

Kit de desenvolvimento DOIT Esp32 v1

O DOIT Esp32 DevKit v1 é uma das placas de desenvolvimento criadas pelo DOIT para avaliar o módulo ESP-WROOM-32. É baseado no [microcontrolador ESP32](#) que possui suporte para Wifi, Bluetooth, Ethernet e Low Power, tudo em um único chip.



Mapeamento de pinos



DOIT ESP32 DevKit v1

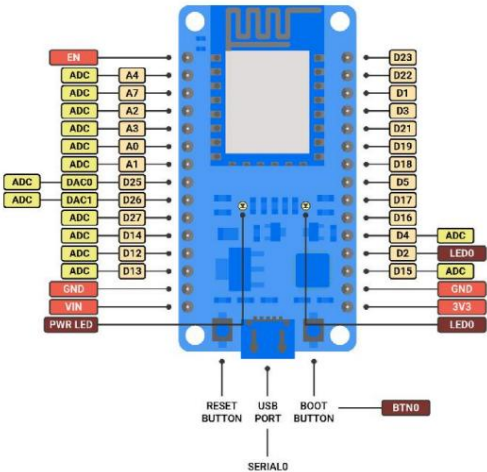


DO NOT USE D6 TO D11

PWM IS ENABLED ON EVERY DIGITAL PIN

ICU NOT SUPPORTED

ADC ON PINS D4, D12, D13, D14, D15, D25, D26, D27
CAN BE READ ONLY WITH WI-FI NOT STARTED



Mais informações sobre o DOIT Esp32 DevKit v1 podem ser encontradas [aqui](#).

Disposição Flash

O flash interno do módulo ESP32 está organizado em uma única área flash com páginas de 4096 bytes cada. O flash começa no endereço 0x000000, mas muitas áreas são reservadas para Esp32 IDF SDK e Zerynth VM. Existem dois layouts diferentes com base na presença de suporte BLE.

Em particular, para VMs não BLE:

Tamanho do endereço inicial	Conteúdo
0x00009000 16Kb Esp32 NVS área 0x0000D000	
8Kb Esp32 OTA dados 0x0000F000 4Kb Esp32	
PHY dados 0x00010000 1Mb Zerynth VM	
0x00110000 1Mb Zerynth VM (FOTA)	
0x00210000 5 Bytecode Zerynth de 12 KB 0x00290000	
Bytecode Zerynth de 512 KB (FOTA) 0x00310000	
512 KB grátis para armazenamento do usuário	
0x00390000 448 KB reservados	

Para VMs BLE:

Tamanho do endereço inicial	Conteúdo
0x00009000 16Kb Esp32 NVS área 0x0000D000	
8Kb Esp32 OTA dados 0x0000F000 4Kb Esp32	
PHY dados 0x00010000 1216Kb Zerynth VM	
0x00140000 1216Kb Zerynth VM (FOTA)	
0x0027 0000 Bytecode Zerynth de 320 KB	
0x002C0000 Bytecode Zerynth de 320 KB	
(FOTA) 0x00310000 512 KB grátis para armazenamento	
do usuário 0x00390000 448 KB reservados	

Resumo do dispositivo

- Microcontrolador: CPU Tensilica de 32 bits de núcleo único/dual Xtensa LX6 •
- Tensão operacional: 3,3V •
- Tensão de entrada: 7-12V •
- Pinos de E/S digitais (DIO): 25 •
- Pinos de entrada analógica (ADC): 6
- Pinos de saída analógica (DAC): 2 •
- UARTs: 3 •
- SPIs: 2 •
- I2Cs: 3 •
- Memória Flash: 4 MB •
- SRAM: 520 KB •
- Velocidade de clock: 240 Mhz
- Wi-Fi: IEEE 802.11 b/g/n/ e/i: o
- Switch TR integrado, balun, LNA, amplificador de potência e rede correspondente o
- Autenticação WEP ou WPA/WPA2, ou redes abertas

Poder

A energia para o DOIT Esp32 DevKit v1 é fornecida através do conector USB Micro B integrado ou diretamente através do pino “VIN”. A fonte de alimentação é selecionada automaticamente.

O dispositivo pode operar com alimentação externa de 6 a 20 volts. Se usar mais de 12 V, o regulador de tensão pode superaquecer e danificar o dispositivo. A faixa recomendada é de 7 a 12 volts.

Conecte, registre, virtualize e programe

O DOIT Esp32 DevKit v1 vem com um chip serial para USB integrado que permite programar e abrir o UART do módulo ESP32. Os drivers podem ser necessários dependendo do seu sistema (Mac ou Windows) e podem ser baixados na página [de documentação oficial do Espressif](#) . Em sistemas Linux, o [DevKit v1](#) deve funcionar imediatamente.

Observação

Para plataforma Linux: para permitir o acesso às portas seriais o usuário precisa de acesso de leitura/gravação ao arquivo do dispositivo serial. Adicionar o usuário ao grupo que possui este arquivo fornece o acesso de leitura/gravação necessário:

- Distribuição **Ubuntu** -> grupo dialout
- Distribuição **Arch Linux** -> grupo uucp

Uma vez conectado em uma porta USB, se os drivers tiverem sido instalados corretamente, o dispositivo DevKit v1 será reconhecido pelo Zerynth Studio. As próximas etapas são:

- **Selecione** o DevKit v1 na **barra de ferramentas de gerenciamento de dispositivos** (desambiguar se necessário);
- **Cadastre** o dispositivo clicando no botão “Z” do Zerynth Studio;
- **Crie** uma Máquina Virtual para o dispositivo clicando uma segunda vez no botão “Z”;
- **Virtualize** o dispositivo clicando no botão “Z” pela terceira vez.

Observação

Nenhuma intervenção do usuário no dispositivo é necessária para o processo de registro e virtualização

Após a virtualização, o DevKit v1 está pronto para ser programado e os scripts Zerynth **carregados**. Basta **selecionar** o dispositivo virtualizado na “Barra de ferramentas de gerenciamento de dispositivos” e **clicar** no botão “upload” dedicado do Zerynth Studio.

Observação

Nenhuma intervenção do usuário no dispositivo é necessária para o processo de uplink.

Atualização de firmware pelo ar (FOTA)

O recurso Firmware Over the Air permite atualizar o firmware do dispositivo em tempo de execução. Zerynth FOTA no dispositivo DevKitC está disponível para bytecode e VM.

O Layout Flash é mostrado na tabela abaixo:

Tamanho do endereço inicial	Conteúdo
0x00010000 VM Zerynth de 1 MB (slot 0)	
0x00110000 VM Zerynth de 1 MB (slot 1)	
0x00210000 Bytecode Zerynth de 512 KB (slot 0)	
0x00290000 Bytecode Zerynth de 512 KB (slot 1)	

Para VMs BLE:

Tamanho do endereço inicial	Conteúdo
0x00010000 VM Zerynth de 1216 KB (slot 0)	
0x00140000 VM Zerynth de 1216 KB (slot 1)	

Tamanho do endereço inicial	Conteúdo
0x00270000	Bytecode Zerynth de 320 KB (slot 0)
0x002C0000	Bytecode Zerynth de 320 KB (slot 1)

Para dispositivos baseados em Esp32, o processo FOTA é implementado principalmente usando as chamadas de sistema fornecidas na estrutura IDF. A seleção da próxima VM a ser executada é, portanto, uma responsabilidade do bootloader Espressif; o bootloader, entretanto, não fornece um mecanismo à prova de falhas para reverter para a VM anterior caso a atualmente selecionada falhe ao iniciar. No momento esta falta de recurso de segurança não pode ser contornada, a não ser alterando o bootloader. Assim que a Espressif lançar um novo IDF com tal recurso, lançaremos VMs atualizadas.

Firmware seguro

O recurso Secure Firmware permite detectar e recuperar falhas e, quando suportado, proteger o firmware em execução (por exemplo, desabilitando o acesso externo ao flash ou atribuindo memória RAM protegida a partes críticas do sistema).

Este recurso depende fortemente da plataforma; mais informações na [seção Firmware Seguro - ESP32](#).

Soquete seguro Zerynth

Para poder usar Zerynth Secure Socket em placas esp32 `NATIVE_MBEDTLS: true` deve ser usado em vez de `ZERYNTH_SSL: true` no arquivo `project.yml`.

Recursos ausentes

Nem todos os recursos IDF foram incluídos nas VMs baseadas em Esp32. Em particular, faltam os seguintes, mas serão adicionados num futuro próximo:

- Suporte para detecção de toque