

---

## Public Notes

[daniel@kaldheim.org](mailto:daniel@kaldheim.org), Daniel Rufus Kaldheim

2020-02-13

# Contents

<b>1</b>	<b>Todo</b>	<b>2</b>
<b>2</b>	<b>Projects</b>	<b>3</b>
<b>3</b>	<b>3D Printing</b>	<b>4</b>
<b>4</b>	<b>Electronics</b>	<b>5</b>
<b>5</b>	<b>Useful stuff</b>	<b>6</b>
<b>6</b>	<b>Reading List</b>	<b>7</b>
6.1	Topics . . . . .	7
6.1.1	Collision engine . . . . .	7
6.1.2	The OAuth 2.0 Authorization Framework . . . . .	7
6.1.3	Event Sourcing . . . . .	7
6.1.4	Micro frontends . . . . .	7
6.1.5	Micro services . . . . .	7
6.1.6	12 factor application . . . . .	7
6.1.7	RabbitMQ RPC . . . . .	7
6.2	Check out . . . . .	8
6.3	Video course . . . . .	8
<b>7</b>	<b>3D Printing</b>	<b>9</b>
7.1	CNC . . . . .	9
7.1.1	Laser cutter . . . . .	9
7.2	Snapmaker . . . . .	9
7.3	Ultimaker . . . . .	10
7.3.1	Repairs . . . . .	10
7.3.1.1	Nozzle . . . . .	10
	Ultimaker 2 - Removing the Nozzle . . . . .	10
<b>8</b>	<b>Amateur radio</b>	<b>11</b>
8.1	RTLSDR . . . . .	11
8.2	Articles . . . . .	11
8.3	Links . . . . .	11
8.4	BaoFeng UV-5R . . . . .	12
8.4.1	Features . . . . .	12
8.4.2	UV-5R SPECIFICATIONS . . . . .	13
8.4.3	Links . . . . .	14

<b>9</b>	<b>Electronics</b>	<b>15</b>
9.1	Canbus . . . . .	15
9.1.1	Tools . . . . .	15
9.1.2	Canbus addresses . . . . .	15
9.2	FTDI . . . . .	15
9.2.1	FT232 . . . . .	15
9.2.1.1	Links . . . . .	17
9.3	i2c . . . . .	17
9.3.1	Articles . . . . .	17
9.4	Led . . . . .	17
9.4.1	Links . . . . .	17
9.5	433mHz . . . . .	18
9.5.1	Tools . . . . .	18
9.5.2	PDF's . . . . .	18
9.6	Audio . . . . .	19
9.6.1	Articles . . . . .	19
9.6.2	Microphone Array . . . . .	19
9.6.2.1	Links . . . . .	19
9.6.3	Synthesizers . . . . .	19
9.6.3.1	Atari Punk Console . . . . .	19
	Links . . . . .	19
9.7	MAX3107 . . . . .	19
9.7.1	Datasheets . . . . .	20
9.8	Espressif . . . . .	20
9.8.1	ESP 32 . . . . .	20
9.8.1.1	Encryption . . . . .	20
9.8.2	ESP 8266 . . . . .	21
9.9	GSM Modules . . . . .	21
9.9.1	Adafruit Fona 3g cellular breakout . . . . .	21
9.9.1.1	GPS specifications . . . . .	22
9.9.1.2	Implementation . . . . .	22
9.9.1.3	Setup . . . . .	22
	Celluar . . . . .	23
	Realtime clock . . . . .	24
	Send SMS . . . . .	24
	Commands log (02.02.2020) . . . . .	25
	Important note . . . . .	25
9.9.1.4	Articles . . . . .	25
9.9.2	Sierra Wireless WP7607-1 . . . . .	25
9.9.2.1	Links . . . . .	26
9.10	Modbus . . . . .	26
<b>10</b>	<b>Hardware</b>	<b>27</b>
10.1	Oculus Rift . . . . .	27
10.1.1	Oculus Rift + Steam VR + Linux . . . . .	27

10.1.2 Links . . . . .	27
10.2 SDR . . . . .	27
10.2.1 LimeSDR . . . . .	27
10.2.2 SDRPlay . . . . .	27
<b>11 Machine learning</b>	<b>28</b>
11.1 Articles . . . . .	28
11.2 Chat bots . . . . .	28
11.2.1 Alternatives . . . . .	28
11.2.2 On premise . . . . .	28
11.3 Chaos . . . . .	28
<b>12 Physics</b>	<b>29</b>
12.1 Articles . . . . .	29
<b>13 Productivity</b>	<b>30</b>
13.1 Articles . . . . .	30
<b>14 Programming</b>	<b>31</b>
14.1 Sorting . . . . .	31
14.2 Hexagonal architecture . . . . .	31
14.3 Frameworks . . . . .	31
14.3.1 Firebase . . . . .	31
14.3.1.1 Alternatives . . . . .	31
14.3.2 Sapphire . . . . .	31
14.3.3 Spring boot . . . . .	31
14.3.3.1 Quarkus . . . . .	31
14.4 Programming languages . . . . .	32
14.4.1 C++ . . . . .	32
14.4.1.1 Articles . . . . .	32
14.4.1.2 Videos . . . . .	32
14.4.2 Golang . . . . .	32
14.4.2.1 Tips and tricks . . . . .	32
Vendors . . . . .	32
14.4.2.2 Links . . . . .	32
Web framework . . . . .	32
ORM . . . . .	32
GUI . . . . .	32
Web . . . . .	32
Div . . . . .	33
14.4.2.3 Articles . . . . .	33
14.4.2.4 Serial UART . . . . .	33
14.4.3 Hammerspoon . . . . .	33
14.4.4 Rust . . . . .	33
14.4.4.1 Links . . . . .	33
Web . . . . .	33

GUI . . . . .	33
ORM . . . . .	33
ESP32 . . . . .	33
<b>15 Projects</b>	<b>34</b>
15.1 Inspiration aka others awesome projects . . . . .	34
15.2 Status Light . . . . .	34
15.2.1 iOS app . . . . .	34
15.3 BMW Media Center . . . . .	34
15.3.1 Articles . . . . .	34
15.3.2 Shoppinglist for BMW . . . . .	34
15.3.2.1 Bmw controller . . . . .	34
15.4 Crudus Markdown Notes . . . . .	35
15.4.1 Platform . . . . .	35
15.4.1.1 iOS / Android . . . . .	35
15.4.1.2 Desktop . . . . .	35
15.4.2 Links . . . . .	35
15.4.3 Libraries . . . . .	35
15.4.3.1 JavaScript . . . . .	35
15.4.4 Other Editors . . . . .	35
15.5 Crudus Photos . . . . .	36
15.5.1 Tensor flow . . . . .	36
15.5.2 Articles . . . . .	36
15.5.3 Photo History . . . . .	36
15.5.4 Tools . . . . .	37
15.5.4.1 Ubuntu . . . . .	37
15.5.5 Links . . . . .	37
15.5.5.1 Caffe . . . . .	37
15.5.5.2 Model zoo . . . . .	37
15.5.5.3 Docker image . . . . .	37
15.5.5.4 Diff image . . . . .	37
15.5.5.5 Image Fingerprint . . . . .	37
15.5.5.6 Frame Hash . . . . .	37
15.5.5.7 Image recognition . . . . .	37
15.5.5.8 Duplicate images . . . . .	38
15.5.5.9 OpenCV Line detection . . . . .	38
15.5.5.10 Detect horizon . . . . .	38
15.5.5.11 OpenCV Auto-level / histogram . . . . .	38
15.5.5.12 OpenCV rotate images . . . . .	38
15.5.5.13 MIT Deep learning . . . . .	38
15.5.5.14 Tensorflow and docker . . . . .	38
15.5.5.15 OpenCV 4 . . . . .	38
15.6 Crudus Sense . . . . .	39
15.6.1 BLE device configuration specification . . . . .	39
15.6.2 MQTT publish Topics . . . . .	40

15.6.3 MQTT Subscribe Topics . . . . .	40
15.6.4 Extensions . . . . .	40
15.7 Kaldheim.org . . . . .	41
15.7.1 Links . . . . .	41
15.8 Maximus . . . . .	41
15.8.1 Components . . . . .	41
15.8.2 Robotics . . . . .	41
15.8.3 Articles . . . . .	41
15.8.4 Artificial Intelligence . . . . .	41
15.8.5 BNO055 . . . . .	41
15.8.5.1 Installation . . . . .	41
i2c configuration . . . . .	41
Create new project . . . . .	43
15.8.5.2 PID controller . . . . .	43
15.8.5.3 Videos . . . . .	43
15.8.5.4 Links . . . . .	43
15.8.5.5 Documents . . . . .	43
15.8.5.6 Books . . . . .	43
15.8.6 Development board . . . . .	44
15.8.6.1 Raspberry Pi pinout . . . . .	46
15.8.7 Dynamixel AX-12A . . . . .	46
15.8.7.1 Specification . . . . .	47
15.8.7.2 Wiring . . . . .	47
15.8.7.3 TTL communications . . . . .	48
15.8.7.4 Drawings . . . . .	49
15.8.7.5 Links . . . . .	49
15.8.8 GPS . . . . .	49
15.8.8.1 Adafruit ultimate GPS breakout . . . . .	49
Technical details . . . . .	50
Links . . . . .	51
15.8.8.2 GPS antenna . . . . .	51
Links . . . . .	51
Datasheets . . . . .	52
15.8.9 Hardware . . . . .	52
15.8.9.1 Jetson Nano . . . . .	52
15.8.9.2 Raspberry Pi 4 - Model B . . . . .	52
15.8.10 Maximus AI . . . . .	53
15.8.10.1 For terminal conversations . . . . .	53
15.8.10.2 AIML . . . . .	53
15.8.10.3 Unicode . . . . .	53
15.8.10.4 Artificial Intelligence . . . . .	53
15.8.10.5 Words, spelling and so on . . . . .	54
15.8.10.6 Grammar . . . . .	54
15.8.10.7 NLP / NER . . . . .	54
15.8.10.8 Intent parser . . . . .	54

15.8.10.9	Object recognition (caffe)	54
15.8.10.10	Image analyze	54
15.8.10.11	Automatic speech recognition	55
15.8.10.12	Questions / answers	55
15.8.10.13	Bucida	55
15.8.10.14	Animations	55
15.8.10.15	OCR / Deep learning	55
15.8.10.16	Neural network (arduino)	55
15.8.10.17	Other Links	55
15.8.11	Inspiration	55
15.8.11.1	Read	55
15.8.11.2	DyRET Robot	56
15.8.11.3	Open Cat	57
15.8.11.4	Pet dog	57
15.9	Mechanical keyboard	58
15.9.1	Inspiration	58
15.9.2	Tools	58
15.10	Motorcycle App	58
15.11	Pip-Boy	60
15.11.1	Montering	60
15.11.2	LCD skjerm	60
15.12	Project Management System	60
15.12.1	Inspiration	60
15.13	RPI Security camera	60
15.13.1	Articles	60
15.14	Reflow Oven	60
15.14.1	Links	60
15.15	USB Media Controller	61
15.15.1	Dimensions	61
<b>16</b>	<b>Security</b>	<b>62</b>
16.1	Articles	62
16.2	LoRaWAN	62
<b>17</b>	<b>Shopping lists</b>	<b>63</b>
17.1	Shopping list for home office	63
17.1.1	Keyboard	63
17.1.2	Network	63
<b>18</b>	<b>Software</b>	<b>64</b>
18.1	Rabbit MQ	64
18.1.1	Security	64
18.2	Ubuntu	64
18.2.1	Articles	64
18.2.2	Ubuntu + Windows dual boot	64
18.2.2.1	Recovering GRUB after reinstalling Windows	64

18.2.2.2 Master Boot Record backup and replacement . . . . .	64
18.3 tmux shortcuts & cheatsheet . . . . .	65
18.3.1 Sessions . . . . .	66
18.3.2 Windows (tabs) . . . . .	66
18.3.3 Panes (splits) . . . . .	66
18.3.4 Sync Panes . . . . .	66
18.3.5 Resizing Panes . . . . .	67
18.3.6 Copy mode . . . . .	67
18.3.7 Misc . . . . .	68
18.3.8 Configurations Options . . . . .	68
18.3.9 Resources . . . . .	69
18.4 Vim Cheat Sheet . . . . .	69
18.4.1 Global . . . . .	69
18.4.2 Cursor movement . . . . .	69
18.4.3 Insert mode - inserting/appending text . . . . .	70
18.4.4 Editing . . . . .	71
18.4.5 Cut and paste . . . . .	71
18.4.6 Marking text (visual mode) . . . . .	72
18.4.7 Visual commands . . . . .	72
18.4.8 Registers . . . . .	72
18.4.9 Search and replace . . . . .	72
18.4.10 Search in multiple files . . . . .	73
18.4.11 Working with multiple files . . . . .	73
18.4.12 Tabs . . . . .	73
18.4.13 Exiting . . . . .	74
<b>19 UX - UI</b>	<b>75</b>
19.1 Methods . . . . .	75
19.2 Colors . . . . .	75
19.2.1 Links . . . . .	75
<b>20 Useful stuff</b>	<b>76</b>
20.1 Useful Commands . . . . .	76
20.1.1 Terminal recording . . . . .	76
20.1.1.1 1. Install . . . . .	76
20.1.1.2 2. Record . . . . .	76
20.1.1.3 3. Play . . . . .	76
20.1.2 WiFi QR-code . . . . .	76
20.1.3 Rsync . . . . .	76
20.1.4 Unite PDF documents . . . . .	77
20.1.4.1 Install . . . . .	77
20.1.4.2 Usage . . . . .	77



Daniel's Public Notes

[Reading list](#)

# 1 Todo

- ☐ Do this
- ☒ Done that

## 2 Projects

- [Projects](#)
- [Crudus MD Notes](#)
- [Crudus Sense](#)
- [Crudus Photos](#)
- [Maximus](#)

## 3 3D Printing

- 3D printing

## 4 Electronics

- [Electronics](#)

## 5 Useful stuff

- [Markdown Cheatsheet](#)
- [Useful commands](#)

## 6 Reading List

- Elixir phoenix absinthe graphql react apollo <https://schneider.dev/blog/elixir-phoenix-absinthe-graphql-react-apollo-absurdly-deep-dive/>
- Uber design: <http://simonpan.com/work/uber/>
- Modern GPS Tracking Platform: <https://www.traccar.org>

### 6.1 Topics

#### 6.1.1 Collision engine

- <https://gamedev.stackexchange.com/questions/26501/how-does-a-collision-engine-work>

#### 6.1.2 The OAuth 2.0 Authorization Framework

- <https://tools.ietf.org/html/rfc6749>

#### 6.1.3 Event Sourcing

- <https://www.martinfowler.com/eaDev/EventSourcing.html>

#### 6.1.4 Micro frontends

- <https://www.martinfowler.com/articles/micro-frontends.html>

#### 6.1.5 Micro services

- <https://www.martinfowler.com/microservices/>

#### 6.1.6 12 factor application

- <https://12factor.net/>

#### 6.1.7 RabbitMQ RPC

- <https://www.rabbitmq.com/tutorials/tutorial-six-python.html>

## 6.2 Check out

- [My business card runs linux](#)
- <https://www.envoyproxy.io/docs/envoy/latest/start/start>
- <https://github.com/heptio/contour>
- <https://www.jaegertracing.io/>
- <https://istio.io/>

## 6.3 Video course

- <https://www.linkedin.com/learning/jhipster-build-and-deploy-spring-boot-microservices/welcome>
- <https://www.linkedin.com/learning/microservices-asynchronous-messaging/getting-work-done-in-microservices>
- <https://vimeo.com/74589816>
- <https://vimeo.com/99531595>
- <https://www.infoq.com/presentations/migration-cloud-native/>



## 7 3D Printing

- [Ultimaker](#)
- [Snapmaker](#)

### 7.1 CNC

#### 7.1.1 Laser cutter

- [Lasersaur](#)

### 7.2 Snapmaker

<https://forum.snapmaker.com/t/reverse-engineering-the-module-wiring/3031>

#### 3D Printing Module:

```
PIN1: VCC, Heater Socket Pin 1, Fan+
PIN2: Stepper Coil A+
PIN3: Heater Socket Pin 2
PIN4: Stepper Coil A-
PIN5: Thermistor Socket Pin 1
PIN6: Stepper Coil B-
PIN7: GND, Fan-, Thermistor Socket Pin 2
PIN8: Stepper Coil B+
```

#### Heated Build Plate:

```
PIN1: Heating Element +
PIN2: UNUSED
PIN3: Heating Element -
PIN4: UNUSED
PIN5: Thermistor +
PIN6: Thermistor -
```

The heating element registered as 12Ohms so 48W at 24V. The Thermistor gave a reading of 80kOhm in my 90 degree F garage.

#### Linear Module:

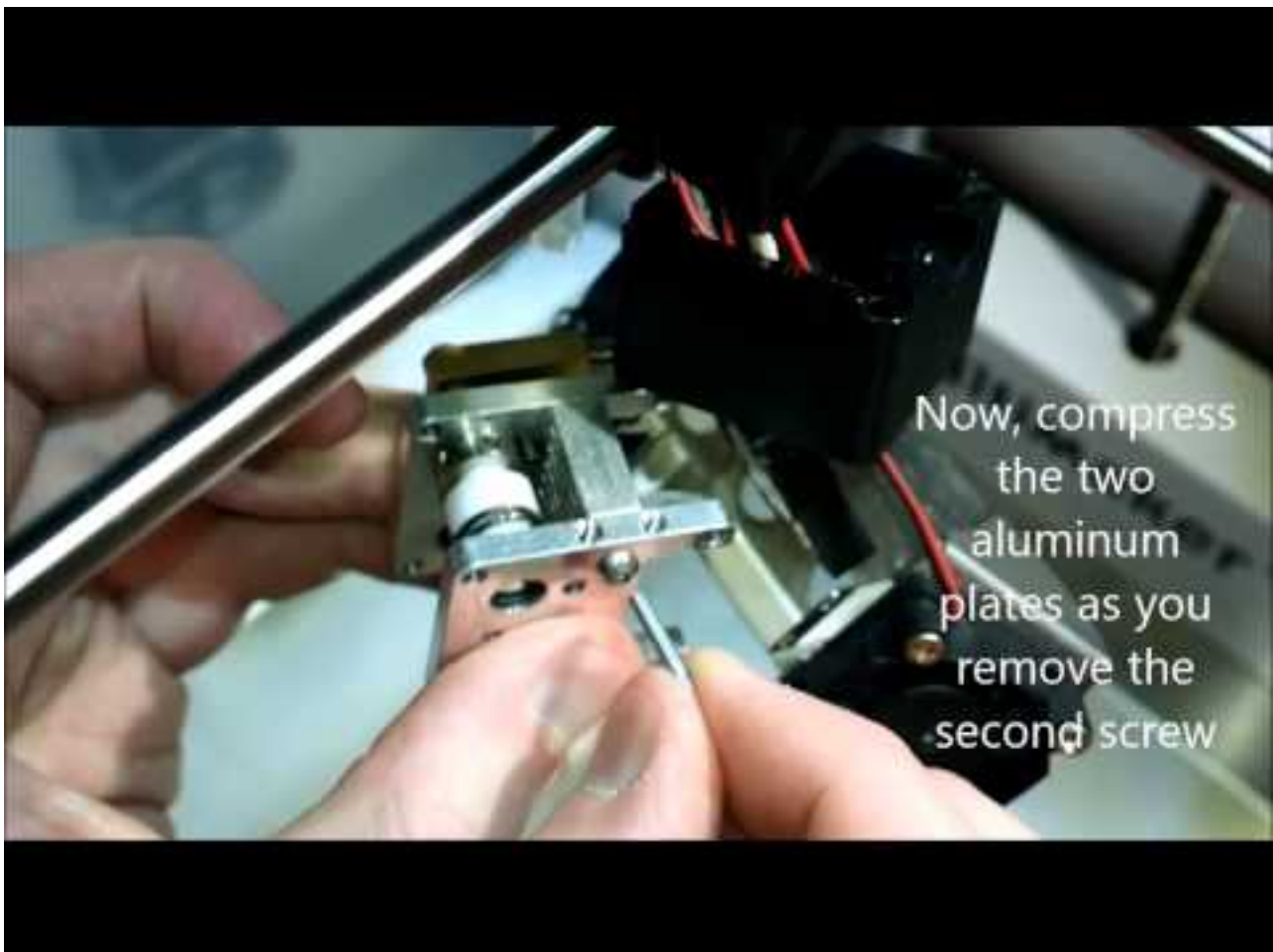
PIN1: Coil A +  
PIN2: Coil A -  
PIN3: Coil B +  
PIN4: Limit Switch +  
PIN5: Coil B -  
PIN6: Limit Switch -

## 7.3 Ultimaker

### 7.3.1 Repairs

#### 7.3.1.1 Nozzle

Ultimaker 2 - Removing the Nozzle <https://www.youtube.com/watch?v=-1Nh0snHLYw>



## 8 Amateur radio

### 8.1 RTLSDR

- [RTLSDR](#)

### 8.2 Articles

- [Walkie Talkies](#)

### 8.3 Links

- [Ham radio Frequency Chart](#)
- [Norske Frekvenser](#)

## 8.4 BaoFeng UV-5R



**Figure 8.1:** Baofeng

### 8.4.1 Features

25KHz/12.5KHz Switchable (Wide/Narrow Band)

FM Radio (65.0MHz-108.0MHz)

Large Inverted LCD Display

LED Flashlight

High /Low RF Power Switchable

VOX

50 CTCSS/ 104 DCS Tones

Tone searching/scanning

Dual standby

PC programmable

Transmitter time-out timer(TOT)

Busy channel lock-out(BCLO)

### 8.4.2 UV-5R SPECIFICATIONS

Frequency range:

[TX] 136 - 174MHz, 400 - 520MHz

[RX] 136 - 174MHz, 400 - 520MHz, 68-108MHz (FM Broadcast)

Channel Capacity:

128 Channels

Channel Spacing

25KHz (wide band)12.5KHz (narrow band)

Sensitivity

$\leq 0.25\mu\text{V}$  (wide band)  $\leq 0.35\mu\text{V}$  (narrow band)

Operation Voltage

7.4V DC  $\pm 20\%$

Battery:

1500mAh

Frequency step:

2.5, 5, 6.25, 10, 12.5, 20, 25, 30 and 50KHz

Antenna:

Antenna Connector: SMA-Female / Antenna Impedance:  $50\Omega$

Accessory Connector:

Kenwood 2 Pin Standard

Stability:

$\pm 2.5\text{ppm}$

Output power:

5W / 1W

Audio Power Output

700mW/10%

### 8.4.3 Links

- [Manual](#)

## 9 Electronics

- [I2C](#)

### 9.1 Canbus

#### 9.1.1 Tools

- <https://github.com/erimoq/cantools>

#### 9.1.2 Canbus addresses

- <https://community.carloop.io/t/list-of-can-id-descriptions-from-opengarages-org/104>
- <http://www.loopybunny.co.uk/CarPC/can/267.html>

### 9.2 FTDI

#### 9.2.1 FT232

This version have 3.3V. The data signals are at 3V and the power line provides 5V. We suggest this for any product that needs FTDI cables. Because the cable is 5V-logic compliant, you can use it with 3v or 5v logic just fine - no level shifting required!.

[Adafruit product page](#)

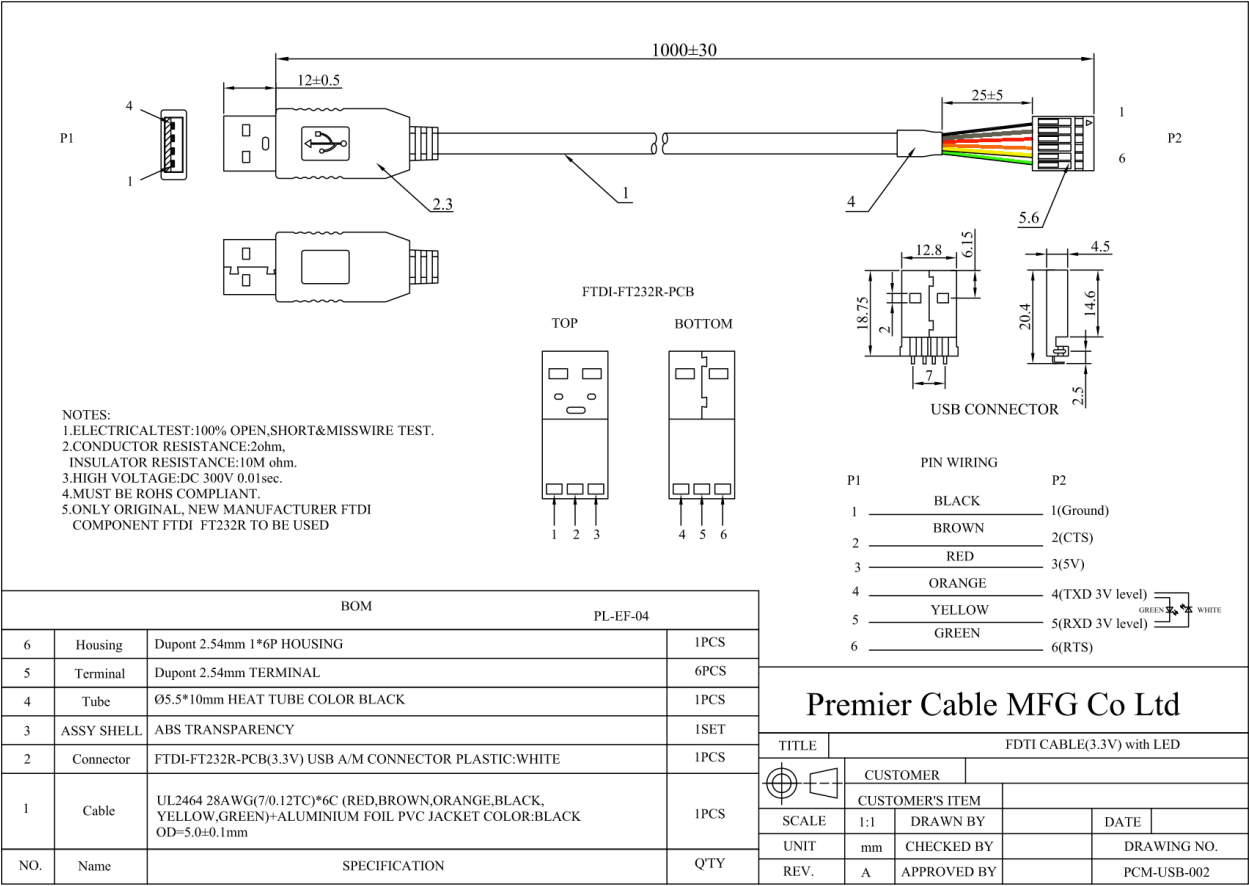


Figure 9.1: FTDI - FT232





**Figure 9.2:** Cable

#### 9.2.1.1 Links

- [DS\\_TTL-232R\\_CABLES\\_V201.pdf](#)

### 9.3 i2c

#### 9.3.1 Articles

- [I2C in a nutshell](#)

### 9.4 Led

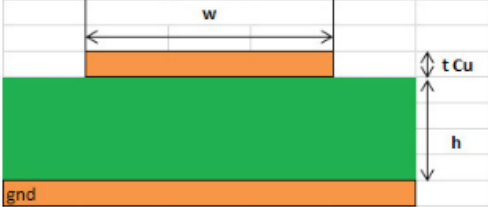
#### 9.4.1 Links

- <https://www.instructables.com/id/WiFi-LED-Light-Strip-Controller/>

## 9.5 433mHz

### 9.5.1 Tools

#### Surface Microstrip

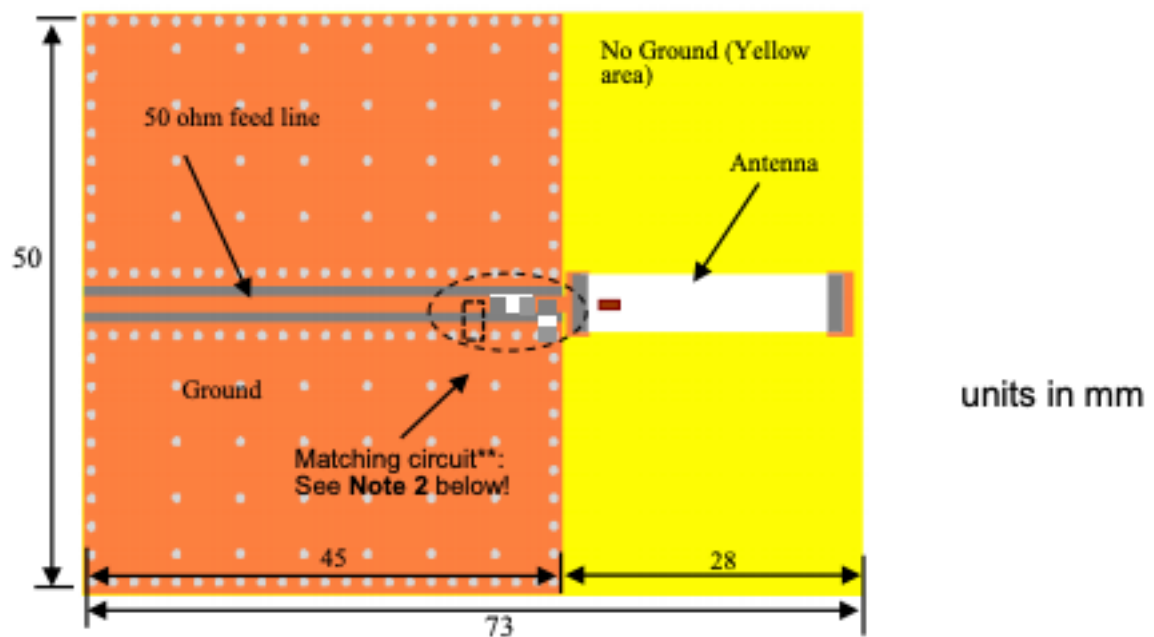


Formula Restrictions:  $0.1 < w/h < 3.0$

w	<input checked="" type="radio"/>	2985,3242	μm	Track width
t Cu	<input type="radio"/>	35	μm	Track height
h	<input type="radio"/>	1600	μm	Isolation height
Er		4,3		Dielectric constant (FR4 - Standard: 4,3)
Z <sub>0</sub>	<input type="radio"/>	50	Ω	Impdedance ca.

**Figure 9.3:** Surface Microstrip

Strip line impedance calculator: <https://www.multi-circuit-boards.eu/en/pcb-design-aid/impedance-calculation.html>



**Figure 9.4:** Feed line

50 ohm impedance feed line: <https://www.disk91.com/2015/technology/hardware/design-a-50ohm-impedance-net-for-rf-signals/>

### 9.5.2 PDF's

- 433 MHz ISM Antenna SMD.pdf

## 9.6 Audio

### 9.6.1 Articles

- [How to build a Microphone Amplifier Circuit](#)

### 9.6.2 Microphone Array

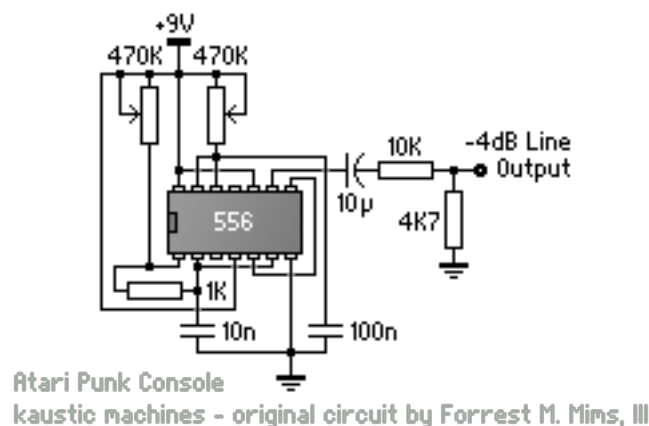
#### 9.6.2.1 Links

- [Quad op-amp LM3900 \(PDF\)](#)
- [Multi-channel audio mixer circuit using LM3900](#)

### 9.6.3 Synthesizers

#### 9.6.3.1 Atari Punk Console

Modification (changed speaker to line output) of the Stepped Tone Generator taken from the “Engineer’s Mini-Notebook - 555 Circuits” by Forrest M. Mims, III (Siliconconcepts, 1984)



Atari Punk Console  
kaustic machines - original circuit by Forrest M. Mims, III

**Figure 9.5:** APC

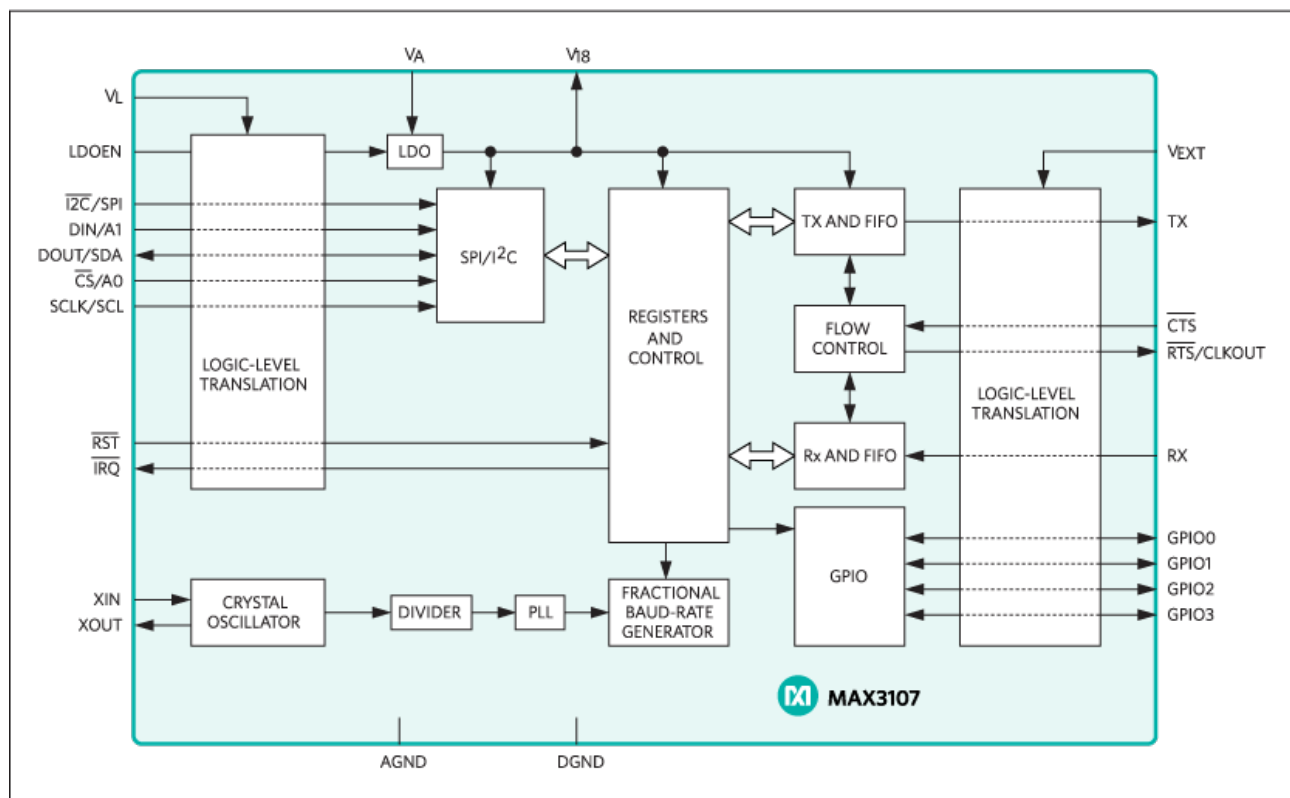
#### Links

- <https://compiler.kaustic.net/machines/apc.html>

## 9.7 MAX3107

The MAX3107 is an advanced universal asynchronous receiver-transmitter (UART) with 128 words each of receive and transmit first-in/first-out (FIFO) that can be controlled through I<sup>2</sup>C or high-speed SPI™. The 2x and 4x rate modes allow a maximum of 24Mbps data rates. A phase-locked loop

(PLL), prescaler, and fractional baud-rate generator allow for high-resolution baud-rate programming and minimize the dependency of baud rate on reference clock frequency.



**Figure 9.6:** MAX3107

## 9.7.1 Datasheets

- [MAX3107.pdf](#)

## 9.8 Espressif

### 9.8.1 ESP 32

#### 9.8.1.1 Encryption

- <https://limitedresults.com/2019/11/pwn-the-esp32-forever-flash-encryption-and-sec-boot-keys-extraction/>

## 9.8.2 ESP 8266

## 9.9 GSM Modules

### 9.9.1 Adafruit Fona 3g cellular breakout

- Quad-band 850MHz GSM, 900MHz EGSM, 1800MHz DCS, 1900MHz PCs - connect onto any global GSM network with any 2G SIM.
- This is the European Version - with dual-band UMTS/HSDPA 900/2100MHz WCDMA + HSDPA
- Fully-integrated GPS (Qualcomm PM8015 GPS) that can be controlled and query over the same serial port
- Make and receive voice calls using a headset or an external 8Ω speaker + electret microphone
- Send and receive SMS messages
- Send and receive GPRS data (TCP/IP, HTTP, etc.)
- AT command interface can be used with 300, 600, 1200, 4800, 9600, 19200, 38400, 57600, 115200, 230K, 461K, 961K, 3.2M, 3.7M and 4.0Mbps
- Native USB support - plug it into a computer and you'll get serial ports for AT commands, GPS NMEA as well as a modem (note we've only tried out the AT&NMEA ports on Windows)

**Cool fact:** The Fona also has a built-in GPS.

**Not-so-cool fact:** The Pi can only make one serial connection. Checkout article *Sending AT commands to SIM900 whilst pppd is active*





**Figure 9.7:** Adafruit fona 3g

#### 9.9.1.1 GPS specifications

- 16 acquisition channels
- GPS L1 C/A code
- Sensitivity
  - Tracking: -157 dBm
  - Cold starts : -144 dBm
- Time-To-First-Fix
  - Cold starts: 100s (typ.)
  - Hot starts: 1s (typ.)
- Accuracy: approx 2.5 meters

#### 9.9.1.2 Implementation

#### 9.9.1.3 Setup

Edit `/boot/config.txt`

```
enable_uart=1
```

## Install dependencies

```
sudo apt-get update
```

```
sudo apt-get install ppp screen
```

## Try it out

```
sudo screen /dev/serial0 115200
```

```
sudo -i
```

```
cd /etc/ppp/peers/
```

```
wget https://raw.githubusercontent.com/adafruit/FONA_PPP/master/fona
```

**Celluar** Open that file to view PPPD settings when “fona” is called. Read more about chat and chatscripts [here](#).

```
vim fona
```

## You should see this:

```
### Example PPPD configuration for FONA GPRS connection on Debian/Ubuntu.

### MUST CHANGE: Change the -T parameter value **** to your network's APN value.
### For example if your APN is 'internet' (without quotes), the line would look like:
### connect "/usr/sbin/chat -v -f /etc/chatscripts/gprs -T internet"
connect "/usr/sbin/chat -v -f /etc/chatscripts/gprs -T telia"

### MUST CHANGE: Uncomment the appropriate serial device for your platform below.
### For Raspberry Pi use /dev/ttyAMA0 by uncommenting the line below:
/dev/ttyUSB0
###/dev/ttyAMA0
### For BeagleBone Black use /dev/ttyO4 by uncommenting the line below:
###/dev/ttyO4

### Speed of the serial line.
115200

### Assumes that your IP address is allocated dynamically by the ISP.
noipdefault

### Try to get the name server addresses from the ISP.
usepeerdns

### Use this connection as the default route to the internet.
defaultroute

### Makes PPPD "dial again" when the connection is lost.
```

```
persist
```

```
### Do not ask the remote to authenticate.
```

```
noauth
```

```
### No hardware flow control on the serial link with FONA
```

```
nocrtscts
```

```
### No modem control lines with FONA.
```

```
local
```

connect to internet:

```
sudo pon fona
```

disconnect

```
sudo poff fona
```

**Realtime clock** Set the realtime clock see [Forum](#).

AT+CLTS? You will get this if it is disabled:

```
+CLTS: 0
```

To enable it enter this:

```
AT+CLTS=1
```

CLTS=1 must be saved in the SIM800's nonvolatile memory so it will be enabled when the module powers up and registers on the network.

Since this setting is not automatically saved in nonvolatile memory, you must save it with:

```
AT&W
```

(This saves all writeable settings)

Now restart your SIM800

After it registers AT+CCLK? will respond with the correct time, as in my case:

```
+CCLK: "14/08/08,02:25:43-16"
```

**Send SMS** Set in sms mode: AT+CMGF=1

Send message

```
AT+CMGS="+4799999999"<ENTER>
```

```
> your message<CTRL+Z>
```

Data sim phone number: +47 580009700018



```
AT+CCLK="02/02/20,23:19:00+01"
```

```
AT+CMGS="+4794835300"
```

```
AT+CMGR=ALL
```

```
AT+CSCA="+47580009700018"
```

```
ATD + +4794835300;
```

```
AT+CMGF=0
```

```
AT+CREG
```

```
AT+CMGF=1
```

```
AT+CHTTPSOPSE="google.com",443,2
```

```
AT+CGDCONT=1,"IP","telia"
```

```
AT+CGDCONT?
```

## Commands log (02.02.2020)

### Important note

I 2019 startet utfasingen av 3G fylke for fylke. Utfasingen startet med 900MHz-båndet, mens 2100MHz fases ut i 2020. En hovedårsak til utfasingen er at 4G-nettet i 2019 var fullt utbygd og kunne overta de fleste funksjonene 3G-nettet hadde, samt at frekvensene til 3G etter en utfasing kan brukes til andre funksjoner. Utstyr uten 4G-mulighet virker etter utfasingen via 2G, slik at telefonfunksjonene virker, mens dataoverføring via mobilnettet ikke lenger er mulig. I følge Telia gikk 4G-hastighetene i Hedmark opp da 3G-nettet ble slukket. [ref. Wikipedia](#)

### 9.9.1.4 Articles

- [Cellular & GPS Enabled Pi 3: Fona + Pi 3](#)
- [FONA Tethering to Raspberry Pi or BeagleBone Black](#)
- [Sending AT commands to SIM900 whilst pppd is active](#)

### 9.9.2 Sierra Wireless WP7607-1

Reliable Cat-1 LTE connectivity for 4G networks with 3G and 2G fallback in EMEA The WP7607-1 module is part of the WP Series offering a secure device-to-cloud architecture enabling IoT developers to build a Linux-based product on a single module – reducing overall system complexity and time-to-market.

This LTE Cat-1 wireless module is ideal for low-bandwidth IoT applications that require real-time communication along with the longevity and network quality that 4G LTE networks bring.

Delivering up to 10Mbps download speed, many organizations are using Cat-1 networks as a stepping stone to low-power wide-area (LPWA) networks as 4G LTE can provide a superior user experience over 2G and 3G technologies.



**Figure 9.8:** WP Series

#### 9.9.2.1 Links

- [Digikey - WP7607-1](#)
- [Sierra Wireless - Embedded solutions](#)
- [AirPrime WP7607-1 IoT Module](#)

## 9.10 Modbus

Tutorial: <https://www.renesas.com/eu/en/www/doc/whitepapers/interface/rs-485-transceiver-tutorial.pdf>

Chip brukt i kontroller: SN65HVD485E Half-Duplex RS-485 Transceiver (<http://www.ti.com/lit/ds/symlink/sn65hvd485e.pdf>)

RS-485 til UART <https://www.sparkfun.com/products/10124>

Anbefalt modbus usb driver: <https://www.sparkfun.com/products/9822>

Datasheets: <https://www.sparkfun.com/datasheets/BreakoutBoards/USB-to-RS485-Breakout-v11.pdf>

For “end of line” motstand kjøp både 120 og 220ohm

# 10 Hardware

## 10.1 Oculus Rift

### 10.1.1 Oculus Rift + Steam VR + Linux

Article: <https://steamcommunity.com/app/221410/discussions/0/1696046342850651740/>

### 10.1.2 Links

- [OpenHMD](#)

## 10.2 SDR

### 10.2.1 LimeSDR

- [LimeSDR Mini](#)

### 10.2.2 SDRPlay

- [SDRPlay](#)

# 11 Machine learning

## 11.1 Articles

- [Contrastive Self-Supervised Learning](#)
- [Dopamine and temporal difference learning: A fruitful relationship between neuroscience and AI](#)

## 11.2 Chat bots

### 11.2.1 Alternatives

- [Wit.ai](#)
- [Botpress](#)
- [Botkit](#)
- [Dialogflow](#)

### 11.2.2 On premise

- [ai-chatbot-framework](#)

## 11.3 Chaos

- 
- [Chaos visual example](#)

# 12 Physics

## 12.1 Articles

- [Gears](#)

# 13 Productivity

## 13.1 Articles

- [Remote software developer productivity](#)

# 14 Programming

- [Rust](#)

## 14.1 Sorting

- [Flash sort algorithm](#)

## 14.2 Hexagonal architecture

The **hexagonal architecture**, or **ports and adapters architecture**, is an architectural pattern used in software design. It aims at creating loosely coupled application components that can be easily connected to their software environment by means of ports and adapters. This makes components exchangeable at any level and facilitates test automation.

[Read more](#)

## 14.3 Frameworks

### 14.3.1 Firebase

#### 14.3.1.1 Alternatives

- [Sapphire](#)

### 14.3.2 Sapphire

Open source alternative to firebase <https://sapphire-db.com/start/main>

### 14.3.3 Spring boot

#### 14.3.3.1 Quarkus

[The JHipster Quarkus demo app](#)

- <https://quarkus.io/>

## 14.4 Programming languages

### 14.4.1 C++

#### 14.4.1.1 Articles

- [Datastructure APIs in C++](#)

#### 14.4.1.2 Videos

- [Bjarne Stroustrup – The Essence of C++ \(2014\)](#)

### 14.4.2 Golang

#### 14.4.2.1 Tips and tricks

```
go mod init  
go mod vendor
```

#### Vendors

#### 14.4.2.2 Links

#### Web framework

- [Gin gonic](#)

#### ORM

- <http://gorm.io/>

#### GUI

- <https://hackernoon.com/how-to-add-a-gui-to-your-golang-app-in-5-easy-steps-c25c99d4d8e0>
- <https://github.com/andlabs/ui>
- <https://github.com/therecipe/qt>

#### Web

- [GraphQL Schema Language Cheat Sheet](#)
- [Learn Golang + GraphQL + Relay #1](#)
- [Learn Golang + GraphQL + Relay #2](#)
- <https://github.com/mingrammer/go-web-framework-stars>



## Div

- [Awesome Go](#) - A curated list of awesome Go frameworks, libraries and software.

### 14.4.2.3 Articles

- [Example of Golang CRUD using MySQL from scratch](#)
- [Real-Time Maps with a Raspberry Pi, Golang, and HERE XYZ](#)
- [Reverse Geocoding NEO 6M GPS Positions with Golang and a Serial UART Connection](#)

### 14.4.2.4 Serial UART

- [Go Serial](#)
- [Tarm Serial](#)

## 14.4.3 Hammerspoon

This is a tool for powerful automation of OS X. At its core, Hammerspoon is just a bridge between the operating system and a Lua scripting engine. What gives Hammerspoon its power is a set of extensions that expose specific pieces of system functionality, to the user.

<https://www.hammerspoon.org/>

## 14.4.4 Rust

### 14.4.4.1 Links

#### Web

- <https://rocket.rs/>

#### GUI

- <https://github.com/PistonDevelopers/conrod>
- <http://relm.ml/relm-intro>

#### ORM

- <http://diesel.rs/guides/getting-started/>

## ESP32

- <https://mabez.dev/blog/posts/esp32-rust/>

# 15 Projects

- [Crudus MD Notes](#)
- [Maximus](#)

## 15.1 Inspiration aka others awesome projects

- [Rotary Cellphone](#)

## 15.2 Status Light

### 15.2.1 iOS app

- <https://stackoverflow.com/questions/23535355/how-to-detect-call-incoming-programmatically>
- <https://www.raywenderlich.com/150015/callkit-tutorial-ios>

## 15.3 BMW Media Center

- BMW Connected Apps Protocol <https://hufman.github.io/stories/bmwconnectedapps>
- [Shopping list](#)

### 15.3.1 Articles

- <https://hackaday.io/project/161745-can-bus-hacker>
- <https://hackaday.com/2019/05/09/sniffing-can-to-add-new-features-to-a-modern-car/>

### 15.3.2 Shoppinglist for BMW

#### 15.3.2.1 Bmw controller

- <https://www.cubietruck.com/products/cubieboard4-cc-a80-high-performance-mini-pc-development-board>
- <https://www.96boards.org/product/hikey960/>

## 15.4 Crudus Markdown Notes

En markdown applikasjon som kan synkronisere med git.

### 15.4.1 Platform

#### 15.4.1.1 iOS / Android

- Nativescript
- <https://libgit2.org/>
- <https://github.com/libgit2/objective-git>
- <https://github.com/Raekye/ObjectiveGit-iOS-Example>

#### 15.4.1.2 Desktop

- Electron

### 15.4.2 Links

- <https://libgit2.org/>
- <https://cocoapods.org/pods/libgit2>
- <https://github.com/libgit2/libgit2#android>

### 15.4.3 Libraries

#### 15.4.3.1 JavaScript

- [Marked](#)
- [Remarkable](#)
- [PageDown](#) (and [PageDown Extra](#))
- [markdown-it](#)
- [Gitdown](#): GitHub markdown preprocessor
- [reMarked.js](#): HTML-to-Markdown processor
- [Kramed](#): Fork of Marked

#### 15.4.4 Other Editors

- [StackEdit](#): In-browser MD document editor
- [Minimalist Online Markdown Editor](#)
- [Mou](#): macOS editor
- [Haroopad](#): Cross-platform editor

## 15.5 Crudus Photos

### 15.5.1 Tensor flow

**A person on a beach flying a kite.**



**A black and white photo of a train on a train track.**



**A person skiing down a snow covered slope.**



**A group of giraffe standing next to each other.**



Image to text

<https://github.com/tensorflow/models/tree/master/research/im2txt>

### 15.5.2 Articles

Building a private, local photo search app using machine learning <https://towardsdatascience.com/building-a-private-local-photo-search-app-using-machine-learning-8aeeef8d245c>

A step by step guide to Caffe <http://shengshuyang.github.io/A-step-by-step-guide-to-Caffe.html>

### 15.5.3 Photo History

Histogram in photography <https://www.phototraces.com/photography-basics/histogram-in-photography/>

Histogram basics [https://docs.opencv.org/3.1.0/d1/db7/tutorial\\_py\\_histogram\\_begins.html](https://docs.opencv.org/3.1.0/d1/db7/tutorial_py_histogram_begins.html)

#### 15.5.4 Tools

- [Tagbox](#)
- [NVIDIA docker support](#)

##### 15.5.4.1 Ubuntu

```
sudo apt install exiftran libjpeg-turbo-progs
```

#### 15.5.5 Links

##### 15.5.5.1 Caffe

- <https://caffe.berkeleyvision.org/>

##### 15.5.5.2 Model zoo

- <https://github.com/BVLC/caffe/wiki/Model-Zoo>

##### 15.5.5.3 Docker image

- <https://github.com/BVLC/caffe/tree/master/docker>

##### 15.5.5.4 Diff image

- <https://stackoverflow.com/questions/5132749/diff-an-image-using-imagemagick>

##### 15.5.5.5 Image Fingerprint

- <https://realpython.com/fingerprinting-images-for-near-duplicate-detection/>

##### 15.5.5.6 Frame Hash

- [https://github.com/sschnug/pyVideoHash/blob/master/frame\\_hash.pyx](https://github.com/sschnug/pyVideoHash/blob/master/frame_hash.pyx)

##### 15.5.5.7 Image recognition

- <https://www.learnopencv.com/image-recognition-and-object-detection-part1/>

#### 15.5.5.8 Duplicate images

- <https://github.com/philipbl/duplicate-images>
- <https://blog.iconfinder.com/detecting-duplicate-images-using-python-cb240b05a3b6>
- <https://www.youtube.com/watch?v=AlyJSGmkFXk>

#### 15.5.5.9 OpenCV Line detection

- <https://www.codepool.biz/opencv-line-detection.html>
- [https://docs.opencv.org/3.4/dd/dd7/tutorial\\_morph\\_lines\\_detection.html](https://docs.opencv.org/3.4/dd/dd7/tutorial_morph_lines_detection.html)

#### 15.5.5.10 Detect horizon

- <https://stackoverflow.com/questions/4705837/horizon-detection-algorithm>

#### 15.5.5.11 OpenCV Auto-level / histogram

- <https://docs.opencv.org/2.4/modules/imgproc/doc/histograms.html?highlight=equalizehist#cv2.equalizeHist>

#### 15.5.5.12 OpenCV rotate images

- <https://www.pyimagesearch.com/2017/01/02/rotate-images-correctly-with-opencv-and-python/>

#### 15.5.5.13 MIT Deep learning

- <https://github.com/lexfridman/mit-deep-learning>

#### 15.5.5.14 Tensorflow and docker

- <https://www.sicara.ai/blog/2017-11-28-set-tensorflow-docker-gpu>
- <https://stackoverflow.com/questions/47068709/your-cpu-supports-instructions-that-this-tensorflow-binary-was-not-compiled-to-u>
- <https://github.com/lakshayg/tensorflow-build>

#### 15.5.5.15 OpenCV 4

<https://www.pyimagesearch.com/2018/08/17/install-opencv-4-on-macos/>

## 15.6 Crudus Sense

### 15.6.1 BLE device configuration specification

Name	Type	R/WKey	UUID	Comment
Device name	String	R/WdeviceName	5759f8cc-69ee-11e9-8a12-1681be663d3e	
WiFi Mac	String	R	51ecb1ca-6b85-11e9-a923-1681be663d3e	Read from ESP and register device to Crudus sense backend
WiFi SSID	String	R/Wwifi-ssid	51ecb440-6b85-11e9-a923-1681be663d3e	
WiFi passwd	String	W wifi-pwd	51ecb594-6b85-11e9-a923-1681be663d3e	
Room	String	R/Wloc-room	51ecb6ca-6b85-11e9-a923-1681be663d3e	
Floor	Integer?	R/Wloc-floor	51ecb7f6-6b85-11e9-a923-1681be663d3e	
Compound	String	R/WLoc-comp	51ecb922-6b85-11e9-a923-1681be663d3e	
MQTT topic	String	R/Wmqtt-topic	51ecba4e-6b85-11e9-a923-1681be663d3e	Only if MQTT-host is changed
MQTT host	String	R/Wmqtt-host	51ecbf26-6b85-11e9-a923-1681be663d3e	Disables default mqtt host
MQTT port	Integer	R/Wmqtt-port	51ecc156-6b85-11e9-a923-1681be663d3e	Only if MQTT-host is changed
MQTT username	String	R/Wmqtt-user	51ecc2c8-6b85-11e9-a923-1681be663d3e	Only if MQTT-host is changed
MQTT password	String	W mqtt-pwd	51ecc3fe-6b85-11e9-a923-1681be663d3e	Only if MQTT-host is changed

Name	Type	R/WKey	UUID	Comment
Crudus Accounts username	String	W crudus-	51ecc52a-6b85-11e9-a923-1681be663d3e	For setting default MQTT topic and MQTT username
Crudus Accounts token	String	W crudus-	51ecc6d8-6b85-11e9-a923-1681be663d3e	For OTA downloads and MQTT password / token
Calibration temperature	String (comma separated)	R/Wcali-temp	51ecca5c-6b85-11e9-a923-1681be663d3e	For calibrate temperature
Calibration humidity	String (comma separated)	R/Wcali-hum	51eccbb0-6b85-11e9-a923-1681be663d3e	For calibrate humidity
Soft reset	boolean	W soft-reset	51eccd18-6b85-11e9-a923-1681be663d3e	For clearing preferences

### 15.6.2 MQTT publish Topics

Topic	Payload	Comment

### 15.6.3 MQTT Subscribe Topics

Topic	Payload	Action	Comment
/sense/ota		Calls OTA for update	

### 15.6.4 Extensions

Sleep Tracking using an Arduino <https://duino4projects.com/sleep-tracking-using-an-arduino/>

Reset: <https://www.esp8266.com/viewtopic.php?t=9558&start=8>

Chip: CCS811 (indoor air quality sensor)



## 15.7 Kaldheim.org

### 15.7.1 Links

- <https://themes.getbootstrap.com/product/milo-magazineblog-theme/>

## 15.8 Maximus

### 15.8.1 Components

- [Stereo Pi](#)
- [Configure BNO055](#)

### 15.8.2 Robotics

- [Robotics](#)

### 15.8.3 Articles

- [Comparing Gyroscope Datasheets](#)

### 15.8.4 Artificial Intelligence

- [AI Notes](#)

### 15.8.5 BNO055

#### 15.8.5.1 Installation

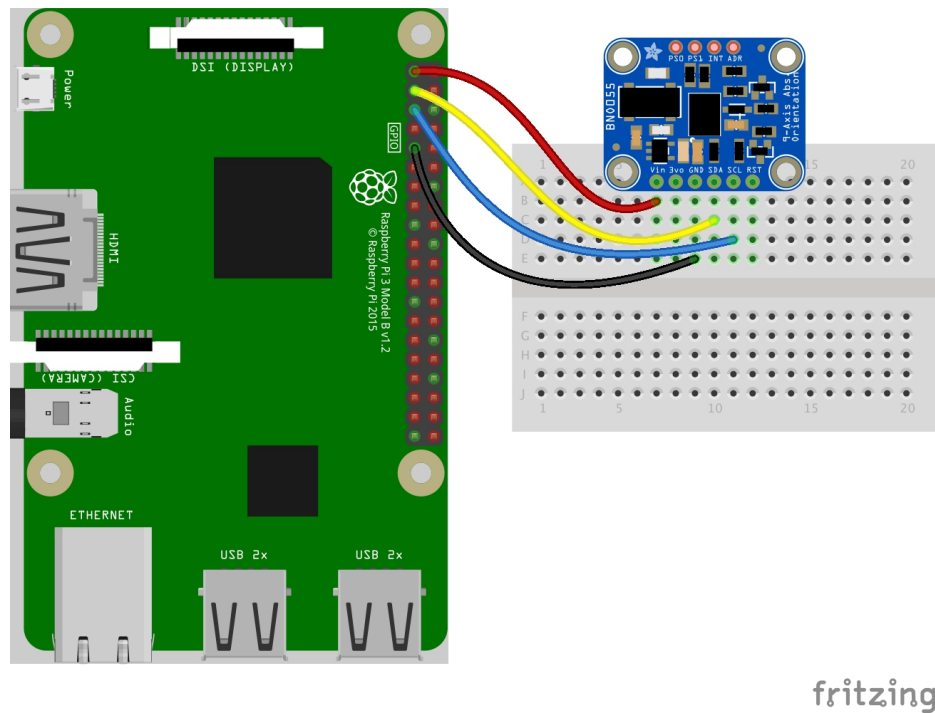
- [BNO055 - Python & CircuitPython](#)

```
pip3 install RPI.GPIO  
pip3 install adafruit-blinka
```

#### i2c configuration

- [I2C Clock Stretching](#)

In order to use certain I2C sensors, such as the BNO055, you'll need to enable I2C clock stretching 'support' by greatly slowing down the I2C clock on the Raspberry Pi using the device tree overlay.



**Figure 15.1:** Raspberry pi Configuration

Edit `/boot/config.txt`

```
### Uncomment some of all of these to enable the optional hardware interfaces
dtparam=i2c_arm=on
dtparam=i2s=on
dtparam=spi=on

### Clock stretching by slowing down to 10KHz
dtparam=i2c_arm_baudrate=10000
```

Reboot the device

```
sudo reboot
```

Check for i2c devices:

```
$ i2cdetect -y 1
   0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  --  28  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
50:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
70:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
```

```
mkdir Maximus && cd Maximus
python3 -m venv .env
source .env/bin/activate
pip3 install adafruit-circuitpython-bno055
```

**Create new project**   Example data from sensor:

```
Temperature: 28 degrees C
Accelerometer (m/s^2): (-0.2, -0.07, -9.77)
Magnetometer (microteslas): (-27.75, -4.0625, 32.5)
Gyroscope (rad/sec): (-0.001090830782496456, -0.004363323129985824, 0.0)
Euler angle: (None, None, None)
Quaternion: (0.011474609375, -0.3623046875, 0.9320068359375, 0.0)
Linear acceleration (m/s^2): (1.28, 0.0, -0.01)
Gravity (m/s^2): (-0.21, -0.08, -9.8)
```

### 15.8.5.2 PID controller

- [Arduino BNO055 PID Gyro sensor](#)
- [PID Control for multiple linear actuators](#)

### 15.8.5.3 Videos

- [How to Implement an Inertial Measurement Unit \(IMU\) Using an Accelerometer, Gyro, and Magnetometer](#)
- [How to Merge Accelerometer with GPS to Accurately Predict Position and Velocity](#)

### 15.8.5.4 Links

- [Adafruit BNO055](#)
- [Adafruit BNO055 absolute orientation sensor](#)

### 15.8.5.5 Documents

- [An introduction and tutorial for PID controllers \(PDF\)](#)

### 15.8.5.6 Books

- [Technician's Guide to Programmable Controllers](#)
- [PID Controllers: Theory, Design, and Tuning](#)
- [PID Control Fundamentals](#)
- [Model-Reference Robust Tuning of PID Controllers \(Advances in Industrial Control\)](#)
- [HANDBOOK OF PI AND PID CONTROLLER TUNING RULES \(3RD EDITION\)](#)

## 15.8.6 Development board

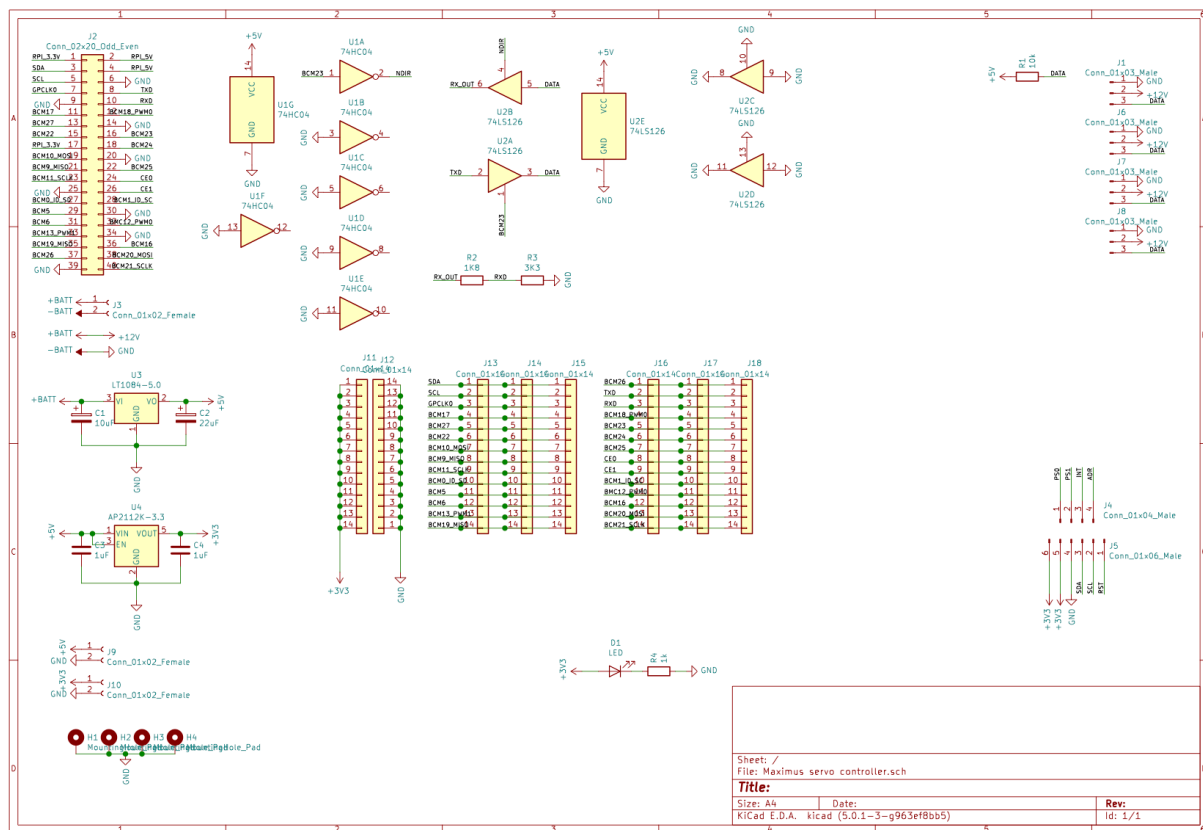


Figure 15.2: Schematics

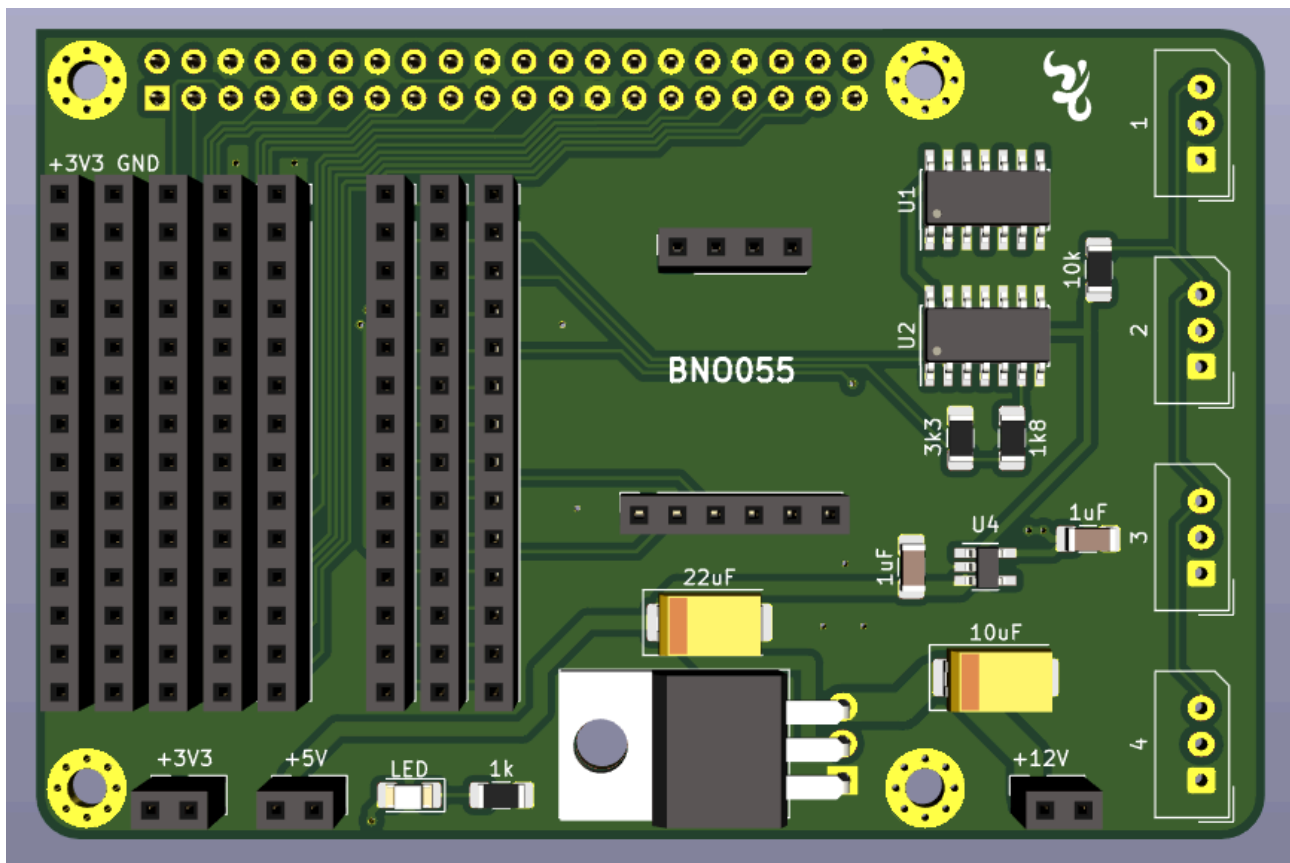


Figure 15.3: Dev board 3d

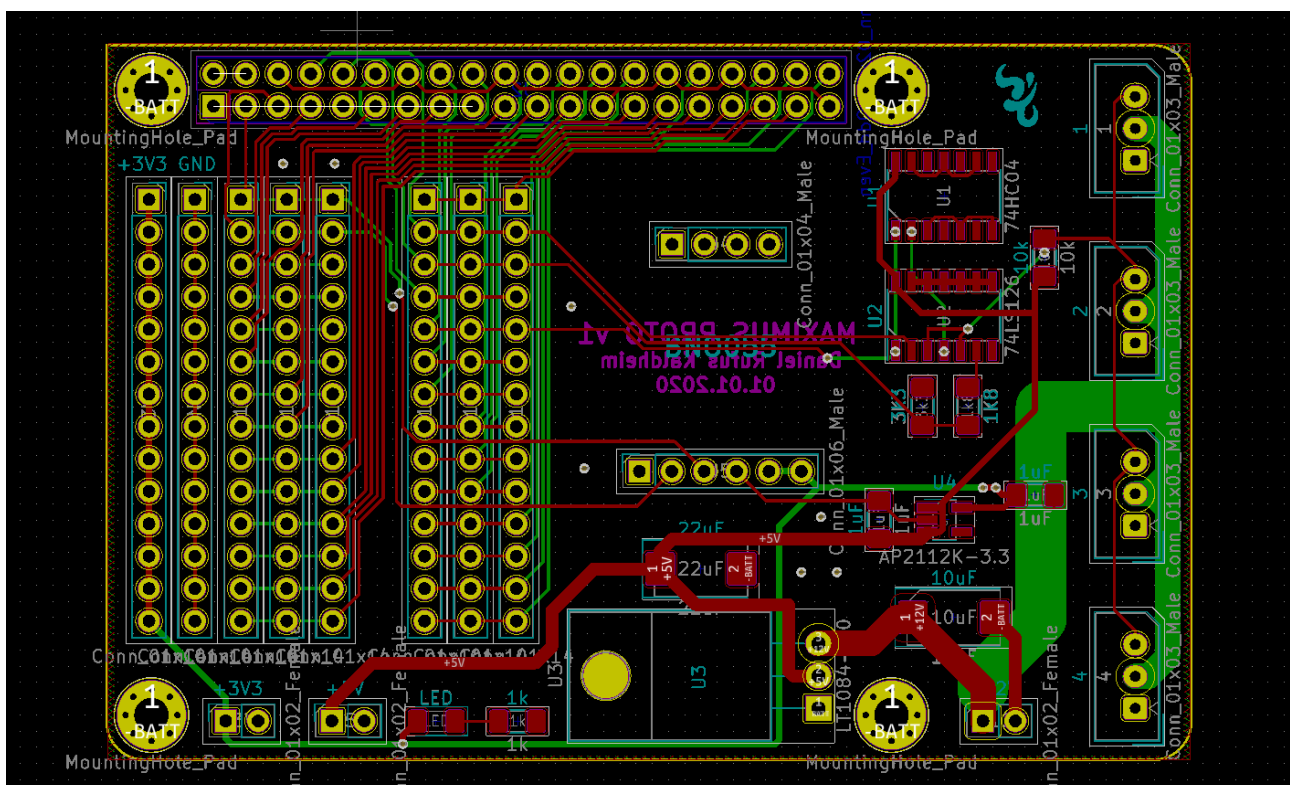


Figure 15.4: Dev board pcb

### 15.8.6.1 Raspberry Pi pinout

Raspberry Pi GPIO BCM numbering

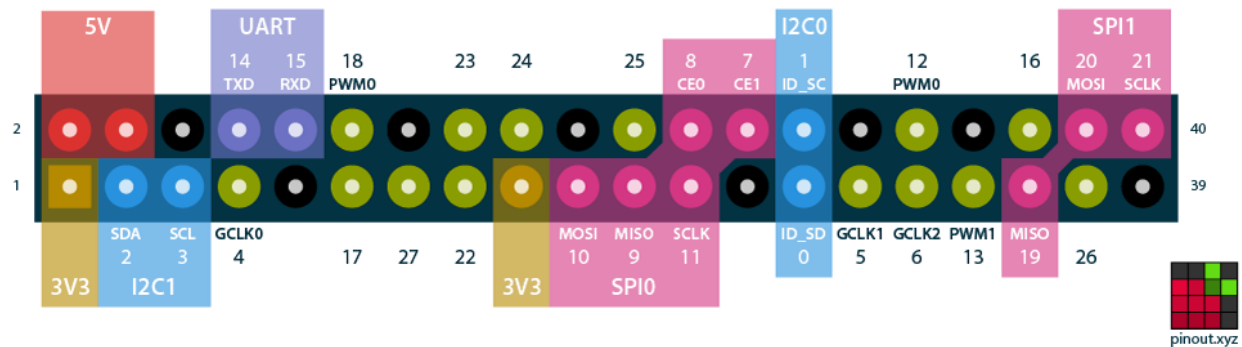


Figure 15.5: Pinout

### 15.8.7 Dynamixel AX-12A

DYNAMIXEL is a robot exclusive smart actuator with fully integrated DC Motor + Reduction Gearhead + Controller + Driver + Network in one DC servo module.


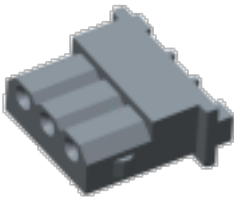


Figure 15.6: Dynamixel AX-12A

### 15.8.7.1 Specification

Item	Specification
Baud Rate	7843 bps ~ 1 Mbps
Resolution	0.29 [°]
Running Degree	0 [°] ~ 300 [°] Endless Turn
Weight	53.5g(AX-12, AX-12+), 54.6g(AX-12A)
Dimensions (W x H x D)	32mm x 50mm x 40mm
Gear Ratio	254 : 1
Stall Torque	1.5 N*m (at 12V, 1.5A)
No Load Speed	59rpm (at 12V)
Operating Temperature	-5 [°C] ~ +70 [°C]
Input Voltage	9.0 ~ 12.0V ( <b>Recommended</b> : 11.1V)
Command Signal	Digital Packet
Protocol Type	Half Duplex Asynchronous Serial Communication (8bit, 1stop, No Parity)
Physical Connection	TTL Level Multi Drop Bus
ID	0 ~ 253
Feedback	Position, Temperature, Load, Input Voltage, etc
Material	Engineering Plastic

### 15.8.7.2 Wiring

Item	TTL
Pinout	1 GND2 VDD3 DATA
Diagram	
Housing	 <a href="#">MOLEX 50-37-5033</a>

Item	TTL
	
PCB Header	<a href="#">MOLEX 22-03-5035</a>
Crimp Terminal	<a href="#">MOLEX 08-70-1039</a>
Wire Gauge	21 AWG

### 15.8.7.3 TTL communications

To control the DYNAMIXEL actuators, the main controller needs to convert its UART signals to the half duplex type.

The recommended circuit diagram for this is shown below.

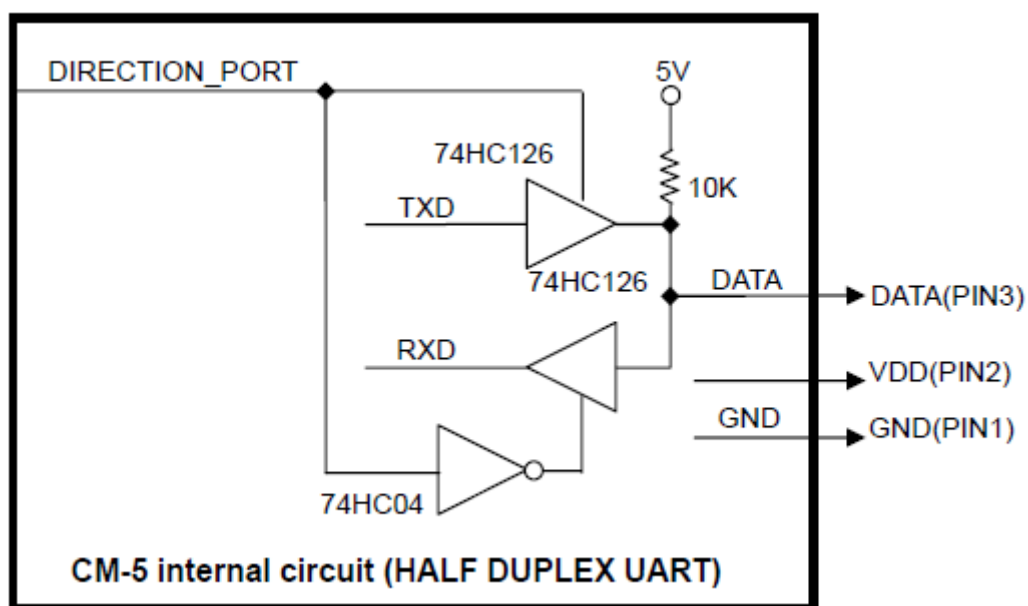
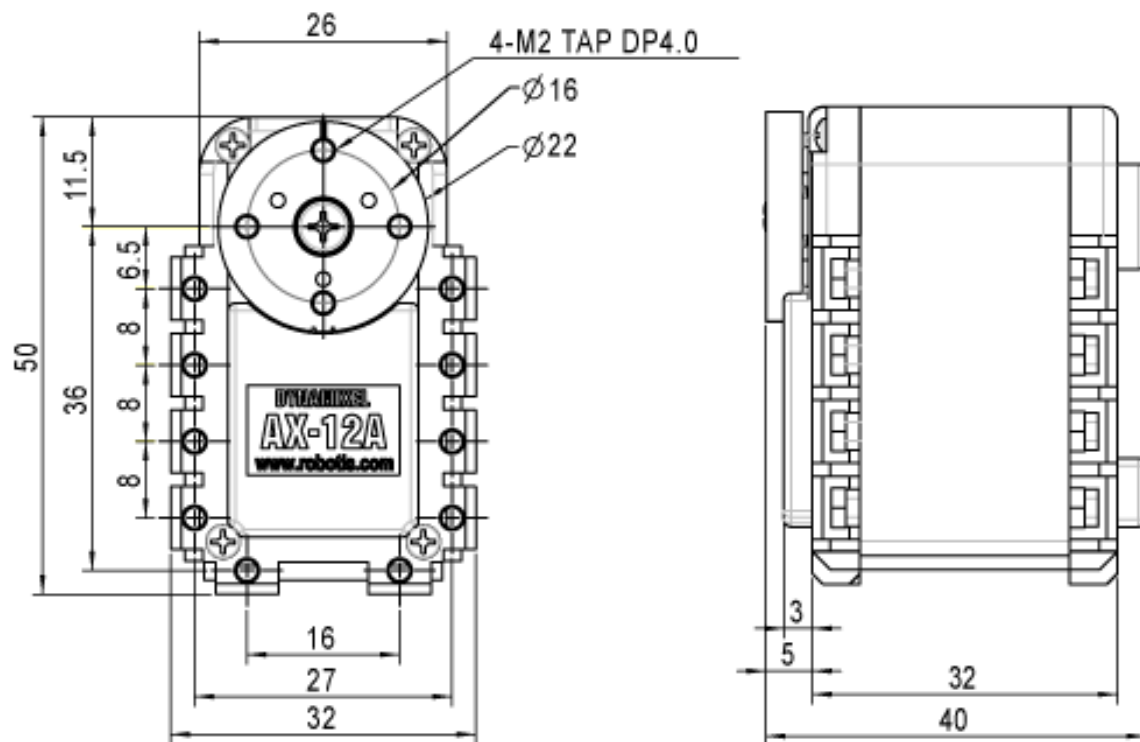


Figure 15.7: TTL



### 15.8.7.4 Drawings



**Figure 15.8:** Drawings

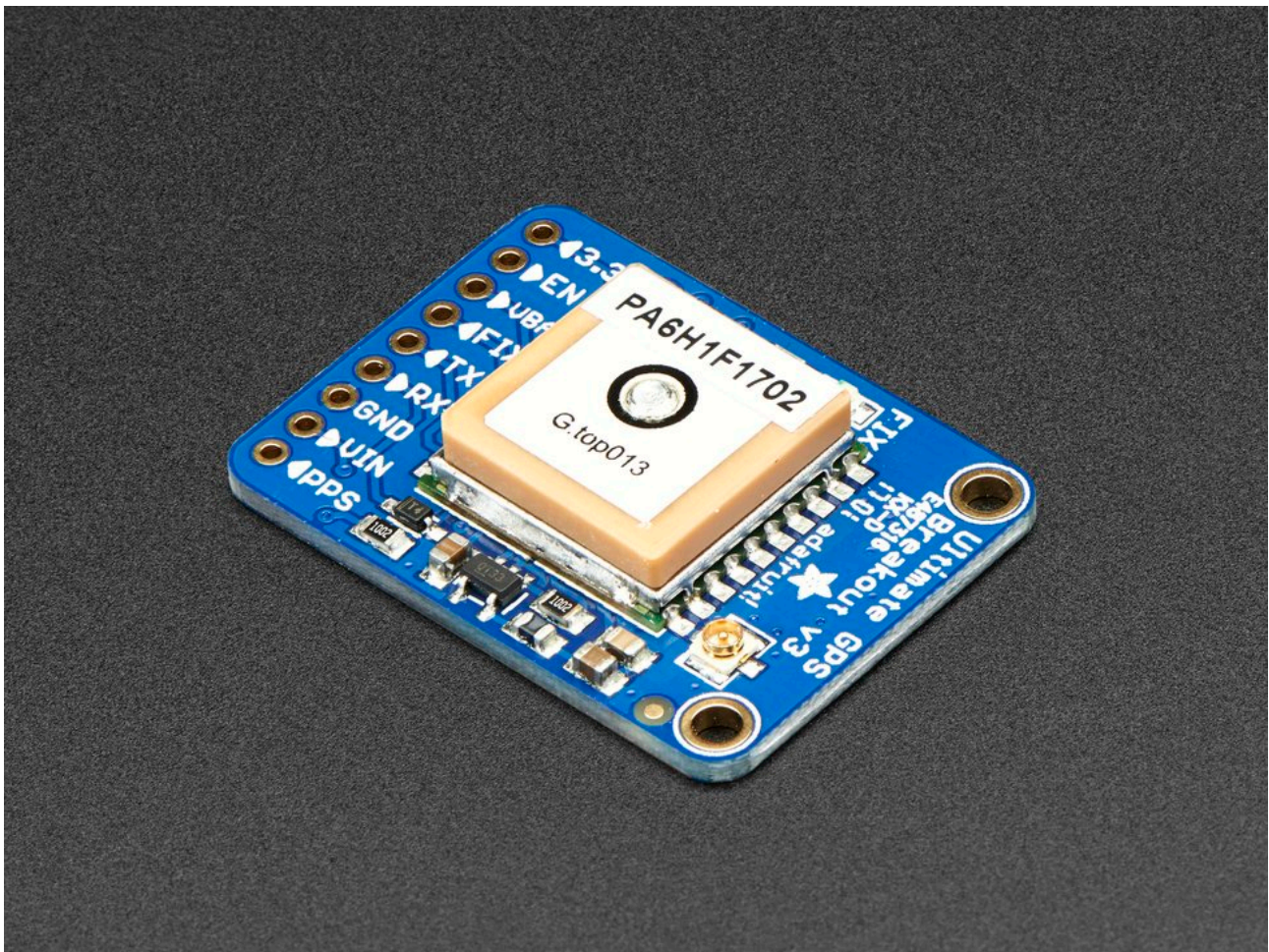
### 15.8.7.5 Links

- [E-manual](#)
- [Dynamixel SDK](#)

## 15.8.8 GPS

### 15.8.8.1 Adafruit ultimate GPS breakout

- -165 dBm sensitivity, 10 Hz updates, 66 channels
- 5V friendly design and only 20mA current draw
- Breadboard friendly + two mounting holes
- RTC battery-compatible
- Built-in datalogging
- PPS output on fix
- Internal patch antenna + u.FL connector for external active antenna
- Fix status LED



**Figure 15.9:** Adafruit ultimate GPS breakout

#### Technical details

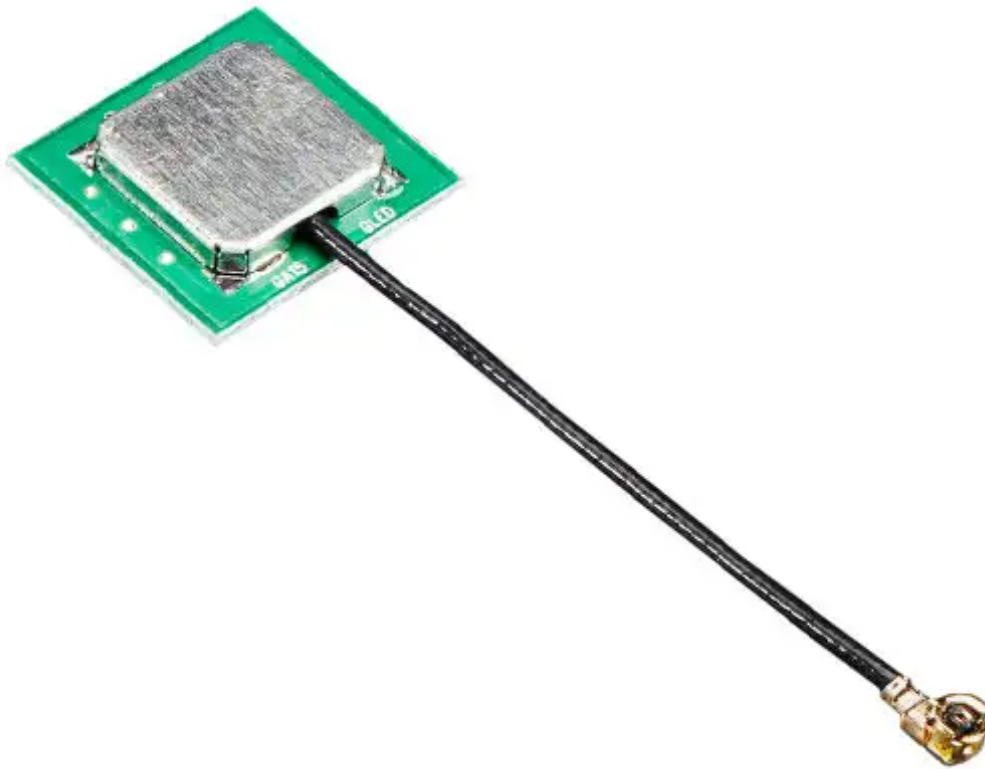
- Satellites: 22 tracking, 66 searching
- Patch Antenna Size: 15mm x 15mm x 4mm
- Update rate: 1 to 10 Hz
- Position Accuracy: < 3 meters (all GPS technology has about 3m accuracy)
- Velocity Accuracy: 0.1 meters/s
- Warm/cold start: 34 seconds
- Acquisition sensitivity: -145 dBm
- Tracking sensitivity: -165 dBm
- Maximum Velocity: 515m/s
- Vin range: 3.0-5.5VDC
- MTK3339 Operating current: 25mA tracking, 20 mA current draw during navigation
- Output: NMEA 0183, 9600 baud default, 3V logic level out, 5V-safe input
- DGPS/WAAS/EGNOS supported
- FCC E911 compliance and AGPS support (Offline mode : EPO valid up to 14 days )
- Up to 210 PRN channels
- Jammer detection and reduction
- Multi-path detection and compensation

**Links**

- [Adafruit product page](#)
- [Adafruit Overview](#)

**15.8.8.2 GPS antenna**

Recommended GPS antenna



**Figure 15.10:** GPS Antenna

**Links**

- [RF ANT 1.575GHZ CER PATCH CAB](#)

## Datasheets

- **Passive GPS Antenna uFL - 15mm x 15mm 1 dBi gain**

### 15.8.9 Hardware

#### 15.8.9.1 Jetson Nano

NVIDIA Jetson Nano enables the development of millions of new small, low-power AI systems. It opens new worlds of embedded IoT applications, including entry-level Network Video Recorders (NVRs), home robots, and intelligent gateways with full analytics capabilities.



**Figure 15.11:** Jetson nano

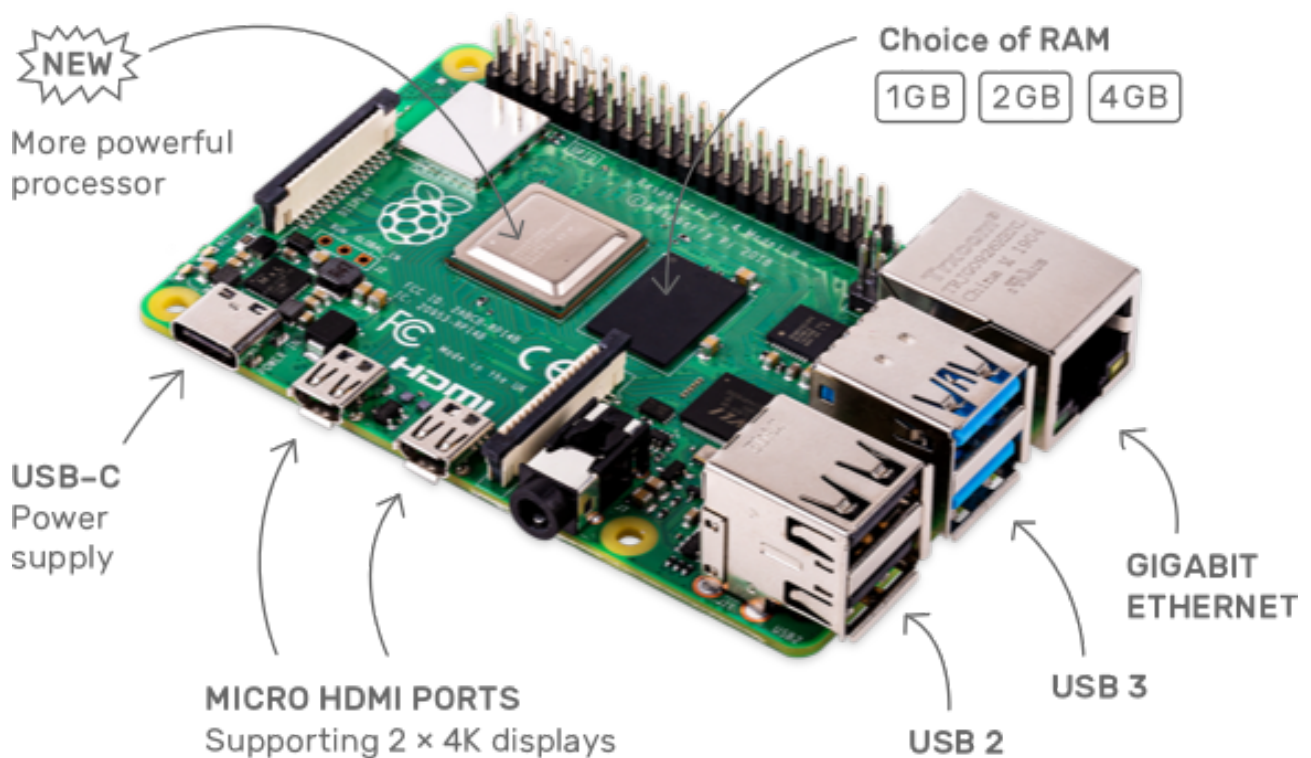
Links:

- <https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-nano/>

#### 15.8.9.2 Raspberry Pi 4 - Model B

The speed and performance of the new Raspberry Pi 4 is a step up from earlier models. For the first time, we've built a complete desktop experience. Whether you're editing documents, browsing the web with a bunch of tabs open, juggling spreadsheets or drafting a presentation, you'll find the experience smooth and very recognisable — but on a smaller, more energy-efficient and much more cost-effective machine.





**Figure 15.12:** Raspberry Pi 4

Links:

- [Raspberry Pi 4](#)

### 15.8.10 Maximus AI

#### 15.8.10.1 For termial conversations

- <http://www.methods.co.nz/asciidoc/>

#### 15.8.10.2 AIML

- <http://www.alicebot.org/aiml.html>
- <https://www.tutorialspoint.com/aiml/>
- <http://www.devdungeon.com/content/ai-chat-bot-python-aiml>
- <https://github.com/pandorabots/rosie/tree/master/lib/aiml>

#### 15.8.10.3 Unicode

hex: “\xf0\x9f\x90\xb6”

#### 15.8.10.4 Artificial Intelligence

- <http://blog.hackerearth.com/2015/12/artificial-intelligence-101-how-to-get-started.html>

#### 15.8.10.5 Words, spelling and so on

- <https://market.mashape.com/wordsapi/wordsapi>
- <https://github.com/montanaflynn/Spellcheck-API/>
- <https://market.mashape.com/sentity/sentity-text-analytics>
- <https://market.mashape.com/aylien/text-analysis>
- <https://market.mashape.com/textanalysis/text-summarization>
- <https://www.meaningcloud.com/developer/>
- <https://market.mashape.com/faceplusplus/faceplusplus-face-detection>
- <http://developers.answers.com/>

#### 15.8.10.6 Grammar

- <https://learnenglish.britishcouncil.org/en/>
- <https://github.com/markfullmer/grammark/tree/Version-3>
- <https://github.com/languagetool-org/languagetool> (<http://wiki.languagetool.org/public-http-api>)

#### 15.8.10.7 NLP / NER

- Part-of-speech tagging (POS)
- Chunking (CHK)
- Name entity recognition (NER)
- Info: <http://nlp.stanford.edu/software/CRF-NER.shtml>
- Download: <http://nlp.stanford.edu/software/stanford-ner-2016-10-31.zip>
- <https://github.com/agentile/PHP-Stanford-NLP> (old) use patrickschur
- <https://packagist.org/packages/patrickschur/stanford-nlp-tagger>
- <http://php-nlp-tools.com/>

#### 15.8.10.8 Intent parser

- <https://github.com/MycroftAI/adapt>

#### 15.8.10.9 Object recognition (caffe)

- <http://tutorial.caffe.berkeleyvision.org/caffe-cvpr15-detection.pdf>

#### 15.8.10.10 Image analyze

- <https://github.com/Samshal/PHP-Photo-Information>
- <http://caffe.berkeleyvision.org/>

#### 15.8.10.11 Automatic speech recognition

- <http://cmusphinx.sourceforge.net/>
- <http://kaldi-asr.org/>

#### 15.8.10.12 Questions / answers

- <https://github.com/TScottJ/OpenEphyra>
- <https://cs.umd.edu/~miyyer/qblearn/>
- <https://github.com/brmson/yodaqa>

#### 15.8.10.13 Lucida

- <http://lucida.ai/media/hpca-lucida-djinn-tutorial.pdf>

#### 15.8.10.14 Animations

- [https://www.youtube.com/watch?v=\\_WlqMqXpyxA](https://www.youtube.com/watch?v=_WlqMqXpyxA)

#### 15.8.10.15 OCR / Deep learning

- <https://blogs.dropbox.com/tech/2017/04/creating-a-modern-ocr-pipeline-using-computer-vision-and-deep-learning/>

#### 15.8.10.16 Neural network (arduino)

- <http://robotics.hobbizine.com/arduinoann.html>

#### 15.8.10.17 Other Links

- <https://github.com/GokuMohandas/practicalAI>
- <http://www.aicheatsheets.com/>

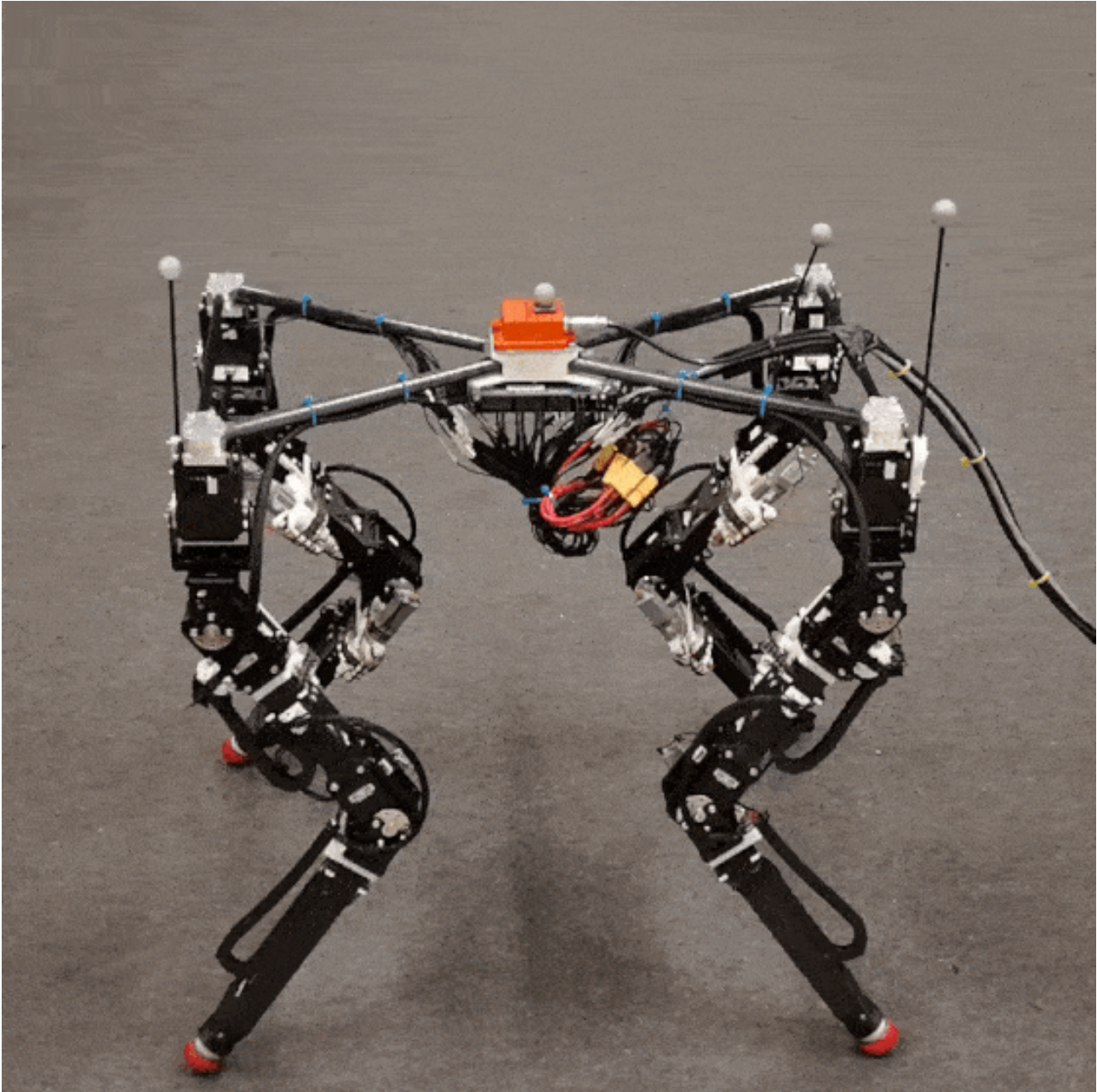
### 15.8.11 Inspiration

#### 15.8.11.1 Read

- [PyRobot](#)

### 15.8.11.2 DyRET Robot

DyRET is a four legged (quadruped) robot designed to be a robust and versatile platform for evolutionary experiments with the unique capability of self-changing morphology. The robot runs on ROS, and uses Gazebo for simulated experiments.



**Figure 15.13:** DyRET

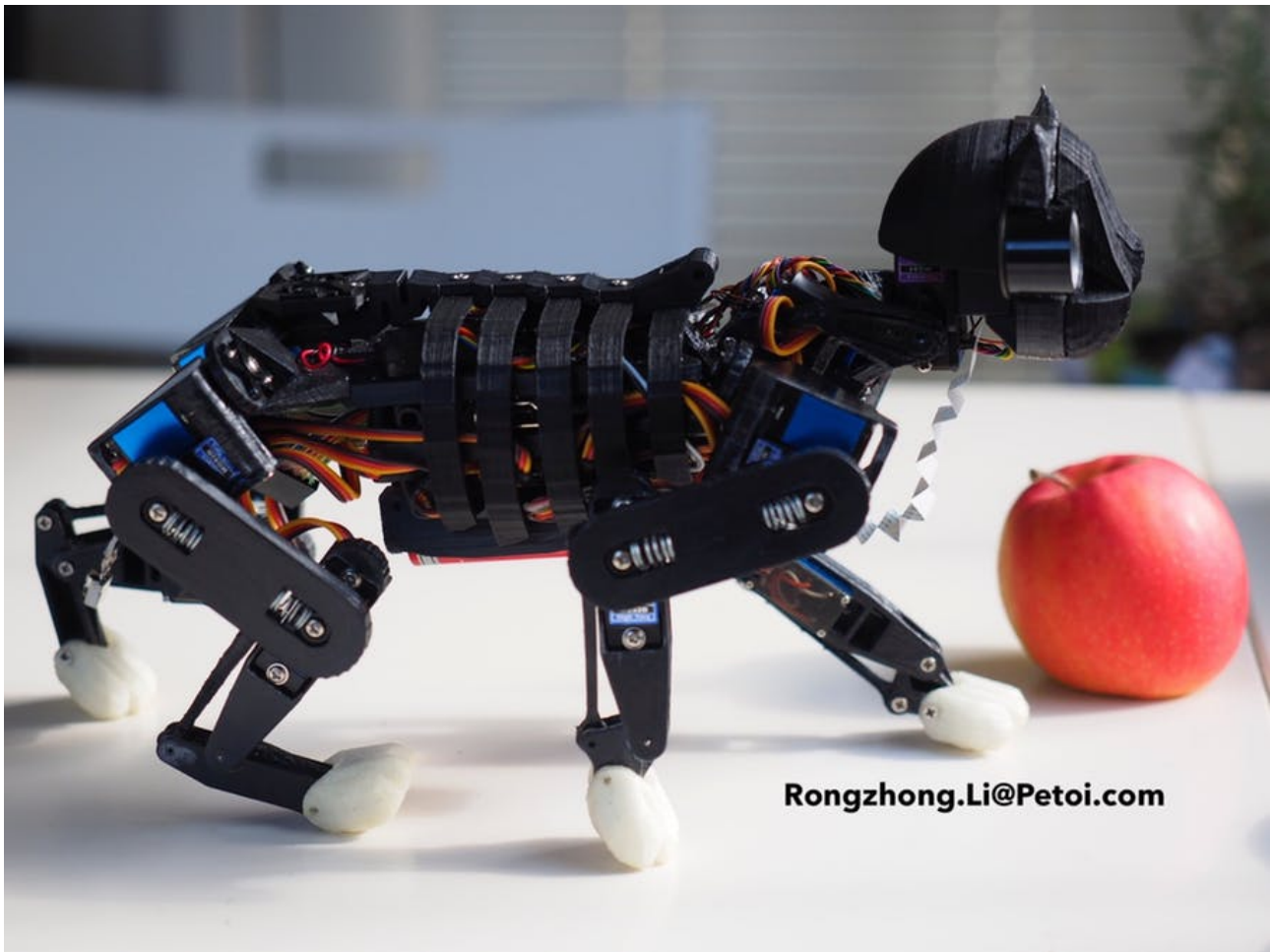
Links:

- [DyRET Documentation](#)



### 15.8.11.3 Open Cat

A programmable and highly maneuverable robotic cat for STEM education and AI-enhanced services.



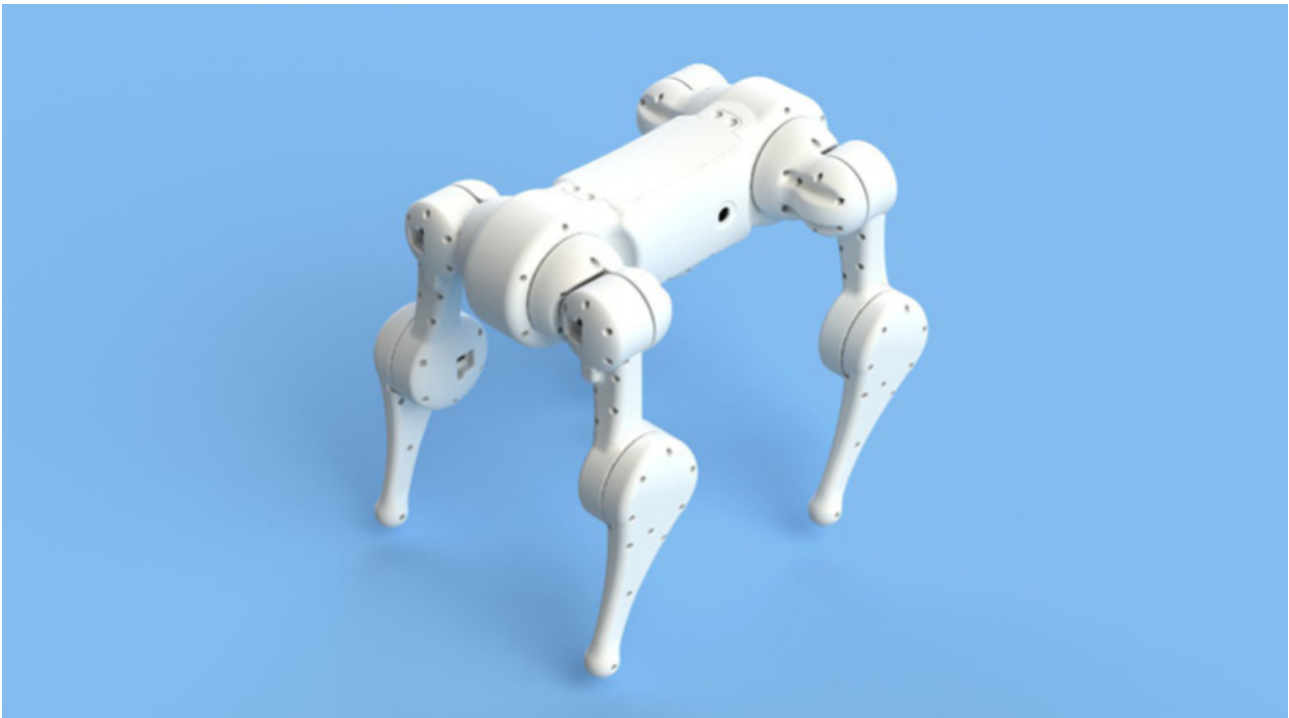
**Figure 15.14:** Open Cat

Article:

<https://www.hackster.io/petoi/opencat-845129>

### 15.8.11.4 Pet dog

A pet robot, just like Boston Dynamics makes



**Figure 15.15:** Pet dog

Articles:

- <https://hackaday.com/2019/03/30/a-pet-robot-just-like-boston-dynamics-makes/>
- <https://hackaday.io/project/164493-dizzy-wolf>

## 15.9 Mechanical keyboard

### 15.9.1 Inspiration

- <https://github.com/ruiqimao/keyboard-pcb-guide>
- <https://imgur.com/gallery/fGa13nZ>
- <https://www.reddit.com/r/MechanicalKeyboards/wiki/customkeyboards>

### 15.9.2 Tools

- Keyboard Layout editor
- Plate & Case Builder
- ai03 Plate Generator

## 15.10 Motorcycle App

- Profile
  - Navn

- epost
- område / by
- kommunikasjon (hjelm)
- Kjøretøy
  - model
  - årsmodel
  - merke
  - Bensin-logging
    - \* Stasjon / lokasjon
    - \* liter
    - \* tripteller
    - \* dato
    - \* drivstoff type (oktan)
    - \* drivstoff-pris
    - \* fulltank / ikke full tank
  - Vedlikehold
    - \* Sjekkliste
    - \* dekkbytte
    - \* bremseklosser
    - \* Diverse
- Venner
- Grupper
  - inviter venner til gruppe (lukket gruppe)
  - åpen gruppe
  - Åpen gruppe men begrenset godkjenning av admin
- Meldinger
  - venn til venn
  - gruppechat
  - turchat
- Ruter
  - Lag rute via kart
  - Lag rute ved å kjøre
  - Logg rute i bakgrunnen
  - legg til stopp punkt (pauser etc)
- Turer
  - planlegg rute via eksisterende rute
  - planlegg rute ved å lage via kart
  - inviter venner
  - inviter gruppe
  - kjør tur

- \* legg til møteplass
- \* legg til stopp (pauser etc)
- \* logg hvem som er med bassert på godkjenning og automatisk synkronisering av lokasjon
- \* logg faktisk kjørt rute
- \* logg tid
- \* logg tilfeldige forbipasserende (bassert på lokasjon og tid) (frivillig)
- statistikk
  - drivstoff forbruk
  - tid på sykkel
  - Avstand på sykkel

## 15.11 Pip-Boy

### 15.11.1 Montering

- <https://ytec3d.com/pip-boy-3000-mark-iv-assembly/>

### 15.11.2 LCD skjerm

- <https://no.mouser.com/ProductDetail/Newhaven-Display/NHD-43-480272MB-ASXN-CTP?qs=sGAEpiMZZMu%2fRY1bNe3bO6wz7D657ZcEt%252bYkb7GRrgfkxyS4IARIZA%3d%3dx>

## 15.12 Project Management System

### 15.12.1 Inspiration

- <https://codetree.com/>

## 15.13 RPI Security camera

### 15.13.1 Articles

- [Raspberry Pi Live Streaming camera: like on Times Square](#)

## 15.14 Reflow Oven

### 15.14.1 Links

**Tutorial:** <http://www.whizoo.com/reflowoven>

**Ovn:** <https://www.skousen.no/hvitevarer/ovn/mini-ovn/product/royal-16-ltr/>

**Isolasjonsteip:** <https://www.skruvat.no/Isolasjonstape-Reflect-A-Gold-P418338.aspx>

**Isolasjonsteppe:** <https://bakerovner.no/produkt/keramisk-isolasjon-rull-1260-c/>

**Fugemasse / lim:** <https://coop.no/sortiment/obs-bygg/maling-og-tilbehor/lim-fug-sparkel/casco-heat#product-info>

## 15.15 USB Media Controller

### 15.15.1 Dimensions

Høyde: 35mm x Bredde: 70 - 100mm

# 16 Security

## 16.1 Articles

- [Exploiting WiFi stack on Tesla Model S](#)

## 16.2 LoRaWAN

[LoRaWAN Encryption Keys Easy to Crack, Jeopardizing Security of IoT Networks](#)

# 17 Shopping lists

## 17.1 Shopping list for home office

### 17.1.1 Keyboard

- <https://www.daskeyboard.com/daskeyboard-4-ultimate/>
  - <https://www.teknikmagasinet.no/produkter/data-o-tv-spill/tastatur/varemerker/das-keyboard/das-keyboard-4-ultimate-with-cherry-mx-blue>
- [https://mechanicalkeyboards.com/shop/index.php?l=product\\_detail&p=3901](https://mechanicalkeyboards.com/shop/index.php?l=product_detail&p=3901)
- <http://www.wasdkeyboards.com/index.php/products/mechanical-keyboard/wasd-v2-105-key-iso-custom-mechanical-keyboard.html>

### 17.1.2 Network

- <https://mikrotik.com/product/RB3011UiAS-RM>
  - <https://www.eurodk.com/en/products/mt-rb/routerboard-3011uias-rm>
  - <https://freak.no/forum/showthread.php?t=219922&page=28>

# 18 Software

- [Rabbit MQ](#)

## 18.1 Rabbit MQ

### 18.1.1 Security

- Rabbit MQ access control: <http://www.rabbitmq.com/access-control.html>
- Multi-tenant SaaS AD: <https://vincentlauzon.com/2016/03/10/multi-tenant-saas-with-azure-active-directory-b2b-b2c/>

## 18.2 Ubuntu

### 18.2.1 Articles

- [Building a Linux Desktop for Cloud Native Development](#)

### 18.2.2 Ubuntu + Windows dual boot

Installing Windows **after** Ubuntu.

From <https://help.ubuntu.com/community/WindowsDualBoot>:

#### 18.2.2.1 Recovering GRUB after reinstalling Windows

Please refer to the [Reinstalling GRUB2](#) guide.

#### 18.2.2.2 Master Boot Record backup and replacement

This method does not work for computers with UEFI boot. In consequence, it won't work for pre-installed Windows 8 and some pre-installed with Windows 7.

Back-up the existing MBR, install Windows, replace your backup overwriting the Windows boot code:

1. Create an NTFS partition for Windows (using fdisk, GParted or whatever tool you are familiar with)



2. Backup the MBR e.g. `dd if=/dev/sda of=/mbr.bin bs=446 count=1`
3. Install Windows
4. Boot into a [LiveCD](#)
5. Mount your root partition in the LiveCD
6. Restore the MBR e.g. `dd if=/media/sda/mbr.bin of=/dev/sda bs=446 count=1`
7. Restart and Ubuntu will boot
8. Setup GRUB to boot Windows

## 18.3 tmux shortcuts & cheatsheet

start new:

```
tmux
```

start new with session name:

```
tmux new -s myname
```

attach:

```
tmux a # (or at, or attach)
```

attach to named:

```
tmux a -t myname
```

list sessions:

```
tmux ls
```

kill session:

```
tmux kill-session -t myname
```

Kill all the tmux sessions:

```
tmux ls | grep : | cut -d. -f1 | awk '{print substr($1, 0, length($1)-1)}' | xargs kill
```

In tmux, hit the prefix `ctrl+b` (my modified prefix is `ctrl+a`) and then:

### 18.3.1 Sessions

```
:new<CR>  new session
s  list sessions
$  name session
```

### 18.3.2 Windows (tabs)

```
c  create window
w  list windows
n  next window
p  previous window
f  find window
,  name window
&  kill window
```

### 18.3.3 Panes (splits)

```
%  vertical split
"  horizontal split

o  swap panes
q  show pane numbers
x  kill pane
+  break pane into window (e.g. to select text by mouse to copy)
-  restore pane from window
   space - toggle between layouts
<prefix> q (Show pane numbers, when the numbers show up type the key to goto that pane)
<prefix> { (Move the current pane left)
<prefix> } (Move the current pane right)
<prefix> z toggle pane zoom
```

### 18.3.4 Sync Panes

You can do this by switching to the appropriate window, typing your Tmux prefix (commonly Ctrl-B or Ctrl-A) and then a colon to bring up a Tmux command line, and typing:

```
:setw synchronize-panes
```

You can optionally add on or off to specify which state you want; otherwise the option is simply toggled. This option is specific to one window, so it won't change the way your other sessions or windows operate. When you're done, toggle it off again by repeating the command. [tip source](#)

### 18.3.5 Resizing Panes

You can also resize panes if you don't like the layout defaults. I personally rarely need to do this, though it's handy to know how. Here is the basic syntax to resize panes:

```
PREFIX : resize-pane -D (Resizes the current pane down)
PREFIX : resize-pane -U (Resizes the current pane upward)
PREFIX : resize-pane -L (Resizes the current pane left)
PREFIX : resize-pane -R (Resizes the current pane right)
PREFIX : resize-pane -D 20 (Resizes the current pane down by 20 cells)
PREFIX : resize-pane -U 20 (Resizes the current pane upward by 20 cells)
PREFIX : resize-pane -L 20 (Resizes the current pane left by 20 cells)
PREFIX : resize-pane -R 20 (Resizes the current pane right by 20 cells)
PREFIX : resize-pane -t 2 20 (Resizes the pane with the id of 2 down by 20 cells)
PREFIX : resize-pane -t -L 20 (Resizes the pane with the id of 2 left by 20 cells)
```

### 18.3.6 Copy mode

Pressing PREFIX [ places us in Copy mode. We can then use our movement keys to move our cursor around the screen. By default, the arrow keys work. we set our configuration file to use Vim keys for moving between windows and resizing panes so we wouldn't have to take our hands off the home row. tmux has a vi mode for working with the buffer as well. To enable it, add this line to .tmux.conf:

```
setw -g mode-keys vi
```

With this option set, we can use h, j, k, and l to move around our buffer.

To get out of Copy mode, we just press the ENTER key. Moving around one character at a time isn't very efficient. Since we enabled vi mode, we can also use some other visible shortcuts to move around the buffer.

For example, we can use "w" to jump to the next word and "b" to jump back one word. And we can use "f", followed by any character, to jump to that character on the same line, and "F" to jump backwards on the line.

Function	vi	emacs
Back to indentation	^	M-m
Clear selection	Escape	C-g
Copy selection	Enter	M-w
Cursor down	j	Down
Cursor left	h	Left
Cursor right	l	Right
Cursor to bottom line	L	
Cursor to middle line	M	M-r
Cursor to top line	H	M-R
Cursor up	k	Up

Delete entire line	d	C-u
Delete to end of line	D	C-k
End of line	\$	C-e
Goto line	:	g
Half page down	C-d	M-Down
Half page up	C-u	M-Up
Next page	C-f	Page down
Next word	w	M-f
Paste buffer	p	C-y
Previous page	C-b	Page up
Previous word	b	M-b
Quit mode	q	Escape
Scroll down	C-Down or J	C-Down
Scroll up	C-Up or K	C-Up
Search again	n	n
Search backward	?	C-r
Search forward	/	C-s
Start of line	0	C-a
Start selection	Space	C-Space
Transpose chars		C-t

### 18.3.7 Misc

```
d detach
t big clock
? list shortcuts
: prompt
```

### 18.3.8 Configurations Options

```
# Mouse support - set to on if you want to use the mouse
* setw -g mode-mouse off
* set -g mouse-select-pane off
* set -g mouse-resize-pane off
* set -g mouse-select-window off

# Set the default terminal mode to 256color mode
set -g default-terminal "screen-256color"

# enable activity alerts
setw -g monitor-activity on
set -g visual-activity on

# Center the window list
```

```
set -g status-justify centre
```

```
# Maximize and restore a pane
```

```
unbind Up bind Up new-window -d -n tmp \; swap-pane -s tmp.1 \; select-  
window -t tmp
```

```
unbind Down
```

```
bind Down last-window \; swap-pane -s tmp.1 \; kill-window -t tmp
```

### 18.3.9 Resources

- [tmux: Productive Mouse-Free Development](#)
- [How to reorder windows](#)

## 18.4 Vim Cheat Sheet

### 18.4.1 Global

:help keyword - open help for keyword

:saveas file - save file as

:close - close current pane

K - open man page for word under the cursor

### 18.4.2 Cursor movement

h - move cursor left

j - move cursor down

k - move cursor up

l - move cursor right

H - move to top of screen

M - move to middle of screen

L - move to bottom of screen

w - jump forwards to the start of a word

W - jump forwards to the start of a word (words can contain punctuation)

e - jump forwards to the end of a word

E - jump forwards to the end of a word (words can contain punctuation)

b - jump backwards to the start of a word

B - jump backwards to the start of a word (words can contain punctuation)

% - move to matching character (default supported pairs: '()', '{}', '[]' - use :h matchpairs in vim for more info)

0 - jump to the start of the line

^ - jump to the first non-blank character of the line

\$ - jump to the end of the line

g\_ - jump to the last non-blank character of the line

gg - go to the first line of the document

G - go to the last line of the document

5G - go to line 5

fx - jump to next occurrence of character x

tx - jump to before next occurrence of character x

Fx - jump to previous occurrence of character x

Tx - jump to after previous occurrence of character x

; - repeat previous f, t, F or T movement

, - repeat previous f, t, F or T movement, backwards

} - jump to next paragraph (or function/block, when editing code)

{ - jump to previous paragraph (or function/block, when editing code)

zz - center cursor on screen

Ctrl + e - move screen down one line (without moving cursor)

Ctrl + y - move screen up one line (without moving cursor)

Ctrl + b - move back one full screen

Ctrl + f - move forward one full screen

Ctrl + d - move forward 1/2 a screen

Ctrl + u - move back 1/2 a screen

### 18.4.3 Insert mode - inserting/ appending text

i - insert before the cursor

I - insert at the beginning of the line

a - insert (append) after the cursor

A - insert (append) at the end of the line

o - append (open) a new line below the current line

O - append (open) a new line above the current line

ea - insert (append) at the end of the word

Esc - exit insert mode

#### 18.4.4 Editing

r - replace a single character  
J - join line below to the current one with one space in between  
gJ - join line below to the current one without space in between  
gwip - reflow paragraph  
cc - change (replace) entire line  
C - change (replace) to the end of the line  
c\$ - change (replace) to the end of the line  
ciw - change (replace) entire word  
cw - change (replace) to the end of the word  
s - delete character and substitute text  
S - delete line and substitute text (same as cc)  
xp - transpose two letters (delete and paste)  
u - undo  
Ctrl + r - redo  
. - repeat last command

#### 18.4.5 Cut and paste

yy - yank (copy) a line  
2yy - yank (copy) 2 lines  
yw - yank (copy) the characters of the word from the cursor position to the start of the next word y\$ -  
yank (copy) to end of line  
p - put (paste) the clipboard after cursor  
P - put (paste) before cursor  
dd - delete (cut) a line  
2dd - delete (cut) 2 lines  
dw - delete (cut) the characters of the word from the cursor position to the start of the next word D -  
delete (cut) to the end of the line  
d\$ - delete (cut) to the end of the line  
x - delete (cut) character

### 18.4.6 Marking text (visual mode)

v - start visual mode, mark lines, then do a command (like y-yank)

V - start linewise visual mode

o - move to other end of marked area

Ctrl + v - start visual block mode

O - move to other corner of block

aw - mark a word

ab - a block with ()

aB - a block with {}

ib - inner block with ()

iB - inner block with {}

Esc - exit visual mode

### 18.4.7 Visual commands

> - shift text right

< - shift text left

y - yank (copy) marked text

d - delete marked text

~ - switch case

### 18.4.8 Registers

:reg - show registers content

"xy - yank into register x

"xp - paste contents of register x

### 18.4.9 Search and replace

/pattern - search for pattern

?pattern - search backward for pattern

\vpattern - 'very magic' pattern: non-alphanumeric characters are interpreted as special regex symbols (no escaping needed)

n - repeat search in same direction

N - repeat search in opposite direction



:%s/old/new/g - replace all old with new throughout file  
:%s/old/new/gc - replace all old with new throughout file with confirmations  
:noh - remove highlighting of search matches

#### 18.4.10 Search in multiple files

:vimgrep /pattern/ {{file}} - search for pattern in multiple files e.g.

```
:vimgrep /foo/ **/*
```

:cn - jump to the next match  
:cp - jump to the previous match  
:copen - open a window containing the list of matches

#### 18.4.11 Working with multiple files

:e file - edit a file in a new buffer  
:bnext or :bn - go to the next buffer  
:bprev or :bp - go to the previous buffer  
:bd - delete a buffer (close a file)  
:ls - list all open buffers  
:sp file - open a file in a new buffer and split window  
:vsp file - open a file in a new buffer and vertically split window  
Ctrl + ws - split window  
Ctrl + ww - switch windows  
Ctrl + wq - quit a window  
Ctrl + wv - split window vertically  
Ctrl + wh - move cursor to the left window (vertical split)  
Ctrl + wl - move cursor to the right window (vertical split)  
Ctrl + wj - move cursor to the window below (horizontal split)  
Ctrl + wk - move cursor to the window above (horizontal split)

#### 18.4.12 Tabs

:tabnew or :tabnew {page.words.file} - open a file in a new tab  
Ctrl + wT - move the current split window into its own tab

gt or :tabnext or :tabn - move to the next tab  
gT or :tabprev or :tabp - move to the previous tab  
#gt - move to tab number #  
:tabmove # - move current tab to the #th position (indexed from 0)  
:tabclose or :tabc - close the current tab and all its windows  
:tabonly or :tabo - close all tabs except for the current one  
:tabdo command - run the command on all tabs (e.g. :tabdo q - closes all opened tabs)

### 18.4.13 Exiting

:w - write (save) the file, but don't exit  
:w !sudo tee % - write out the current file using sudo  
:wq or :x or ZZ - write (save) and quit  
:q - quit (fails if there are unsaved changes)  
:q! or ZQ - quit and throw away unsaved changes  
:wqa - write (save) and quit on all tabs

# 19 UX - UI

- [Colors](#)

## 19.1 Methods

- <https://material.io/design/>
- <http://www.designkit.org/methods>

## 19.2 Colors

### 19.2.1 Links

- <https://www.canva.com/colors/color-palette-generator/>

## 20 Useful stuff

### 20.1 Useful Commands

#### 20.1.1 Terminal recording

[Asciinema](#)

##### 20.1.1.1 1. Install

```
brew install asciinema
```

##### 20.1.1.2 2. Record

```
asciinema rec filename.cast
```

##### 20.1.1.3 3. Play

```
asciinema play filename.cast
```

#### 20.1.2 WiFi QR-code

```
qrencode -o wifi.png "WIFI:T:WPA;S:<SSID>;P:<PASSWORD>; "
```

#### 20.1.3 Rsync

[Rsync cheatsheet](#)

```
## syncing folder src into dest:
rsync -avzP ./src /dest
## syncing the content of src into dest:
rsync -avzP ./src/ /dest
```

## 20.1.4 Unite PDF documents

### 20.1.4.1 Install

```
brew install poppler
```

### 20.1.4.2 Usage

```
pdfunite file1.pdf file2.pdf output.pdf
```