IoT Caseiro

Hugo Djemaa - 181596

Sumario

1. Como e porque

Criar

Prototipar

Monitorar

Ensinar

2. Raspberry e Arduino

Raspberry pi

Arduino

Os fazer comunicar

E mais

3. Hardware não é tudo

Open Data

Umas ferramentas

Aplicaçao

Como e Porque

- IoT é barato
- Tem uma comunidade enorme
- Muitos projetos Open-Source
- O único limite é a imaginação

1.
Criar

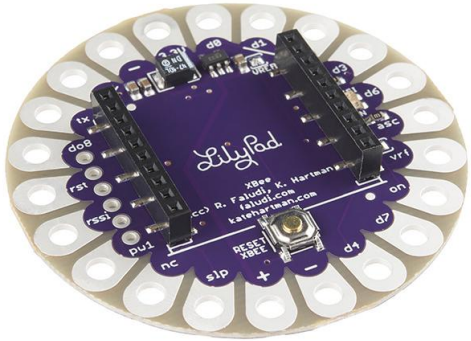
Criar

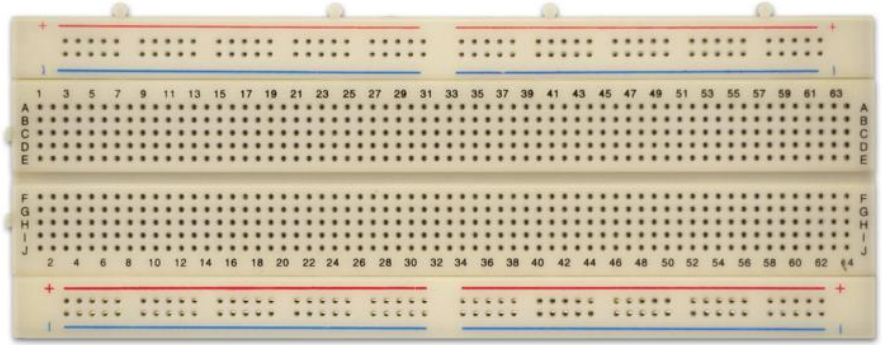
- Objeto + Raspberry pi => Objeto inteligente
- O único limite é a imaginação
- Espelho + Tela + Raspberry => Smart Mirror
- Várias ideias e projetos abertos

2.
Prototipar

Prototipar

- Criar um protótipo antes de o produzir
- Acessibilidade de fazer e desfazer => custo baixo
- Componente muito baratos => custo baixo
- Confiança nos produtos => Apoio da comunidade (crowdfunding)

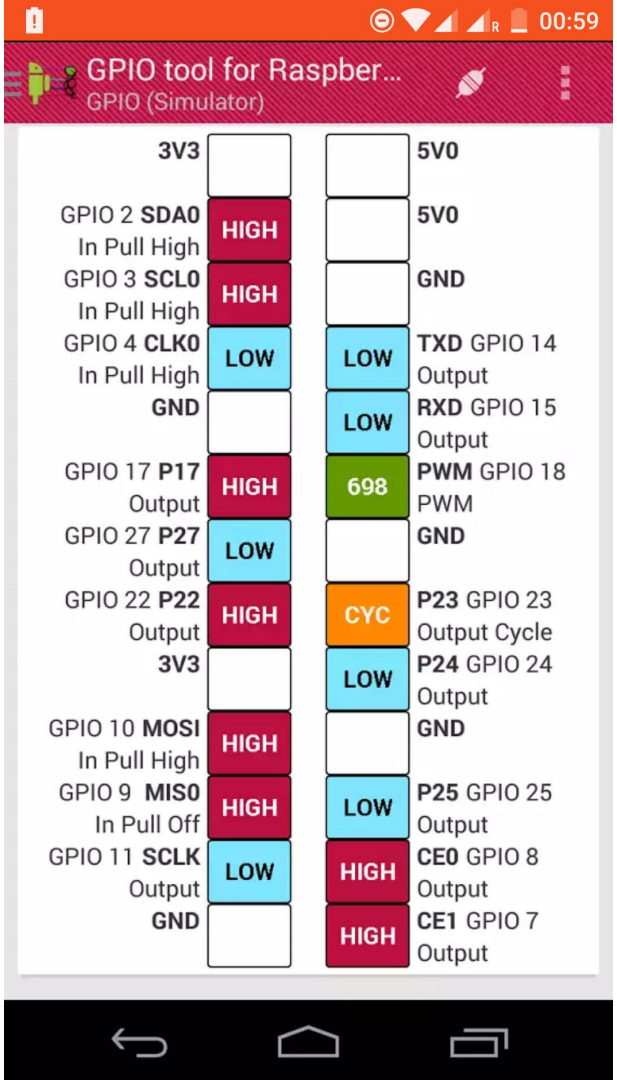




3.
Monitorar

Monitorar

- Diagnosticar e monitorar um carro novo via ODB-II
- Muitos aplicativos disponíveis ou Open Source, ou gratuito no celular para monitorar



4.
Ensinar

Ensinar

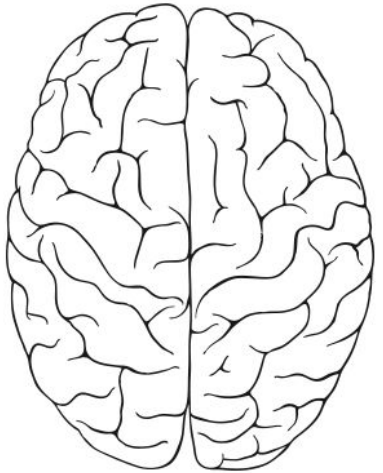
- Software acessíveis para crianças
- Comunidade
- Estimula a criatividade

Raspberry e Arduino

E mais...

1.
Raspberry Pi

Raspberry Pi



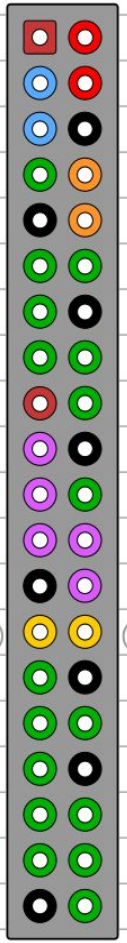
- Microprocessador
- Tem um Sistema Operativo (Maioria com linux)
- Basicamente um computador
- GCC / Python / MEAN / Scratch / Bash …
- 40 GPIO
- 1Gb de RAM
- Memoria SD
- USB / HDMI / Auxiliar / Ethernet (/ Bluetooth / WiFi no 3)
- CPU : 1.2GHz 64-bit quad-core ARMv8

Raspberry Pi : I/O

- Nao tem pins analogico
- É possível simular uma saída analogica com a PWM (GPIO_GENi)

Raspberry Pi 3 GPIO 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

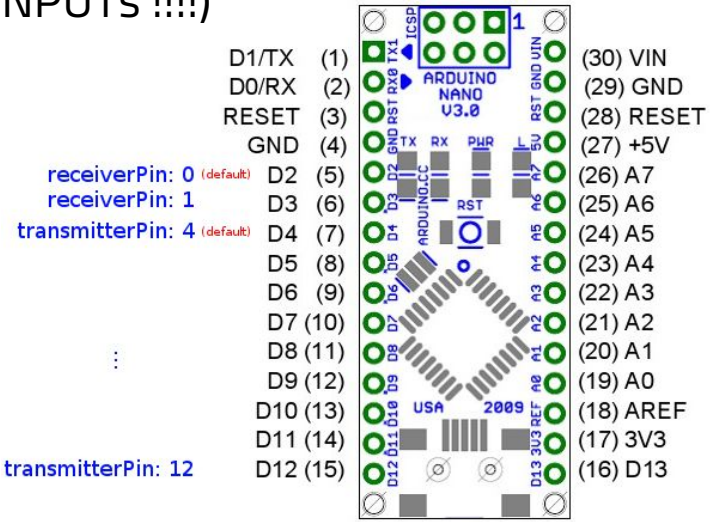


www.element14.com/RaspberryPi

2.
Arduino

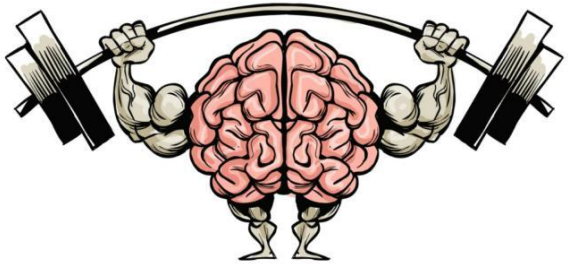
Arduino

- Microcontrolador
- Nao tem Sistema Operativo
- void setup() & void loop()
- CPU ATmega328P 16MHz
- Um monte de GPIO (E ANALOGs INPUTs !!!!)



3.
Comunicaçao

Juntar as forças



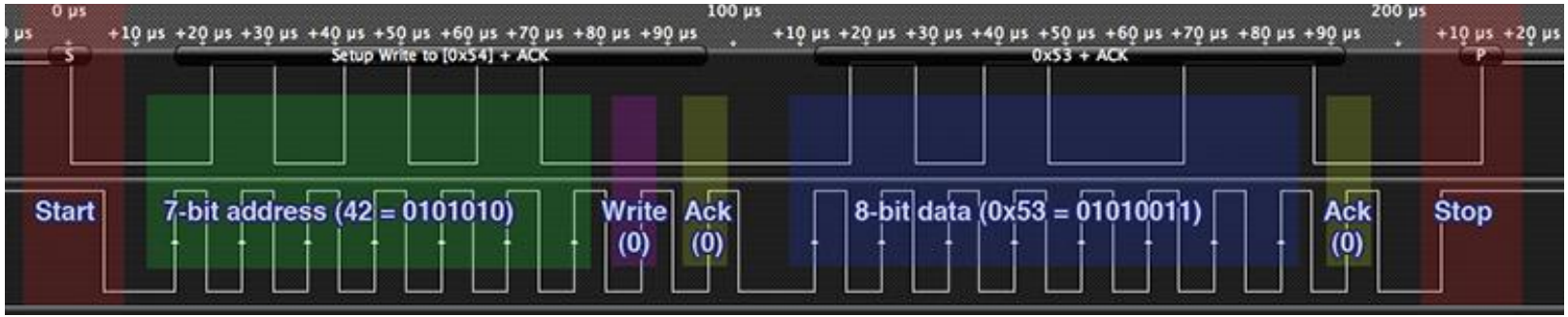
- Tem vários jeitos de fazer comunicar um Arduino e um Raspberry
- Capacidade de processamento do Raspberry Pi
- Muitas I/O via Arduino
- Por exemplo um servidor web de gerenciamento de relay e sensores (Pi)
- + um Arduino em cada quarto para atuar com os captores
- ↳Ja temos um Smart Home

1 Wire

- 1 fio
- Master to Slave unicamente (simplex)
- Endereço de 64 bits => na teoria o limite de devices é muito alto
- Fluxo ~ 1.5Kbyte/s
- Uso : temperature sensor, track ID …
- Bit menos significativo primeiro

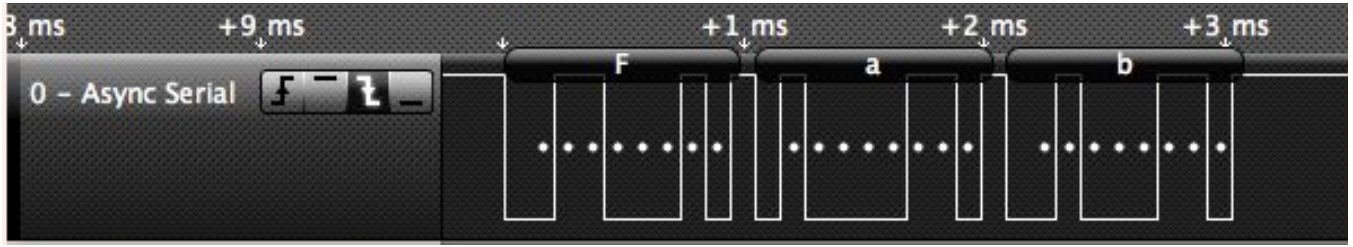
I2C

- 2 fios : SDA (Data), SCL (Clock) default : 100KHz
- Half duplex
- 120 endereço +1 de broadcast
- Fluxo ~ 4.5Kbytes/s
- Uso : comunicação entre processos, tela LCD pequena
- Bit mais significativo primeiro



UART / Serial (RS232)

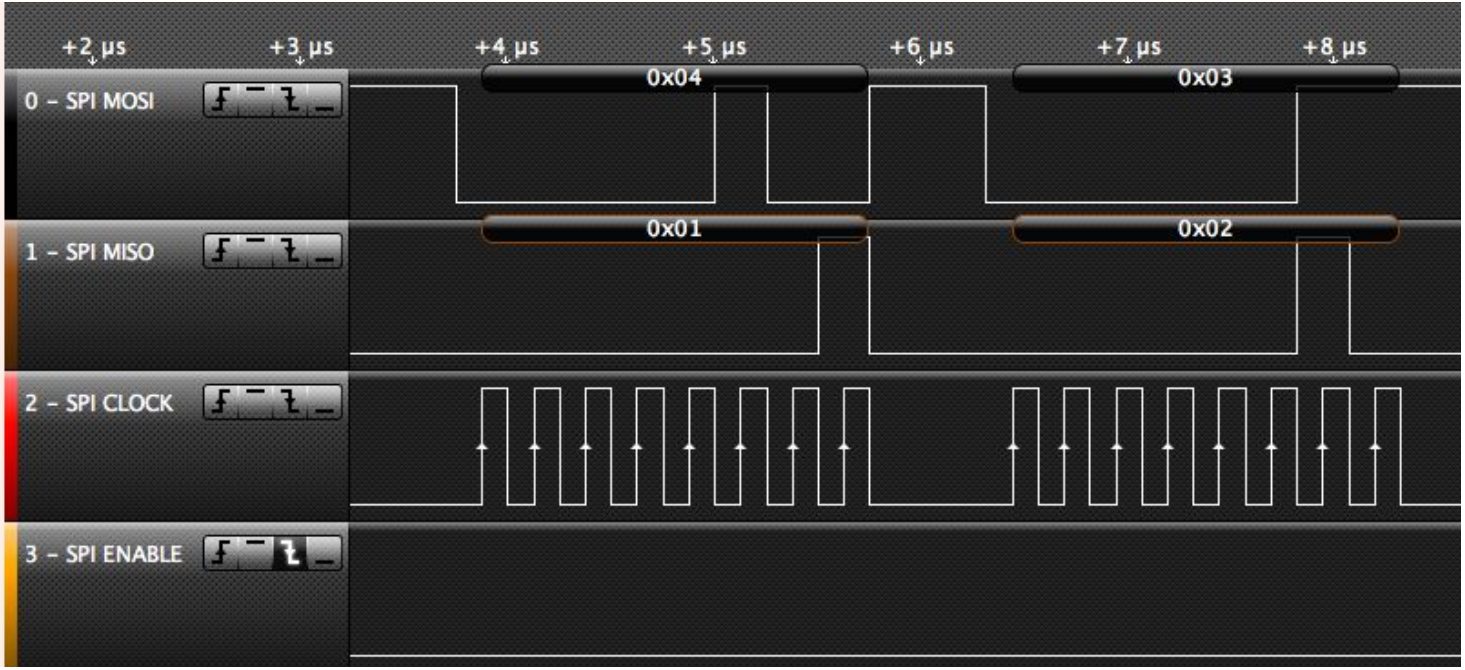
- 2 fios : Rx (Receiver), Tx (Transmitter)
- Half duplex
- Entre dois devices
- Uso : ?
- Fluxo ~ 1Kbyte/s
- Bit menos significativo primeiro



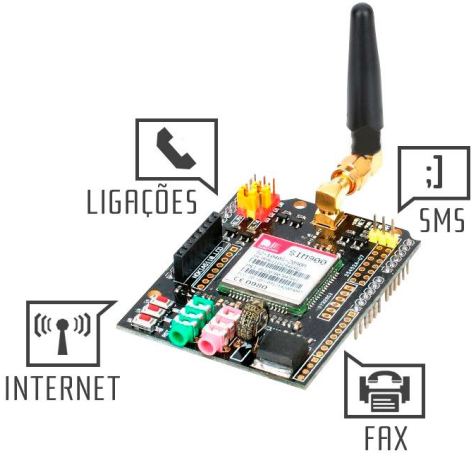
SPI

- 4 Fios :
  - SCK (Serial Clock)
  - MISO (Master In, Slave Out)
  - MOSI (Master Out, Slave In)
  - SS (Slave Select)
- Full duplex
- Fluxo ~ 330Kbyte/s
- Clock a 4MHz
- Bit mais significativo primeiro
- Comunicaçao

SPI

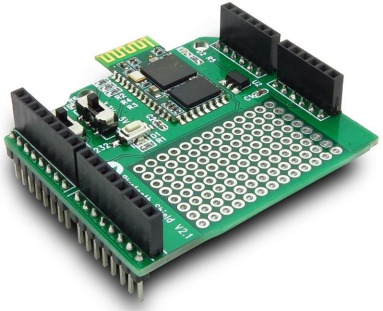

+2 µs
+3 µs
+4 µs
+5 µs
+6 µs
+7 µs
+8 µs
0 – SPI MOSI
0x04
0x03
1 – SPI MISO
0x01
0x02
2 – SPI CLOCK
3 – SPI ENABLE

E tambem


LIGAÇÕES
SMS
INTERNET
FAX


HanRun
HR911105A
10/49

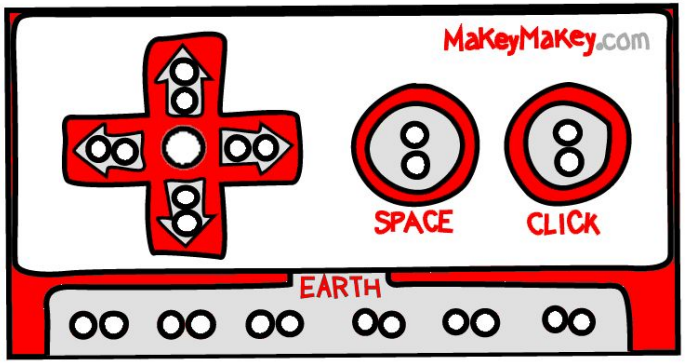

IPv6




Max-Stream
XBee
PRO
FCC ID: OUR-XBEEPRO
IC ID: 4214A-XBEEPRO
www.maxstream.net

4.
E mais

Novos produtos sempre



Hardware não é tudo

1.
Open data


Hackear nunca foi tão fácil
Praça Ramos de Azevedo. Chega em 1 minuto.

2.
Umas ferramentas

Como aproveitar dos dados abertos

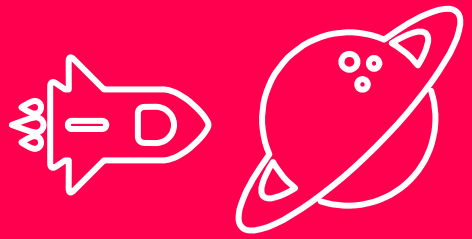
- Python que é uma linguagem de scripts muito simples e eficiente
- cURL em linha de comandos
- MEAN Stack muito mais simples :
  - MongoDB
  - Express.js
  - AngularJS
  - NodeJS

3.
Aplicaçao

Je veux un vélo

Referencias

- **http://www.gammon.com.au/forum/?id=10918**
  Nick Gammon
- **Raspberry Pi Schematics**
  James Adams
- **Arduino Schematics**
  Arduino
- **http://www.instructables.com/**



Perguntas ?