

A. Berkenalan dengan ESP32

Memahami sedikit seluk beluk apa itu ESP32

- [Salam kenal, aku ESP 32](#)
- [Spesifikasi ESP32](#)
- [ESP 32 pinout](#)

Salam kenal, aku ESP 32

dibuat untuk tugas mata kuliah EL5057 Sistem Penginderaan

Halo teman-teman, mari berkenalan dengan mikrokontroler **ESP32**. jadi, apasih sebenarnya ESP32? ESP 32 adalah mikrokontroler serbaguna yang memiliki dua inti prosesor untuk kinerja yang cepat dan efisien. Dengan kemampuan WiFi dan Bluetooth terintegrasi, ESP32 memungkinkan pengembang untuk membuat proyek-proyek IoT yang terhubung ke internet tanpa perlu tambahan perangkat keras. Kelebihan ESP32 terletak pada fleksibilitasnya; dapat diprogram menggunakan Arduino IDE, MicroPython, atau bahasa pemrograman lainnya. Dengan dukungan sensor yang beragam, ESP32 memungkinkan pembuatan proyek-proyek yang mencakup pengukuran suhu, kendali perangkat, dan pengiriman data ke cloud. Kesederhanaan penggunaan dan ukurannya yang kecil membuat ESP32 menjadi pilihan populer untuk berbagai aplikasi, mulai dari perangkat pintar hingga robotika, memberikan para pengembang alat yang kuat untuk mewujudkan ide-ide kreatif mereka dengan mudah.

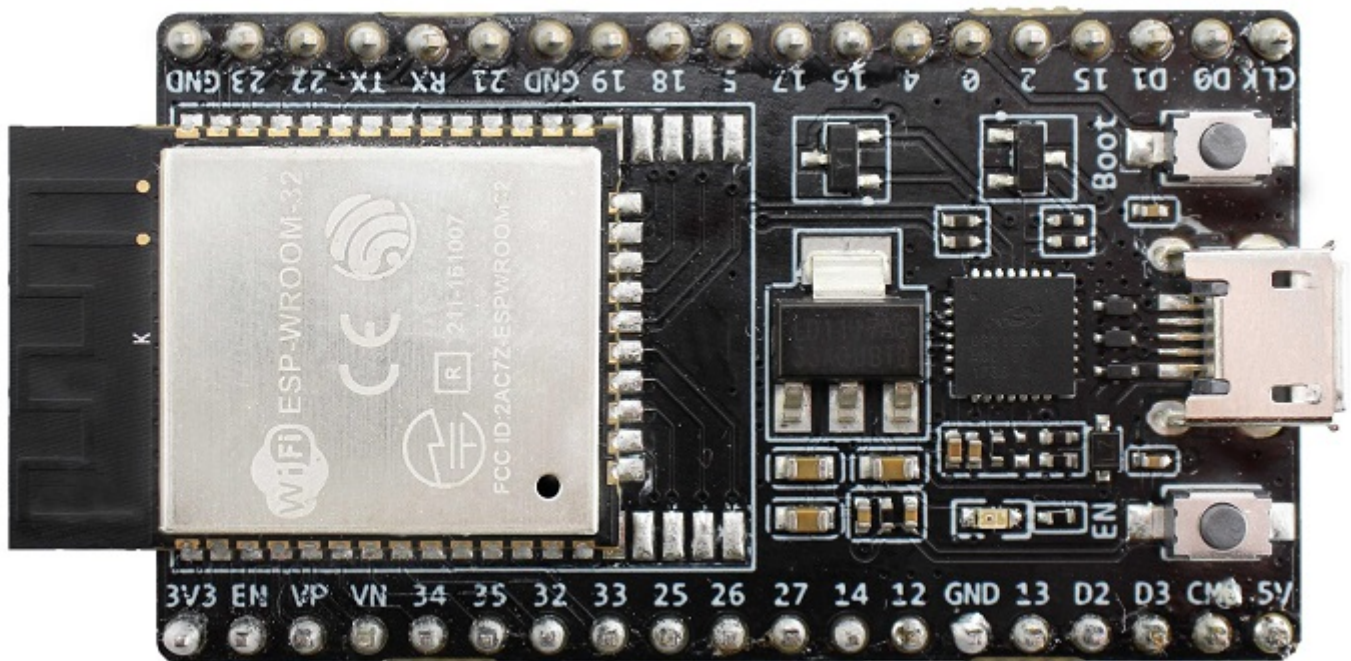


Jadi, bayangkan ESP32 seperti otak kecil yang super pintar untuk proyek-proyek elektronik kamu. Ini adalah mikrokontroler canggih yang bisa melakukan banyak hal seru. Jadi, apa yang bikin ESP32 istimewa?

1. **Dua dalam Satu:** ESP32 sebenarnya seperti paket hemat, karena seolah-olah kamu mendapatkan dua mikrokontroler dalam satu! Ada dua inti prosesor di dalamnya, yang artinya ESP32 bisa melakukan beberapa tugas sekaligus tanpa jadi lelet.
2. **WiFi dan Bluetooth:** Nah, ESP32 juga punya kemampuan WiFi dan Bluetooth bawaan. Bayangkan kamu bisa membuat gadget yang bisa terhubung ke internet tanpa perlu mikrokontroler ekstra. Seru, kan?
3. **Luar Biasa Ringkas:** Meskipun kecil, ESP32 itu luar biasa hebat. Kamu bisa pakai ini untuk buat perangkat pintar, seperti sensor suhu yang bisa dikontrol dari jauh, atau mungkin robot mini yang bisa diarahkan pakai smartphone.

4. **Sahabat Programmer:** ESP32 juga sangat ramah buat para programmer. Kamu bisa pakai Arduino IDE atau bahkan MicroPython untuk membuatnya. Jadi, meskipun kamu bukan ahli, kamu bisa tetap membuat proyek seru dengan ESP32.
5. **Banyak Fitur Keren:** Sensor, koneksi ke internet, kemampuan pengolahannya, semuanya lengkap di ESP32. Kamu bisa mengukur suhu, mengontrol perangkat, bahkan memonitor data dan mengirimnya ke cloud. Gila, kan?

Jadi, singkatnya, ESP32 itu seperti sihir untuk dunia elektronik. Kamu bisa berkreasi tanpa batas dan membuat proyek-proyek seru dengan mudah. Nah, biasanya ESP32 yang dijual di pasaran sudah termasuk dengan *development board* nya agar kita bisa langsung pakai untuk pemrograman. Contoh *dev-board* nya adalah sebagai berikut.

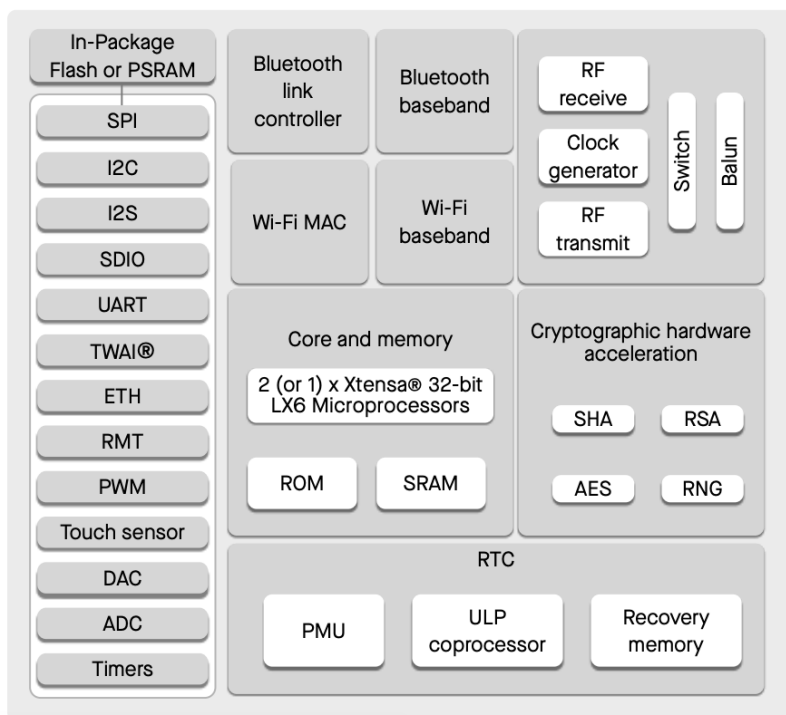


Spesifikasi ESP32

dibuat untuk tugas mata kuliah EL5057 Sistem Penginderaan

Mari mengenal lebih detail ESP32 ini, dari namanya sih sepertinya singkatan dari nama perusahaan yang membungkus chipset ini kedalam satu chip yaitu ESPRESSIF. Espressif Systems adalah perusahaan semikonduktor yang berkantor pusat di Tiongkok, yang terkenal dengan pengembangan mikrokontroler, khususnya seri ESP8266 dan ESP32 yang sangat populer di komunitas elektronika dan *Internet of Things* (IoT). Didirikan pada tahun 2008, Espressif fokus pada inovasi dalam desain chip nirkabel dan sistem chip terintegrasi untuk aplikasi WiFi dan Bluetooth. Produk unggulan mereka, seperti ESP8266 dan ESP32, telah memainkan peran penting dalam mendorong pertumbuhan ekosistem IoT dan DIY (*Do It Yourself*).

Berdasarkan data dari laman resmi Espressif System tentang chipset ESP32 (<https://esp32.net>), berikut adalah blok fungsi (*function block*) lengkap dari ESP32 yang akan kita gunakan untuk belajar pemrograman mikrokontroler.



Nah, terlihat sangat lengkap sekali bukan, walaupun ukurannya sekecil kotak penghapus, ternyata kelengkapannya sangat mirip dengan personal computer yang kita punya. Berikut adalah fitur-fitur tambahan dari ESP32 yang sudah tertanam dalam SoC (*System on Chip*).

Wi-Fi

- 802.11b/g/n
 - 802.11n (2.4 GHz), up to 150 Mbps
 - WMM
 - TX/RX A-MPDU, RX A-MSDU
 - Immediate Block ACK
 - Defragmentation
 - Automatic Beacon monitoring (hardware TSF)
 - 4 × virtual Wi-Fi interfaces
 - Simultaneous support for Infrastructure Station, SoftAP, and Promiscuous modes Note that when ESP32 is in Station mode, performing a scan, the SoftAP channel will be changed.
 - Antenna diversity
-

Bluetooth®

- Compliant with Bluetooth v4.2 BR/EDR and Bluetooth LE specifications
 - Class-1, class-2 and class-3 transmitter without external power amplifier
 - Enhanced Power Control
 - +9 dBm transmitting power
 - NZIF receiver with -94 dBm Bluetooth LE sensitivity
 - Adaptive Frequency Hopping (AFH)
 - Standard HCI based on SDIO/SPI/UART
 - High-speed UART HCI, up to 4 Mbps
 - Bluetooth 4.2 BR/EDR and Bluetooth LE dual mode controller
 - Synchronous Connection-Oriented/Extended (SCO/eSCO)
 - CVSD and SBC for audio codec
 - Bluetooth Piconet and Scatternet
 - Multi-connections in Classic Bluetooth and Bluetooth LE
 - Simultaneous advertising and scanning
-

CPU and Memory

- Xtensa® single-/dual-core 32-bit LX6 microprocessor(s)
 - CoreMark® score:
 - 1 core at 240 MHz: 504.85 CoreMark; 2.10 CoreMark/MHz
 - 2 cores at 240 MHz: 994.26 CoreMark; 4.14 CoreMark/MHz
 - 448 KB ROM
 - 520 KB SRAM
 - 16 KB SRAM in RTC
 - QSPI supports multiple flash/SRAM chips
-

Clocks and Timers

- Internal 8 MHz oscillator with calibration
- Internal RC oscillator with calibration

- External 2 MHz ~ 60 MHz crystal oscillator (40 MHz only for Wi-Fi/Bluetooth functionality)
 - External 32 kHz crystal oscillator for RTC with calibration
 - Two timer groups, including 2 × 64-bit timers and 1 × main watchdog in each group
 - One RTC timer
 - RTC watchdog
-

Advanced Peripheral Interfaces

- 34 × programmable GPIOs
 - 5 strapping GPIOs
 - 6 input-only GPIOs
 - 6 GPIOs needed for in-package flash/PSRAM (ESP32-D0WDR2-V3, ESP32-U4WDH)
 - 12-bit SAR ADC up to 18 channels
 - 2 × 8-bit DAC • 10 × touch sensors
 - 4 × SPI
 - 2 × I2S
 - 2 × I2C
 - 3 × UART
 - 1 host (SD/eMMC/SDIO)
 - 1 slave (SDIO/SPI)
 - Ethernet MAC interface with dedicated DMA and IEEE 1588 support
 - TWAI®, compatible with ISO 11898-1 (CAN Specification 2.0)
 - RMT (TX/RX)
 - Motor PWM
 - LED PWM up to 16 channels
-

Power Management

- Fine-resolution power control through a selection of clock frequency, duty cycle, Wi-Fi operating modes, and individual power control of internal components
 - Five power modes designed for typical scenarios: Active, Modem-sleep, Light-sleep, Deep-sleep, Hibernation
 - Power consumption in *Deep-sleep* mode is **10 μA**
 - Ultra-Low-Power (ULP) coprocessors
 - RTC memory remains powered on in Deep-sleep mode
-

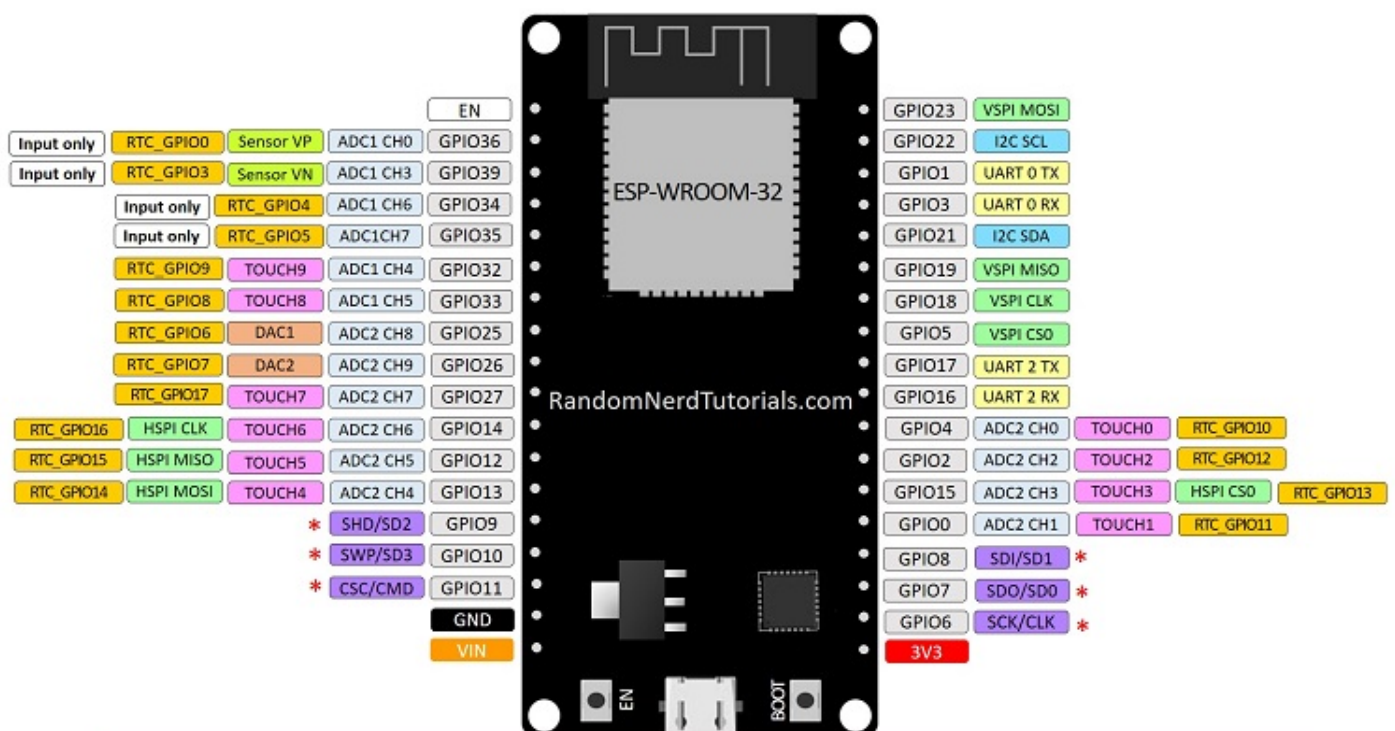
Nah, satu lagi saya tambahkan disini adalah data konsumsi daya dari **ESP32**. **Kenapa konsumsi daya penting?**, karena sistem IoT yang dipasang di lapangan biasanya menggunakan baterai sehingga dengan mendesain ESP32 agar irit daya, menjadikan alat IoT kita tahan lama. Berikut adalah tabel konsumsi daya dari ESP32.

Power mode	Description			Power Consumption
Active (RF working)	Wi-Fi Tx packet			Please refer to Table 4-4 for details.
	Wi-Fi/BT Tx packet			
	Wi-Fi/BT Rx and listening			
Modem-sleep	The CPU is powered up.	240 MHz [*]	Dual-core chip(s)	30 mA ~ 68 mA
			Single-core chip(s)	N/A
		160 MHz [*]	Dual-core chip(s)	27 mA ~ 44 mA
			Single-core chip(s)	27 mA ~ 34 mA
		Normal speed: 80 MHz	Dual-core chip(s)	20 mA ~ 31 mA
			Single-core chip(s)	20 mA ~ 25 mA
Light-sleep	-			0.8 mA
Deep-sleep	The ULP coprocessor is powered up.			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 down.			1 μA

ESP 32 pinout

ESP-32 Devkit V1 Pinout 30 Pin

ESP32 DEVKIT V1 – DOIT version with 36 GPIOs



* Pins SCK/CLK, SDO/SD0, SDI/SD1, SHD/SD2, SWP/SD3 and SCS/CMD, namely, GPIO6 to GPIO11 are connected to the integrated SPI flash integrated on ESP-WROOM-32 and are not recommended for other uses.

ESP-32 Devkit V1 Pinout 38 Pin

- PWM
- PIN NUMBER
- NAME
- GROUND
- POWER
- CONTROL
- I/O
- ADC
- COMM. INTERFACE
- DAC
- I2C
- HS
- TOUCH

