

Esse artigo foi escrito pelo engenheiro Ismael Lopes da Silva, exclusivamente para o site "[www.embarcados.com.br](http://www.embarcados.com.br)". O link para o artigo é "<https://www.embarcados.com.br/blue-pill-alimentacao-e-primeiros-passos-com-st-link/>".

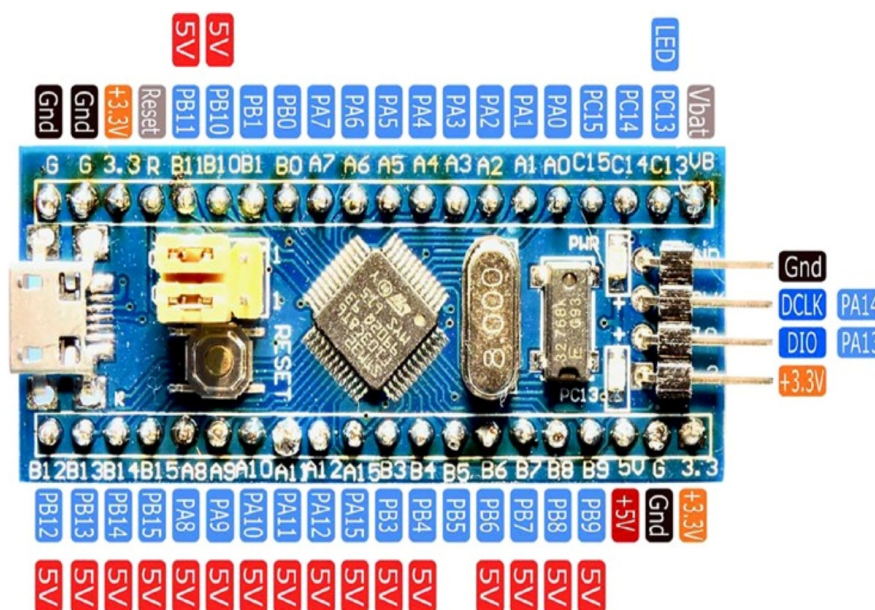
No quinto artigo da série "A Blue Pill", daremos continuidade no estudo. A dica é acompanhar a série desde o primeiro artigo porque é um caminho estruturado, onde a sequência traz benefícios no aprendizado (linha de raciocínio). Nesse artigo veremos sobre os terminais de alimentação, consumo, o regulador de tensão, o RESET e o programador ST-Link.

## Alimente o Blue Pill e o LED do usuário pisca

O Blue Pill que você comprou provavelmente já foi pré-programada para piscar quando é alimentado. Isso é um teste simples de que a placa está funcionando. Existem alguns outros detalhes importantes sobre consumo de energia, reset e LEDs que também serão visto nesse artigo. Depois, finalmente veremos o uso do programador ST-Link V2.

## Alimentação

A Blue Pill possui várias conexões, incluindo algumas para alimentação. Não é necessário usar todas as conexões de uma só vez. De fato, é melhor usar apenas um conjunto de conexões. Para esclarecer esse ponto, vamos começar com um exame de suas opções de energia. A figura 1 ilustra as conexões ao redor das bordas da placa de circuito impresso.



**Figura 1** – Terminais da Placa Blue Pill

Os quatro pinos na extremidade lateral direita da placa são usados para programar o dispositivo. Observe que a conexão de programação denominada DIO também é capaz de ser uma GPIO PA13. Da mesma forma, o DCLK é capaz de ser uma GPIO PA14. Sabemos detalhadamente como configurar o STM32. No conector de programação, observe que a tensão de entrada é de +3,3 Volts, cor laranja. Essa conexão eletricamente é igual a qualquer uma das outras conexões identificadas como "+ 3.3V" na PCB, portanto, estão interligadas.

## **Regulador de tensão**

O chip STM32F103C8T6 foi projetado para operar com qualquer Voltagem entre 2 e 3,3 Volts. A PCB Blue Pill tem um pequeno regulador de + 3,3 Volts com o rótulo "U1". O esquema oficial da placa Blue Pill especifica o regulador RT9193-33, que suporta 300mA. Hoje temos no mercado PCB Blue Pill clone usando um regulador mais barato, regulador da série XC6204, que suporta apenas 150mA.

A menos que você conheça bem sua PCB Blue Pill, é mais seguro assumir que sua placa suporta apenas 150mA. Veremos o desempenho de energia do MCU, em detalhes, em outros artigos. Somente como referência inicial, o programa que vem no dispositivo, que pisca o LED, utiliza cerca de 30mA, medido na fonte de entrada de + 5 Volts. Essa medição inclui a pequena corrente adicional usada pelo próprio regulador de tensão.

A folha de dados do STM32F103C8T6 documenta que seu consumo máximo é de 50mA. Essa medição é obtida com clock externo e todos os periféricos ativados, operando no modo "run" a 72MHz. Subtraindo 50mA do regulador que no máximo suporta 150mA, resta apenas 100mA, a ser consumido na saída do regulador de + 3,3 Volts. É sempre bom ficar atento quanto aos limites!

## **Alimentação de +5 Volts da USB**

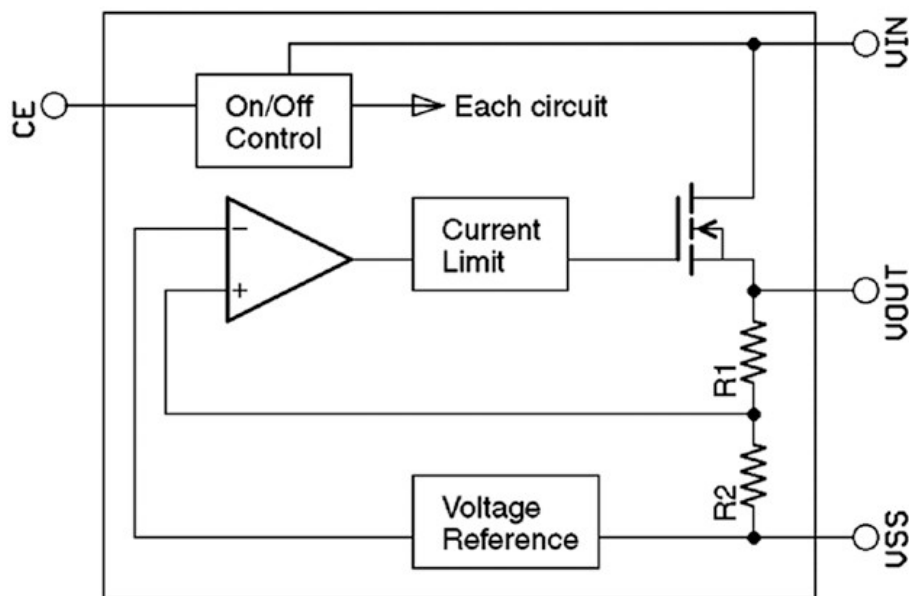
Quando alimentado por um cabo USB, a energia chega pelo conector Micro-USB B. Este alimentação de + 5 Volts é regulada nos +3,3 Volts necessários ao MCU. No canto inferior direito, da figura 1, há um pino identificado como "+ 5V", com um sinal de mais, que pode ser usado como um entrada de energia. Esse pino é interligado com a entrada do regulador, mesmo ponto que o conector USB fornece os 5 Volts.

Devido o baixo consumo de corrente do MCU, você também pode alimentá-lo com um módulo adaptador serial TTL. Muitos adaptadores seriais USB terão +5 Volts disponível que poderia alimentar o MCU. Sempre verifique o seu adaptador serial para obter especificações corretas. Cuidado para alimentar a placa Blue Pill através de duas fontes simultaneamente. Isso causará danos ao seu desktop/Notebook, através do cabo USB.

## **Alimentação de +3,3 Volts**

Se você tiver uma fonte de alimentação de + 3,3 Volts, poderá deixar as entradas de +5V desconectadas. Conecte sua fonte de alimentação de + 3,3 Volts diretamente à entrada de +3,3V (verifique se o cabo USB está desconectado). Isso funciona porque o regulador fica desativado quando não há entrada fornecido na entrada de 5 Volts.

Ao fornecer alimentação diretamente para a entrada de + 3,3 Volts, você está conectando ao terminal VOUT do regulador, ilustrado na figura 2. Nesse caso, não há alimentação de 5 Volts no terminal VIN do regulador. O terminal CE também está conectado do terminal VIN, mas, quando o VIN é desconectado, o pino CE fica aterrado por um capacitor. Um nível baixo no terminal CE desabilita os subsistemas internos do regulador.



**Figura 2** - Diagrama de blocos do regulador interno de +3,3 Volts

No entanto, há uma pequena quantidade de fluxo de corrente no divisor de tensão do regulador. Essa corrente fluirá de seus +3,3 Volts para o terra (VSS), através das resistências internas R1 e R2 do regulador. Essas resistências são altas e a corrente envolvida é insignificante. Mas esteja ciente disso ao medir a corrente para aplicações de bateria com consumo de energia muito baixo.

### **ATENÇÃO!**

Não forneça +5 Volts e +3,3 Volts ao mesmo tempo. Isso poderia causar danos ao regulador ou ao seu Desktop/Notebook quando o cabo USB estiver conectado. Use uma única fonte de alimentação.

### **Regra de uma fonte de alimentação**

O conselho é usar apenas uma fonte de alimentação. Alimentar sua PCB com mais de uma fonte pode causar danos. Reforço bem esse aspecto porque alimentamos a placa com +3,3 Volts ou +5 Volts, e depois usamos o cabo USB, então, isso pode causar danos.

Certas aplicações podem exigir o uso de alimentação adicional; por exemplo, quando acionamos motores ou relés. Nesses casos, você alimenta os circuitos externos com uma fonte adequada, mas, não pela PCB do MCU. Somente a referência GND precisa compartilhar conexões (ser comum).

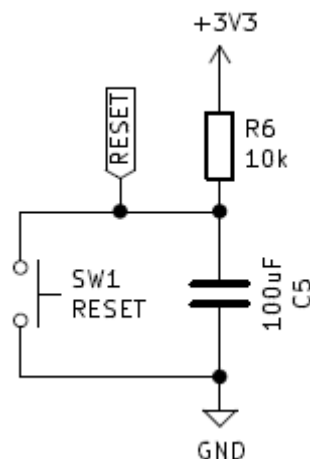
### **Potencial de referência (Ground)**

O pólo negativo da fonte de alimentação é conhecido como terminal GND. A figura 1 ilustra esses terminais na cor preto. Todas essas conexões ao GND estão eletricamente interligadas.

### **RESET**

A PCB Blue Pill tem um botão rotulado como "RESET", e uma conexão ao lado, rotulada como

"R". Essa conexão permite que um circuito externo reset o MCU. A figura 3 ilustra o circuito do botão, incluindo a conexão que vai ao MCU.



**Figura 3** – Esquema elétrico do reset

Pelo circuito de reset podemos ver que o MCU é resetado quando um nível baixo é aplicado no terminal de reset. O circuito RC garante que ao ligar a fonte de +3,3V o reset será executado.

### **Vamos alimentar nossa Blue Pill pela primeira vez**

A maneira mais segura e fácil de alimentar sua Blue Pill é usando um cabo USB com um conector Micro-USB B. Conecte seu cabo a uma fonte de alimentação USB, que necessariamente não precisa ser um computador. Uma vez ligado, o LED da sua PCB deve piscar. Caso contrário, tente pressionar o botão Reset. Verifique também se os jumpers boot-0 e boot-1 estão posicionados como ilustra a figura 1, os dois jumpers devem estar posicionados no lado rotulado como "0".

Existem dois LEDs na PCB Blue Pill. O LED rotulado como "PWR" indica que a alimentação foi aplicada na placa. O LED rotulado como "PC13" é ativado pela porta GPIO PC13, sob controle da aplicação.

Se o LED não piscar, após ter alimentado a placa, então, com um multímetro verifica se tem +3,3 Volts, entre o terminal GND e o terminal +3.3V. Se não houver alimentação, então, o cabo USB pode estar com algum problema. Se você não possui um cabo USB adequado, pode alimentar a PCB até mesmo com um par pilhas de 1,5 Volts em série. Lembre-se de que este MCU funcionará com alimentação entre 2 Volts e 3,3 Volts.

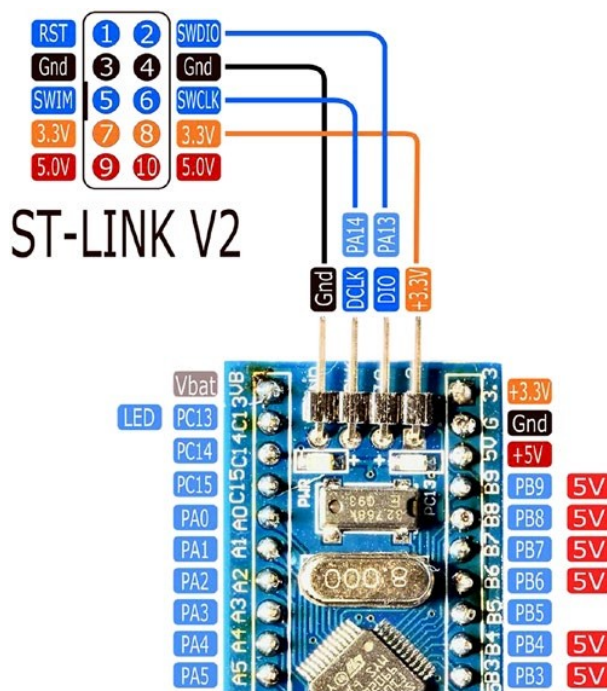
Se você tem uma fonte de alimentação de bancada, então, poderá alimentar através dos terminais + 3,3 Volts e GND. Tenha cuidado ao usar garras de jacaré, garantindo que elas não encostem em outros pinos. Os fios da jumper podem ser usados com maior segurança.

### **O ST-Link V2**

Agora veremos como trabalhar com o programador ST-Link V2. Quando você adquiriu seu programador, provavelmente já vem com os cabos adequados para interligá-lo com seu Desktop/ Notebook e também com a placa Blue Pill.

O diagrama de conexão do programador é ilustrado na figura 4. Diferentes modelos do programador estão disponíveis no mercado, usando diferentes conexões e fiação.

Com o programador ST-Link V2 conectado de acordo com a ilustração da figura 4, verifique os jumpers boot-0 e boot-1, localizados ao lado do botão de Reset, eles devem aparecer como ilustrado na figura 1, com os dois jumpers próximos ao lado marcado com "0".



**Figura 4** – Interligação do ST-Link V2 e Blue Pill

Conecte seu programador ST-Link V2 a uma porta USB. Depois de fazer isso, o LED de "PWR" deve acender imediatamente. Além disso, o LED PC13 também deve piscar se sua Blue Pill ainda tiver o programa padrão.

No seu desktop/Notebook, execute o comando conforme mostrado a seguir:

```
~$ st-info -- probe
```

```
Found 1 stlink programmers
serial: 383f6e064e57343524592143
openocd: "\"\\x38\\x3f\\x6e\\x06\\x4e\\x57\\x34\\x35\\x24\\x59\\x21\\x43\""
flash: 65536 (pagesize: 1024)
sram: 20480
chipid: 0x0410
descr: F1 Medium-density device
```

O comando `st-info` encontrou seu programador ST-Link V2 e o STM32F103C8T6 conectado a ele. Observe que o número de série da CPU é relatado junto com a SRAM (20K). Quantidade da memória flash deveria ser 128K, mas você pode ver 64K. Provavelmente suporte 128K de qualquer maneira.

## O utilitário st-flash

Agora veremos como você pode usar o utilitário st-flash para ler, escrever ou apagar o programa no seu dispositivo STM32.

### Ler um programa

Salvar o conteúdo da memória do seu dispositivo em um arquivo permitirá restaurar o programa original, caso precise mais tarde. O exemplo a seguir lê a memória flash do Blue Pill, iniciando no endereço 0x80000000, e salva 0x1000 (4K) de dados em um arquivo chamado saved.img. A convenção de prefixo 0 da programação C será usado para indicar números hexadecimais.

```
~$ st-flash read ./saved.img 0x80000000 0x1000
```

```
st-flash 1.5.1-45-g393e942
2020-02-22T18:48:44 INFO common.c: Loading device parameters...
2020-02-22T18:48:44 INFO common.c: Device connected is: F1 Medium-density device, id
0x20036410
2020-02-22T18:48:44 INFO common.c: SRAM size: 0x5000 bytes (20 KiB), Flash: 0x10000 bytes (64
KiB) in pages of 1024 bytes
```

Podemos verificar que o arquivo foi criado.

```
~$ ls
```

Desktop Music Public stm32f103c8t6 Videos Documents Downloads Pictures **saved.img** stlink  
Templates

### Escrever um programa

Escrever memória flash é o inverso da leitura. Uma imagem de memória salva pode ser “flashed” usando o subcomando write do utilitário st-flash. Observe que omitimos o tamanho do arquivo de dados. Neste exemplo, escrevemos de Volta no mesmo endereço:

```
~$ st-flash write ./saved.img 0x80000000
```

```
st-flash 1.5.1-45-g393e942
2020-02-23T08:50:24 INFO usb.c: -- exit_dfu_mode
2020-02-23T08:50:25 INFO common.c: Loading device parameters...
2020-02-23T08:50:25 INFO common.c: Device connected is: F1 Medium-density device, id
0x20036410
2020-02-23T08:50:25 INFO common.c: SRAM size: 0x5000 bytes (20 KiB), Flash: 0x10000 bytes (64
KiB) in pages of 1024 bytes
2020-02-23T08:50:25 INFO common.c: Ignoring 544 bytes of 0xff at end of file
2020-02-23T08:50:25 INFO common.c: Attempting to write 3552 (0xde0) bytes to stm32 address:
134217728 (0x80000000)
Flash page at addr: 0x08000c00 erased
2020-02-23T08:50:25 INFO common.c: Finished erasing 4 pages of 1024 (0x400) bytes
```

```
2020-02-23T08:50:25 INFO common.c: Starting Flash write for VL/F0/F3/F1_XL core id
2020-02-23T08:50:25 INFO flash_loader.c: Successfully loaded flash loader in sram
4/4 pages written
2020-02-23T08:50:25 INFO common.c: Starting verification of write complete
2020-02-23T08:50:25 INFO common.c: Flash written and verified! jolly good!
```

Esta operação restaurará o arquivo de imagem salvo no seu Desktop/Notebook para a memória flash do seu Blue Pill. Nesse caso o LED começará a piscar imediatamente, caso contrário, pressione o botão RESET para forçar uma reinicialização.

## Apagar um programa

Pode haver momentos em que você deseja apagar totalmente a memória do MCU, então:

~\$ st-flash erase

```
st-flash 1.5.1-45-g393e942
2020-02-23T09:01:12 INFO common.c: Loading device parameters....
2020-02-23T09:01:12 INFO common.c: Device connected is: F1 Medium-density device, id
0x20036410
2020-02-23T09:01:12 INFO common.c: SRAM size: 0x5000 bytes (20 KiB), Flash: 0x10000 bytes (64
KiB) in pages of 1024 bytes
Mass erasing
```

Antes de apagar a memória flash, se o LED estava piscando, então, observe que após o comando erase o programa no MCU estará apagado, e o LED apaga, portanto, não pisca. Como salvamos o programa, então, para recuperá-lo, basta escrever novamente o arquivo saved.img para a memória do MCU, usando o comando write.

Conforme a proposta inicial estamos evoluindo no aprendizado, portanto, aqui concluo o quinto artigo da série.