NAME	NAME	Pin#
01	3.3v DC Power	DC Power 5v	02
03	GPIO02 (SDA1 , I2C)	DC Power 5v	04
05	GPIO03 (SCL1 , I2C)	Ground	06
07	GPIO04 (GPIO_GCLK)	(TXD0) GPIO14	08
09	Ground	(RXD0) GPIO15	10
11	GPIO17 (GPIO_GEN0)	(GPIO_GEN1) GPIO18	12
13	GPIO27 (GPIO_GEN2)	Ground	14
15	GPIO22 (GPIO_GEN3)	(GPIO_GEN4) GPIO23	16
17	3.3v DC Power	(GPIO_GEN5) GPIO24	18
19	GPIO10 (SPI_MOSI)	Ground	20
21	GPIO09 (SPI_MISO)	(GPIO_GEN6) GPIO25	22
23	GPIO11 (SPI_CLK)	(SPI_CE0_N) GPIO08	24
25	Ground	(SPI_CE1_N) GPIO07	26
27	ID_SD (I2C ID EEPROM)	(I2C ID EEPROM) ID_SC	28
29	GPIO05	Ground	30
31	GPIO06	GPIO12	32
33	GPIO13	Ground	34
35	GPIO19	GPIO16	36
37	GPIO26	GPIO20	38
39	Ground	GPIO21	40

Nashwan Azhari

5v brick LEDs

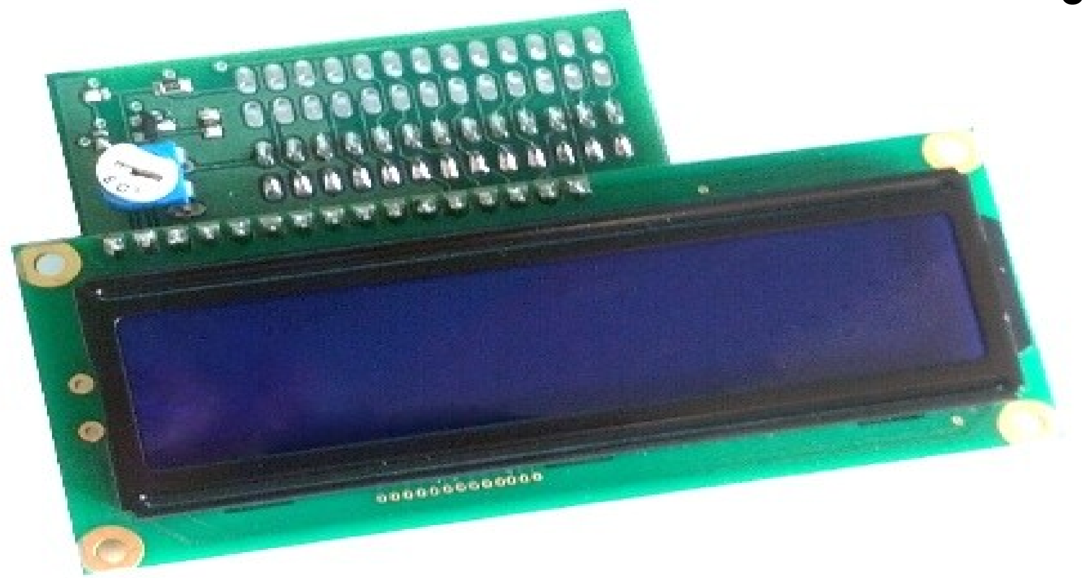


```
GPIO = require('periphery').GPIO

ledpin = GPIO(24, 'out')

ledpin:write(true)  -- on
ledpin:write(false) -- off
```



LCD

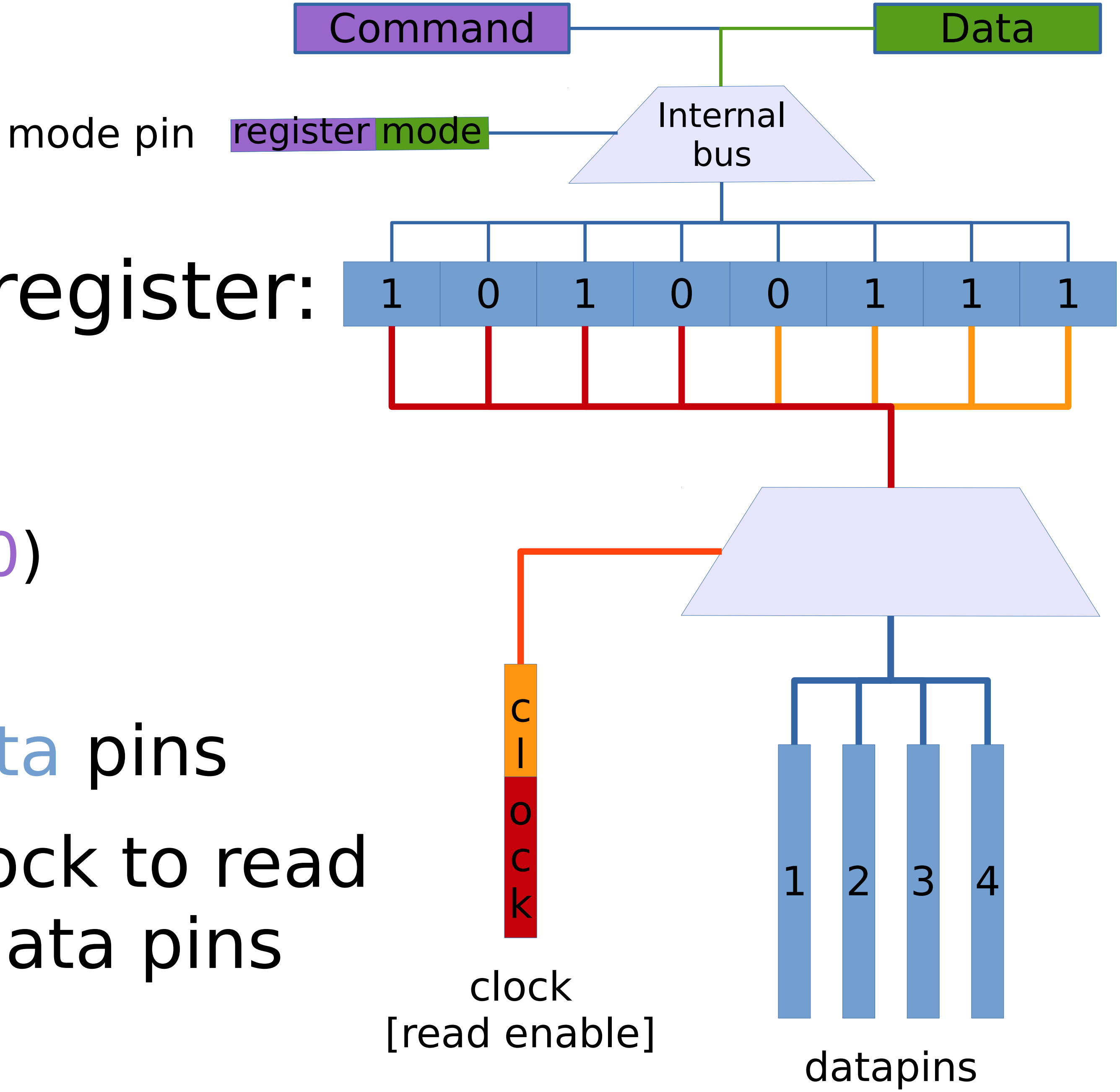
- Adafruit 16x2 LCD SHIELD
- 8 bit operation mode, fully ASCII capable



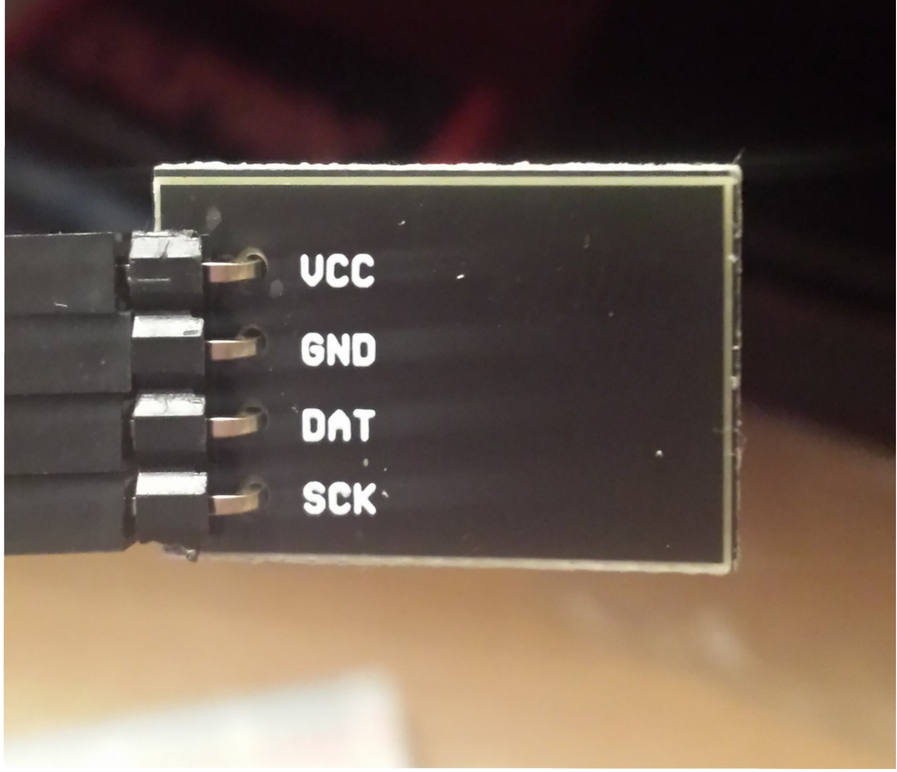
- uses 6 GPIO pins:
  - 1 mode selector
  - 1 serial clock
  - 4 data pins

Writing to the register:

- mode pin:
  - command (0)
  - data (1)
- output to data pins
- tick serial clock to read 4 bits from data pins

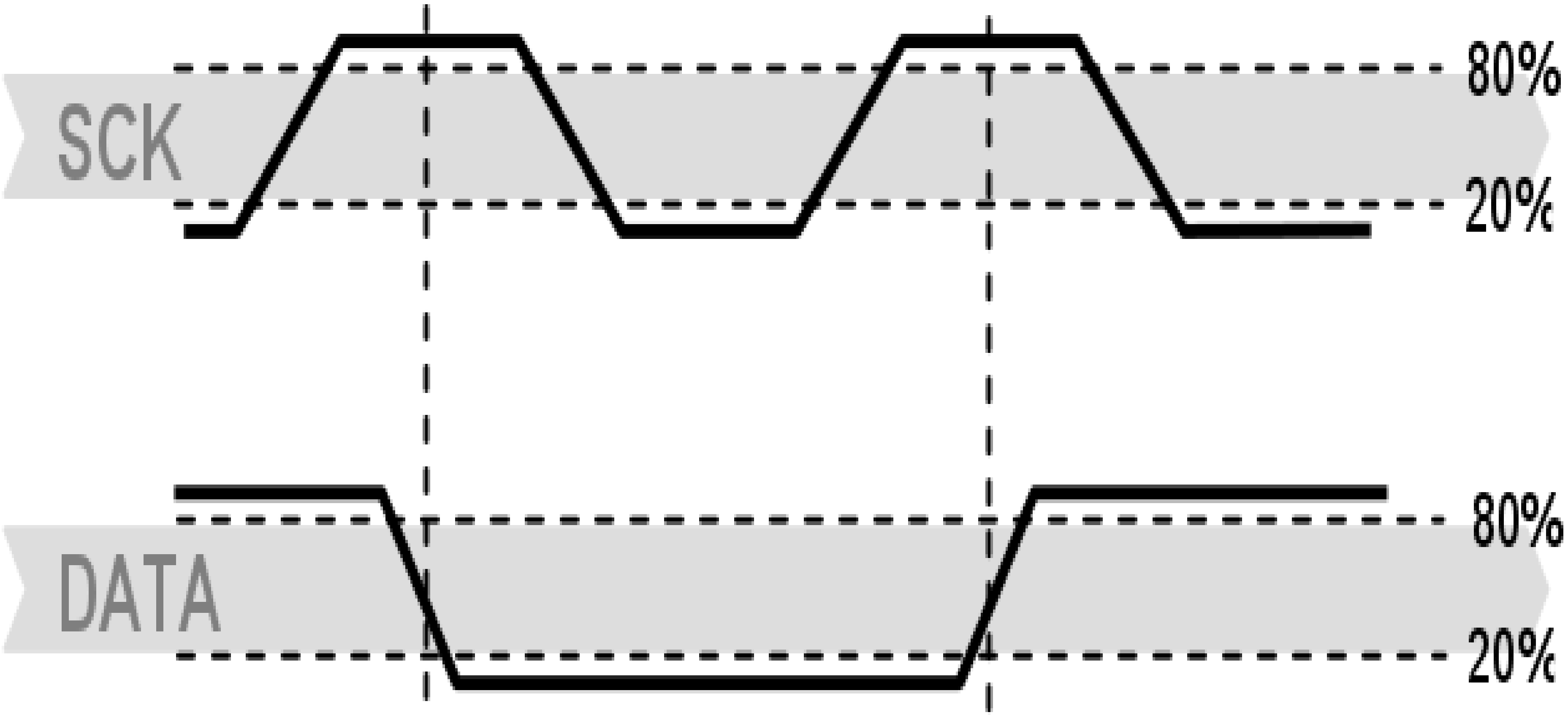


SHT11

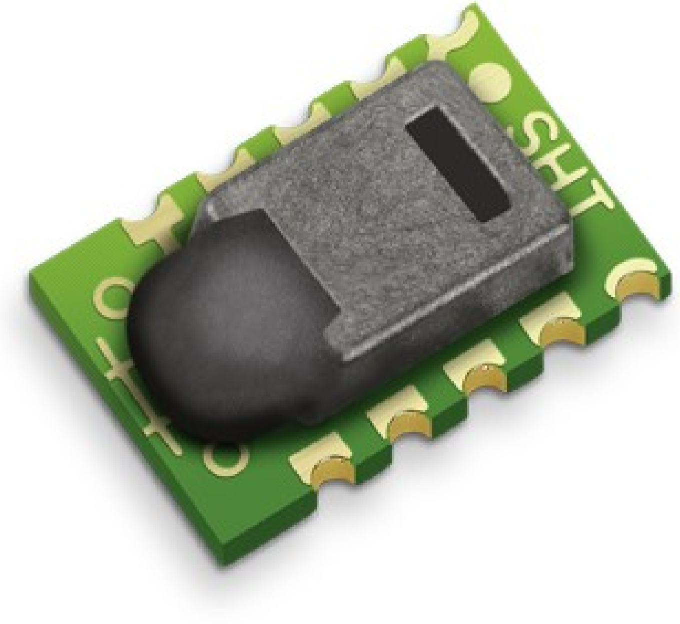


- temperature and humidity sensing digital IC
- VCC usually around 4v
- bi-directional DATA line
- All I/O coordinated by the Serial ClocK

Initiating SHT11 Transmission

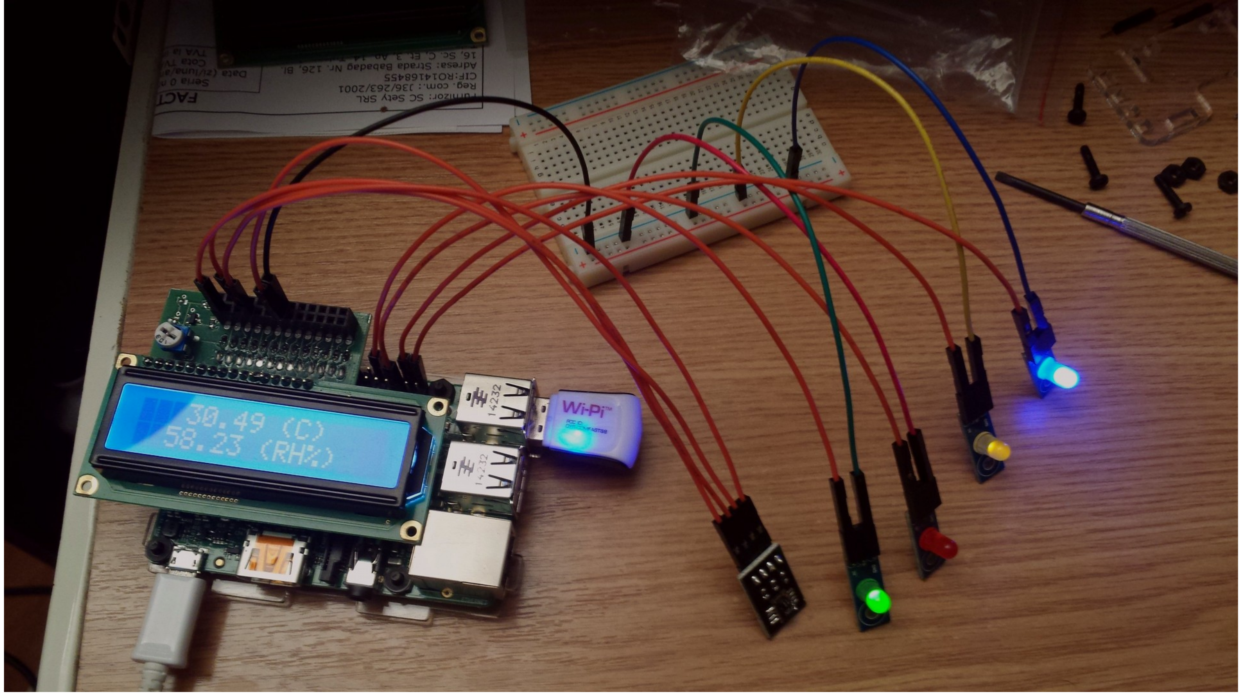


Communicating with the sensor:



- write 1 bit/clock cycle
- commands are 1 byte:
  - 0x03 for temperature
  - 0x05 for humidity
- results are on 2 bytes + 1 CRC byte
  - raw results require minor processing

PI-sense



- each component has a Lua table
- central WeatherStation table which:
  - reads all necessary parameters from config
  - queries the SHT11 for temperature and humidity
  - writes to the LCD
  - triggers the appropriate LEDs

DEMO

Q&A

github.com/aznashwan/lua-sense

Nashwan Azhari