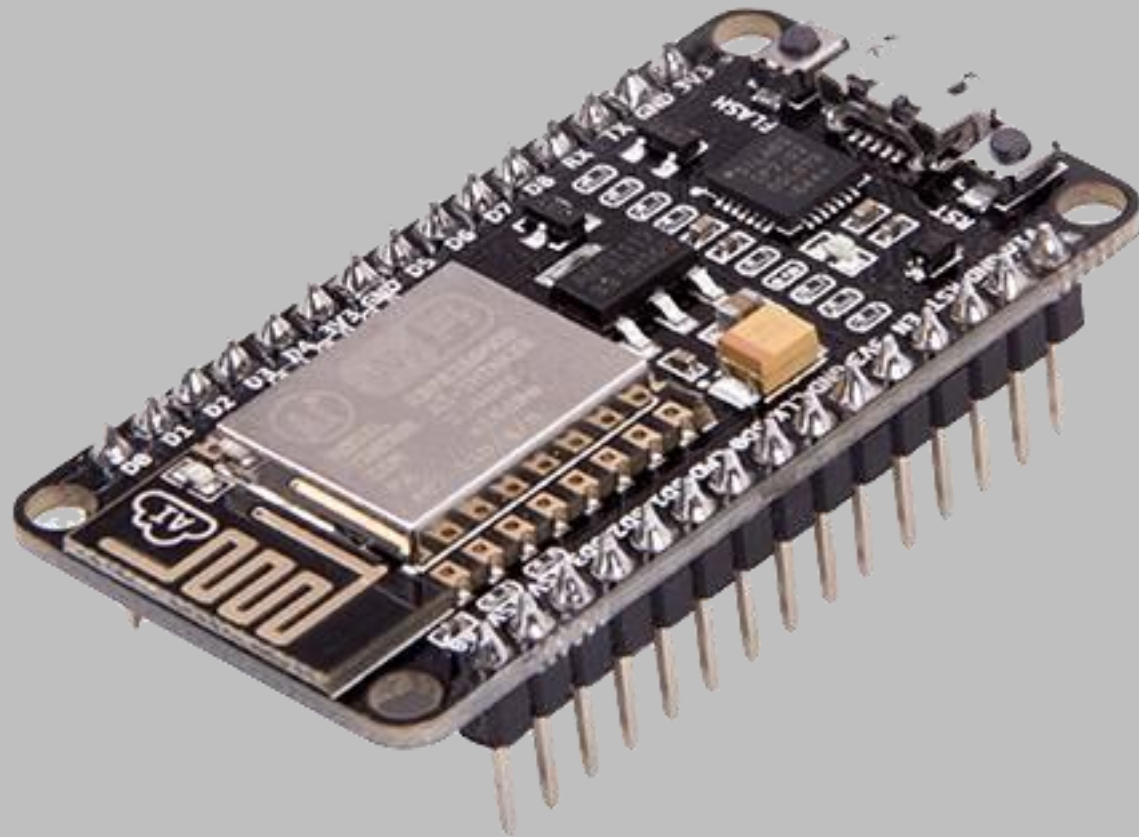


Detalhes e pinagem NodeMCU 1.0 ESP-12E



Por Fernando Koyanagi

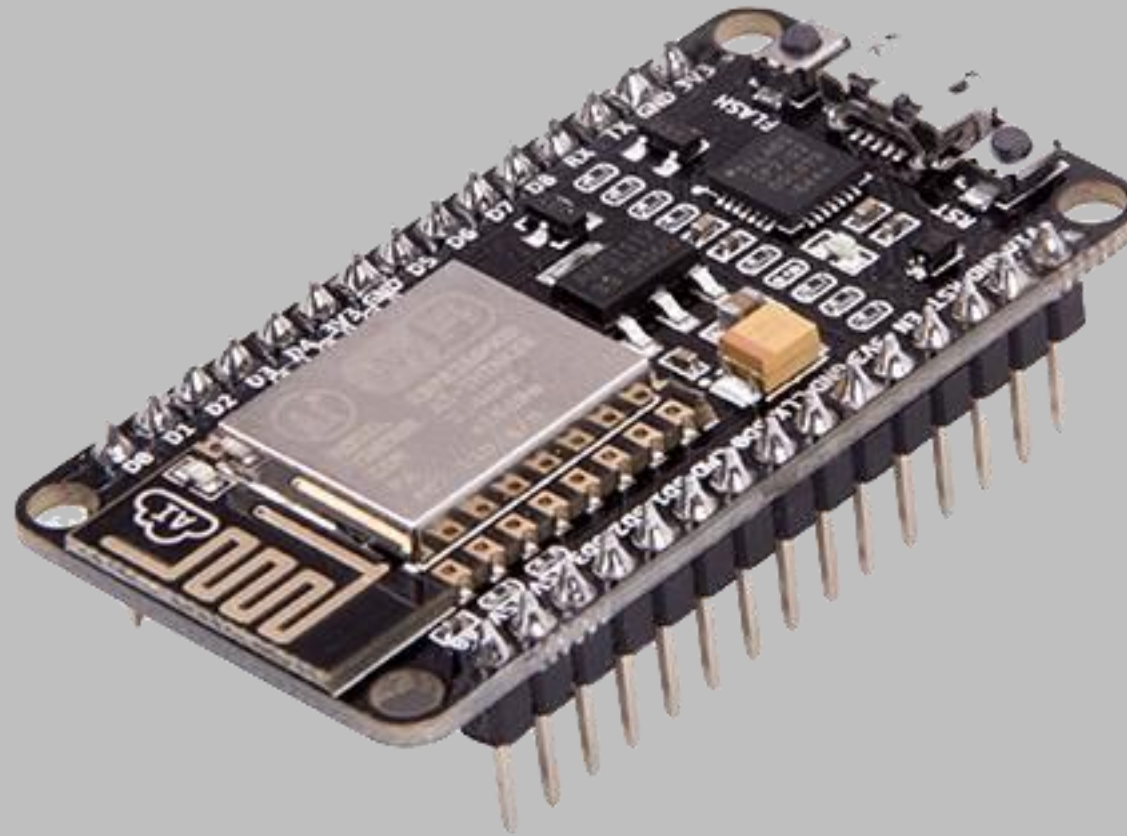
Intenção dessa aula

- 1. Aprender qual a identificação correta dos pinos olhando o datasheet do NodeMCU**
 - 2. Informar quais os pinos funcionam com digitalWrite, digitalRead, analogWrite e analogRead**
- 1. Explicar sobre o boot**

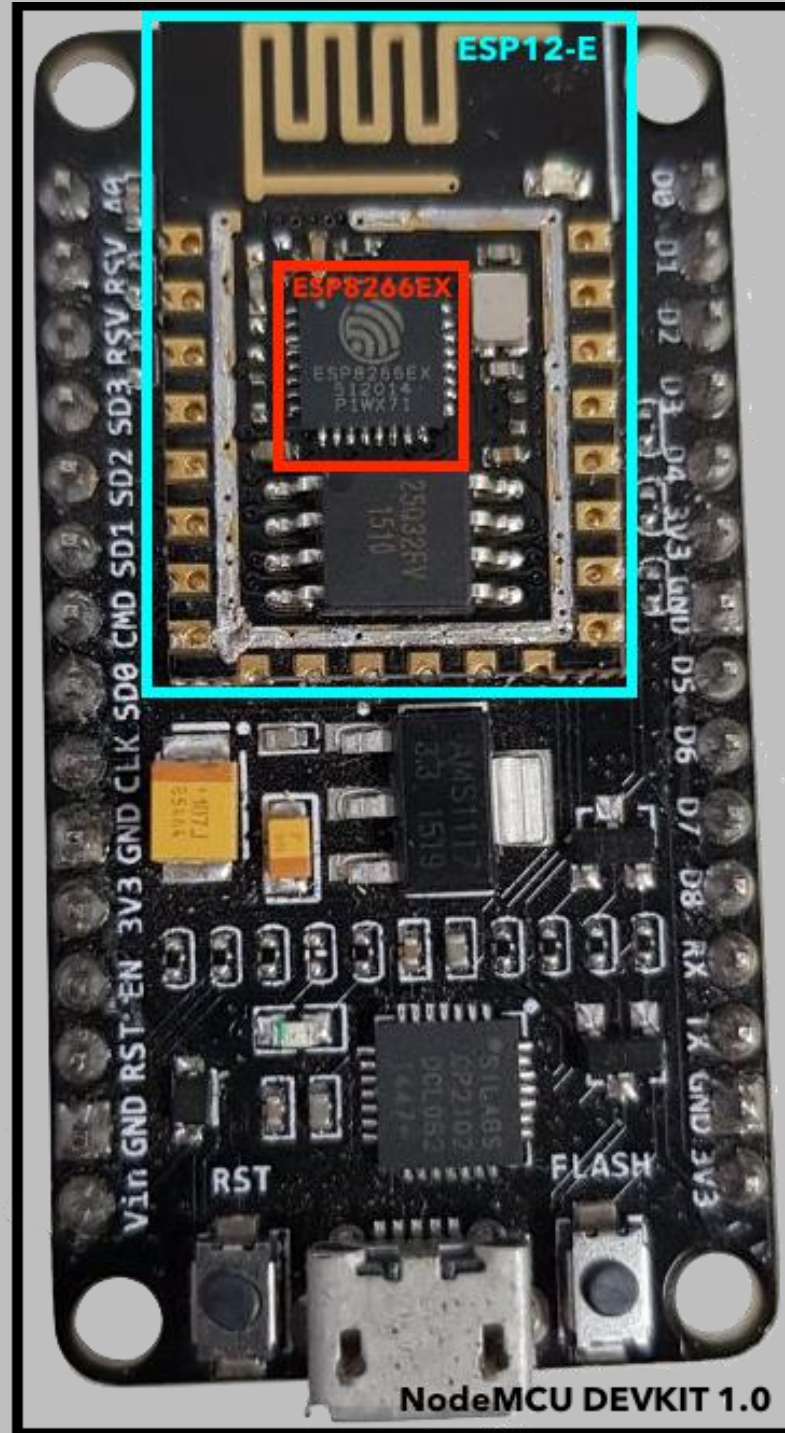


NodeMCU Devkit 1.0

O termo **NodeMCU** geralmente se refere ao **firmware**, enquanto a placa é denominada **devkit**.
O NodeMCU Devkit 1.0 é consituído de um **ESP-12E** em uma placa que facilita o seu uso.

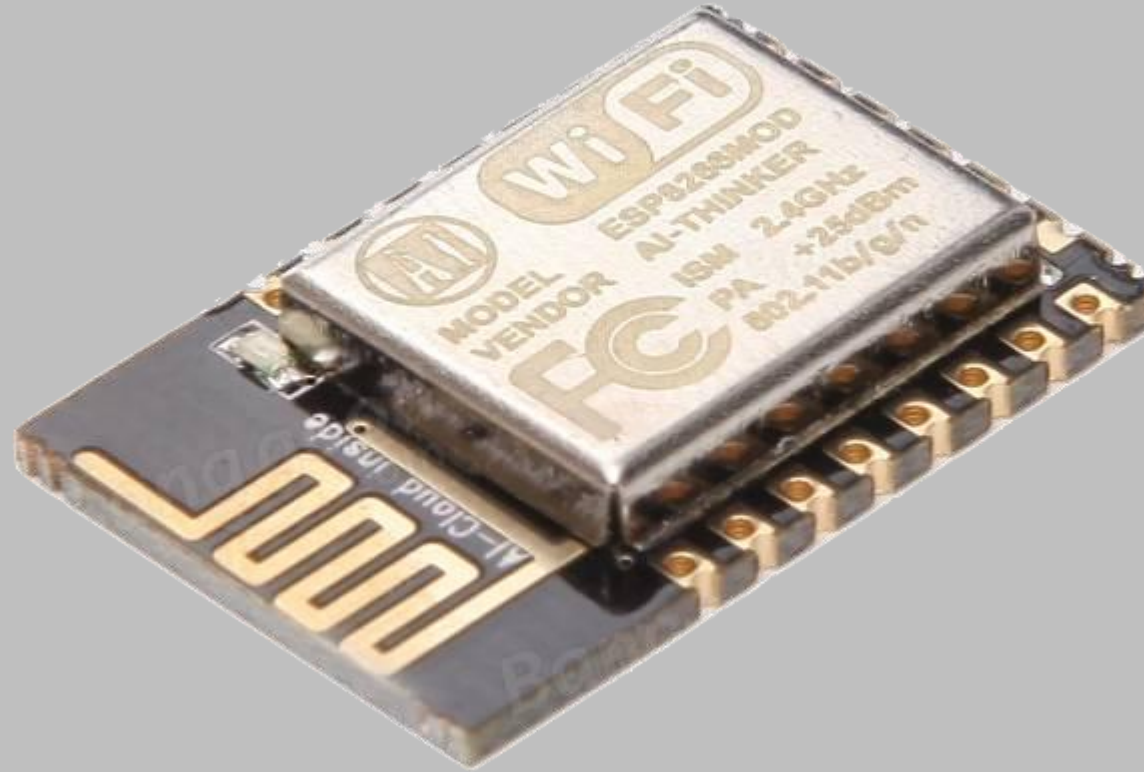


NodeMCU Devkit 1.0



ESP-12E

ESP-12E é uma placa criada pela AI-THINKER sendo constituído por sua vez por um ESP8266EX (dentro da capa de metal)



ESP8266EX

Feito pela Espressif, este microchip possui WiFi integrado e baixo consumo de energia.
Processador RISC Tensilica L 106 32bit com clock máximo de 160 MHz

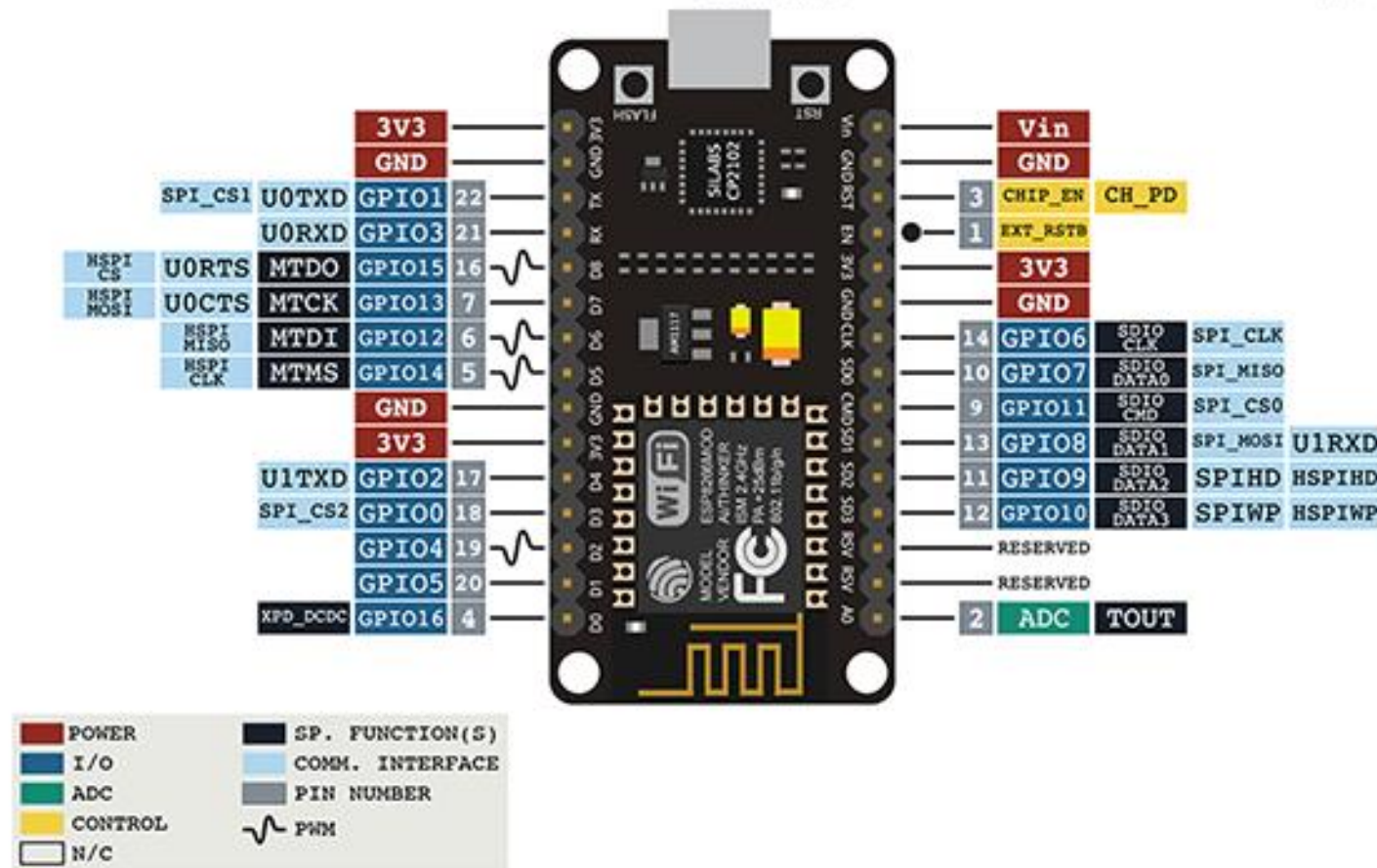


NodeMCU 1.0 ESP-12E Pinout

ESP-12E DEVELOPMENT BOARD PINOUT

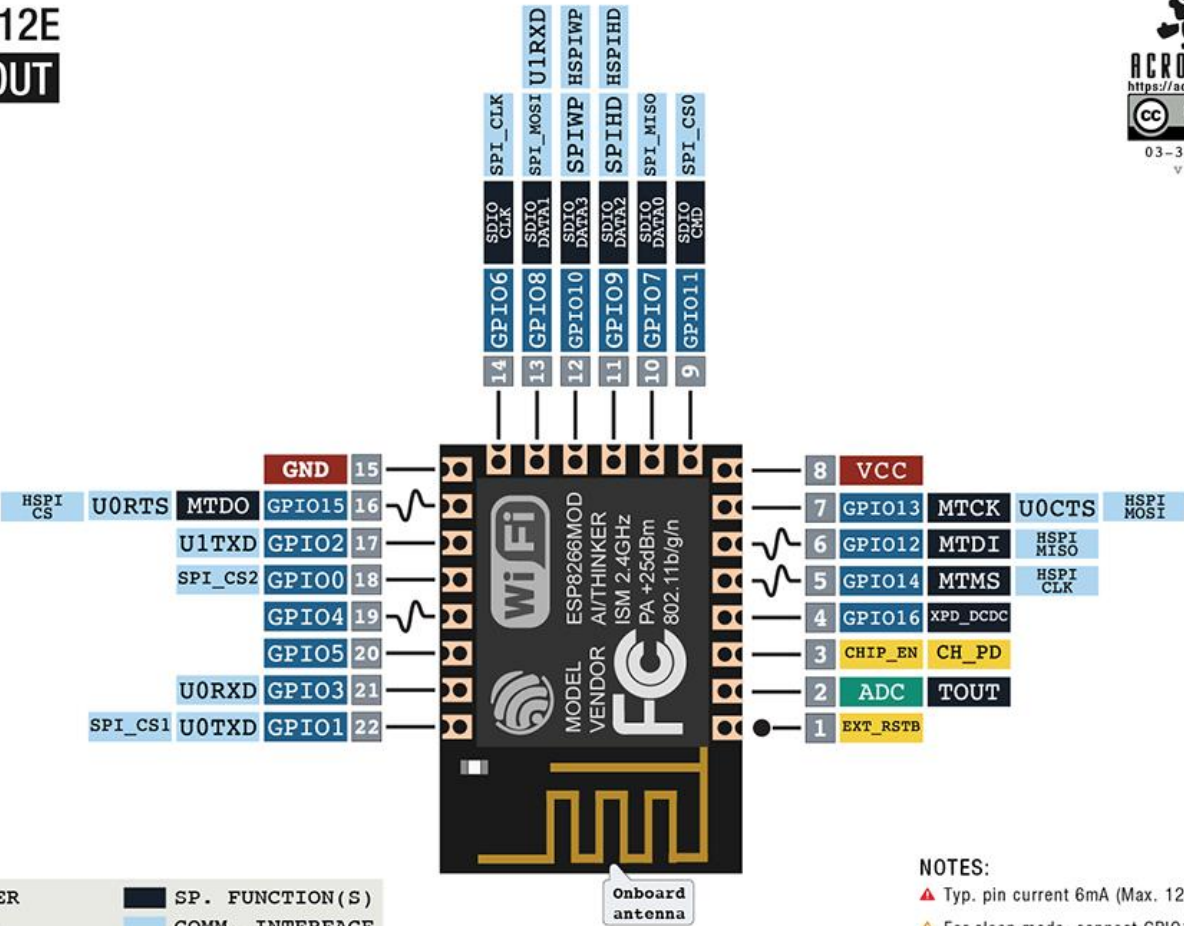
NOTES:

- ▲ Typ. pin current 6mA (Max. 12mA)
- ▲ For sleep mode, connect GPIO16 and EXT_RSTB. On wakeup, GPIO16 will output LOW for system reset.
- ▲ On boot/reset/wakeup, keep GPIO15 LOW and GPIO2 HIGH.



ESP-12E Pinout

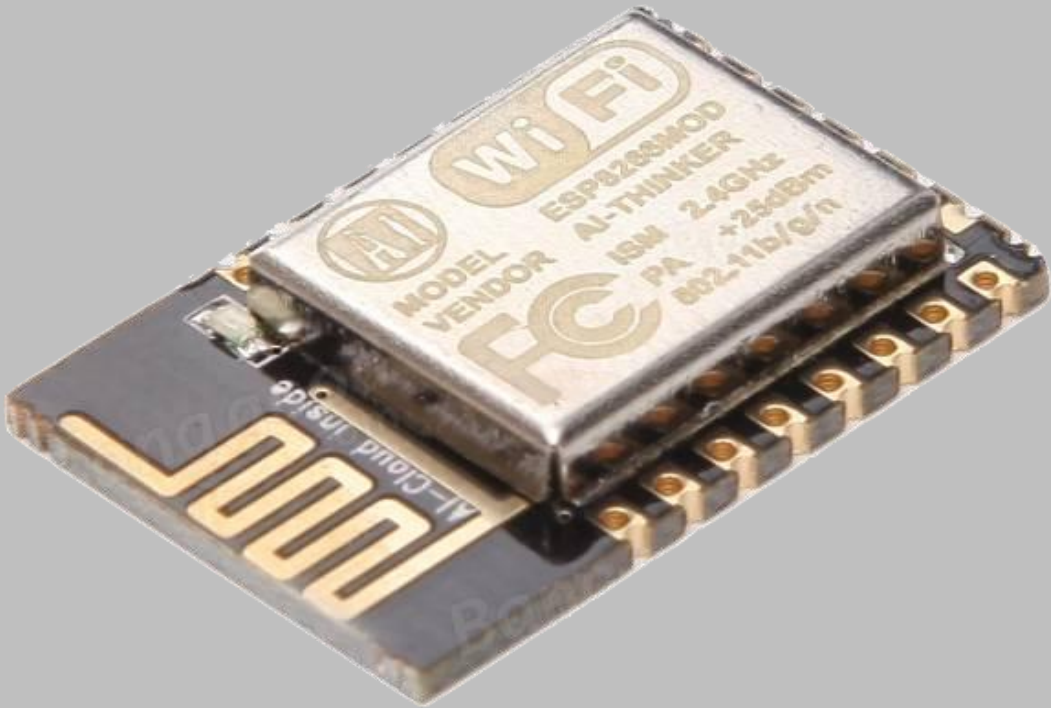
ESP-12E PINOUT



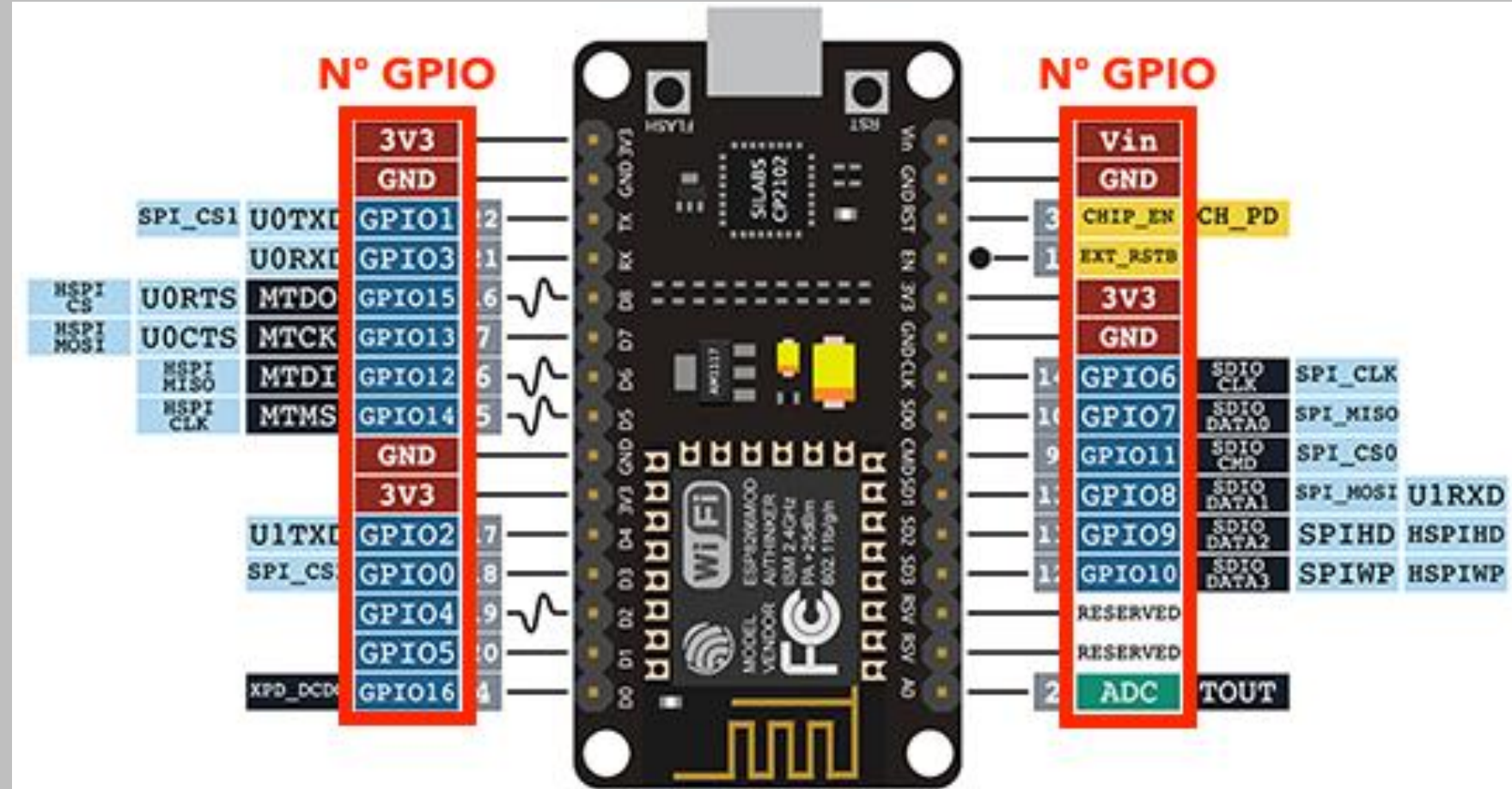
POWER	SP. FUNCTION(S)
I/O	COMM. INTERFACE
ADC	PIN NUMBER
CONTROL	PWM
N/C	

NOTES:

- ▲ Typ. pin current 6mA (Max. 12mA)
- ▲ For sleep mode, connect GPIOD and EXT_RSTB. On wakeup, GPIOD will output LOW for system reset.
- ▲ On boot/reset/wakeup, keep GPIOD LOW and GPIOD HIGH.



Mas qual o número que coloco quando estou programando?



Utilize o **número** que está na frente do **GPIO** ou as constantes A0, D0, D1, D2, D3, D4, D5, D6, D7 e D8



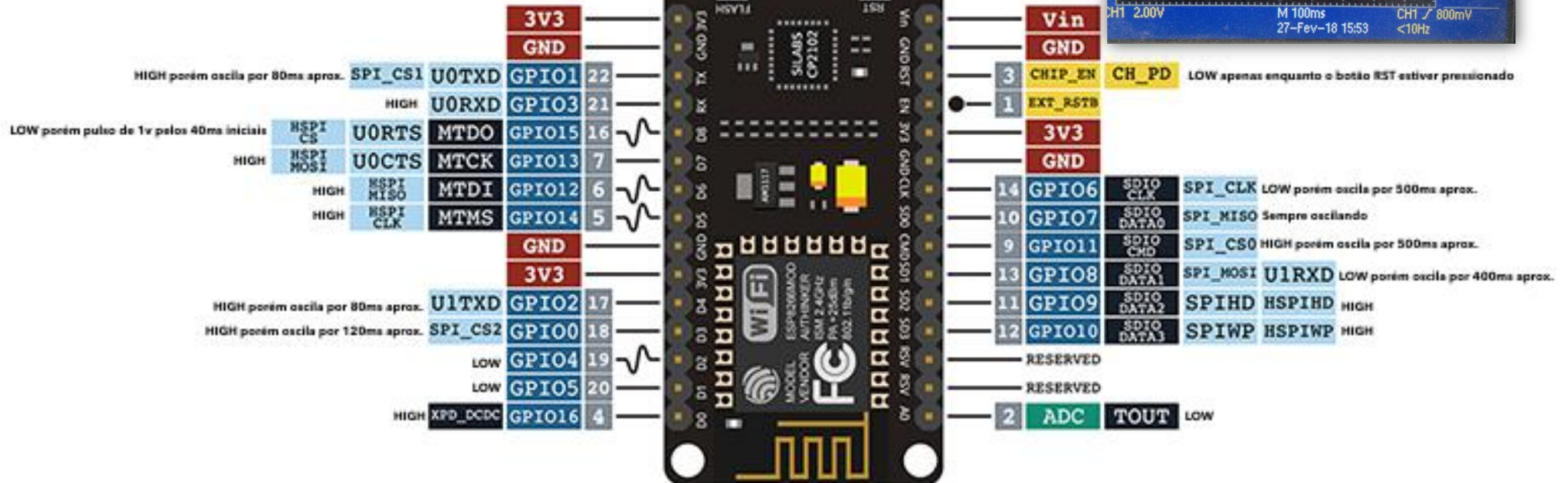
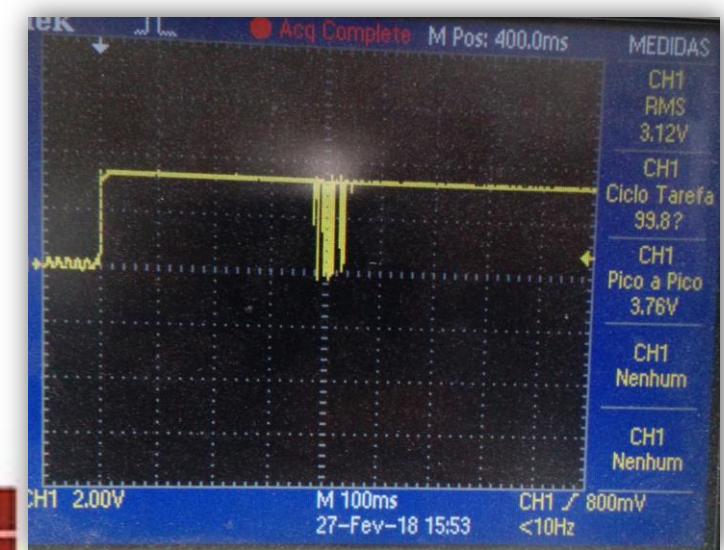
Boot

ESP-12E DEVELOPMENT BOARD PINOUT

Observe o comportamento de cada pino quando o NodeMCU é iniciado (boot).

NOTES:

- ▲ Typ. pin current 6mA (Max. 12mA)
- ▲ For sleep mode, connect GPIO16 and EXT_RSTB. On wakeup, GPIO16 will output LOW for system reset.
- ▲ On boot/reset/wakeup, keep GPIO15 LOW and GPIO2 HIGH.



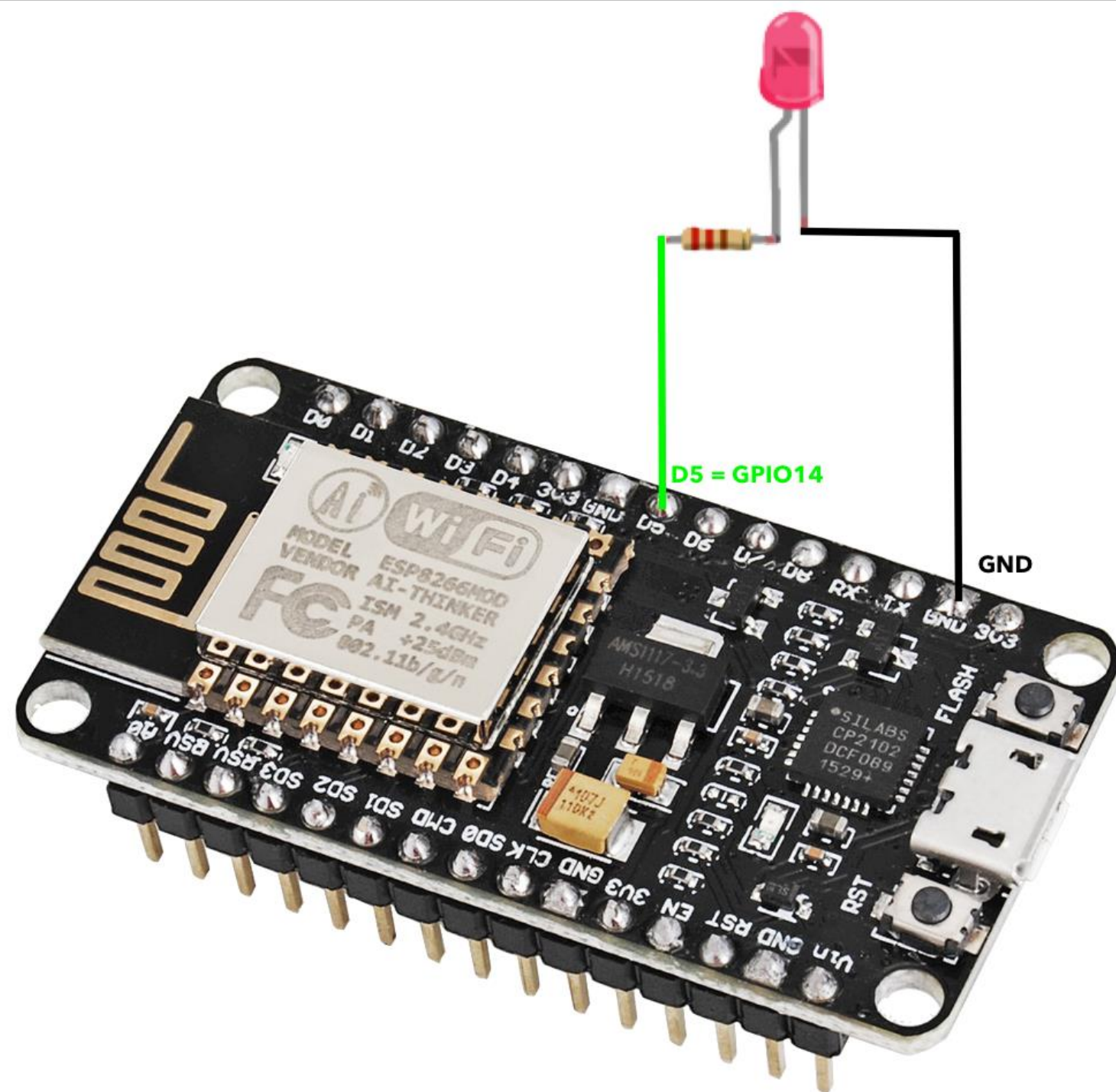
Constantes que já estão predefinidas



Constante	Valor
D0	16
D1	5
D2	4
D3	0
D4	2
D5	14
D6	12
D7	13
D8	15
A0	17



Exemplo Blink



```
//O led está no GPIO14
```

```
#define LED 14
```

```
//ou usar a constante D5 que já está definida
```

```
//#define LED D5
```

```
void setup()
```

```
{
```

```
    pinMode(LED, OUTPUT);
```

```
}
```

```
void loop()
```

```
{
```

```
    digitalWrite(LED, HIGH);
```

```
    delay(1000);
```

```
    digitalWrite(LED, LOW);
```

```
    delay(1000);
```

```
}
```

INPUT / OUTPUT

Ao realizar testes de INPUT e OUTPUT nos pinos, obtivemos os seguintes resultados:

digitalWrite NÃO funcionou com os GPIOs **6, 7, 8, 11** e o ADC (A0)

digitalRead NÃO funcionou com os GPIOs **1, 3, 6, 7, 8, 11** e o ADC (A0)

analogWrite NÃO funcionou com os GPIOs **6, 7, 8, 11** e o ADC (A0) (Os GPIOs **4, 12, 14, 15** possuem PWM por hardware os demais por software)

analogRead funcionou apenas com o ADC (A0)

6, 7, 8, 11 não funcionam para os quatro comandos acima



Em www.fernandok.com

Download arquivos PDF e **INO** do código fonte

