Guia de Utilização da ESP-WROOM-32

Para utilizar com a IDE do Arduino:
https://github.com/espressif/arduino-esp32/tree/master/docs/arduino-ide

Datasheets:
Módulo:
https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf
Placa:
https://www.espressif.com/sites/default/files/documentation/esp-wroom-32_datasheet_en.pdf

Utilizar os cores:
https://techtutorialsx.com/2017/05/09/esp32-running-code-on-a-specific-core/

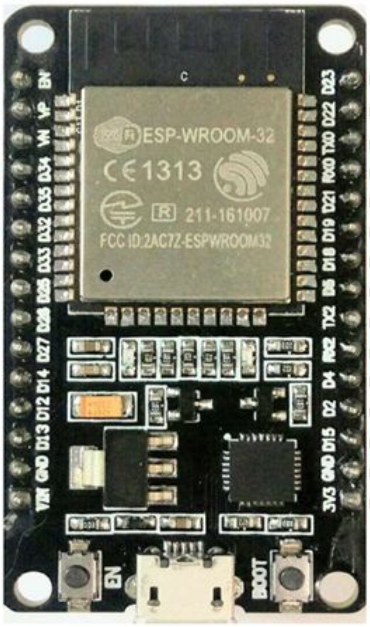
http://esp32.info/docs/esp_idf/html/dd/d3c/group__xTaskCreate.html

É possível criar threads e deixar o escalonador trabalhar. É possível definir a prioridade da task na criação da thread, quanto maior o número, maior a prioridade.
É possível atrelar a execução de uma thread a um dos 2 cores.

Pinout

NodeMCU - datalogger:
https://portal.vidadesilicio.com.br/modulo-cartao-micro-sd-nodemcu-datalogger/

SP32 DEVKIT V1 PINOUT



			Chip-enable signal,Active High.				EN	pin15
			ADC PA	RTC_GPIO0	ADC1_CH0	SENSOR_VP	GPIO36	pin14
			ADC PA	RTC_GPIO3	ADC1_CH3	SENSOR_VN	GPIO39	pin13
				RTC_GPIO4	ADC1_CH6	VDET1	GPIO34	pin12
				RTC_GPIO5	ADC1_CH7	VDET2	GPIO35	pin11
		XTAL_32kHz	Touch9	RTC_GPIO9	ADC1_CH4		GPIO32	pin10
		XTAL_32kHz	Touch8	RTC_GPIO8	ADC1_CH5		GPIO33	pin9
			DAC_1	RTC_GPIO6	ADC2_CH8	EMAC_RXD0	GPIO25	pin8
			DAC_2	RTC_GPIO7	ADC2_CH9	EMAC_RXD1	GPIO26	pin7
			Touch7	RTC_GPIO17	ADC2_CH7	EMAC_RX_DV	GPIO27	pin6
	HSPI_CLK	MTMS	Touch6	RTC_GPIO16	ADC2_CH6	EMAC_TXD2	GPIO14	pin5
2	HSPI_MISO	MTDI	Touch5	RTC_GPIO15	ADC2_CH5	EMAC_TXD3	GPIO12	pin4
3	HSPI_MOSI	MTCK	Touch4	RTC_GPIO14	ADC2_CH4	EMAC_RX_ER	GPIO13	pin3
							GND	pin2
							VIN	pin1

pin15	GPIO23		SPI_MOSI	HS1_STROBE				
pin14	GPIO22	EMAC_TXD1	U0RTS	I2C_SCL				
pin13	GPIO1	EMAC_RXD2	U0TXD	CLK OUT3				
pin12	GPIO3		U0RXD	CLK OUT2				
pin11	GPIO21	EMAC_TX_EN		I2C_SDA				
pin10	GPIO19	EMAC_TXD0	U0CTS	SPI_MISO				
pin9	GPIO18		SPI_CLK	HS1_DATA7				
pin8	GPIO5	EMAC_RX_CLK	SPI_CS0	HS1_DATA6				
pin7	GPIO17	EMAC_CLKOUT180	U2_TXD	HS1_DATA5				
pin6	GPIO16	EMAC_CLKOUT	U2_RXD	HS1_DATA4				
pin5	GPIO4	EMAC_TX_ER	ADC2_CH0	RTCIO10	Touch0		HSPIHD	SD_DATA1
pin4	GPIO2		ADC2_CH2	RTCIO12	Touch2		HSPIWP	
pin3	GPIO15	EMAC_RXD3	ADC2_CH3	RTCIO13	Touch3	MTDO	HSPI_CS0	SD_CMD
pin2	GND							
pin1	VDD 3V3							

Consumo de energia nos diferentes modos:

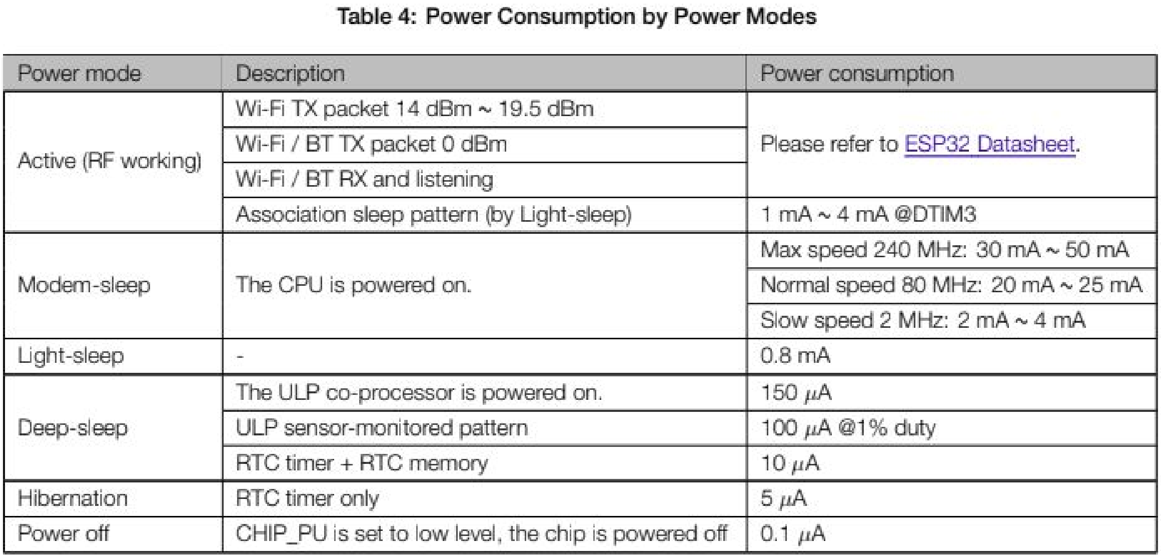


Table 4: Power Consumption by Power Modes

Power mode	Description	Power consumption
Active (RF working)	Wi-Fi TX packet 14 dBm ~ 19.5 dBm	Please refer to ESP32 Datasheet.
	Wi-Fi / BT TX packet 0 dBm	
	Wi-Fi / BT RX and listening	
	Association sleep pattern (by Light-sleep)	1 mA ~ 4 mA @DTIM3
Modem-sleep	The CPU is powered on.	Max speed 240 MHz: 30 mA ~ 50 mA
		Normal speed 80 MHz: 20 mA ~ 25 mA
		Slow speed 2 MHz: 2 mA ~ 4 mA
Light-sleep	-	0.8 mA
Deep-sleep	The ULP co-processor is powered on.	150 μA
	ULP sensor-monitored pattern	100 μA @1% duty
	RTC timer + RTC memory	10 μA
Hibernation	RTC timer only	5 μA
Power off	CHIP_PU is set to low level, the chip is powered off	0.1 μA

Modificar a partição da ESP32:

I have had success using the following default.csv based off of minimal.csv:

```
# Name,   Type, SubType, Offset,   Size, Flags
nvs,      data, nvs,     0x9000,   0x5000,
otadata,  data, ota,     0xe000,   0x2000,
app0,     app,  ota_0,   0x10000,  0x340000,
eeprom,   data, 0x99,    0x350000, 0x1000,
spiffs,   data, spiffs,  0x351000, 0xAF000,
```

and boards.txt to have modified values of `.upload.maximum_size=3407872`

Wasn't sure if we can omit the otadata or spiffs partitions altogether if we aren't directly using those features.

Tutorial: https://desire.giesecke.tk/index.php/2018/01/30/change-partition-size/