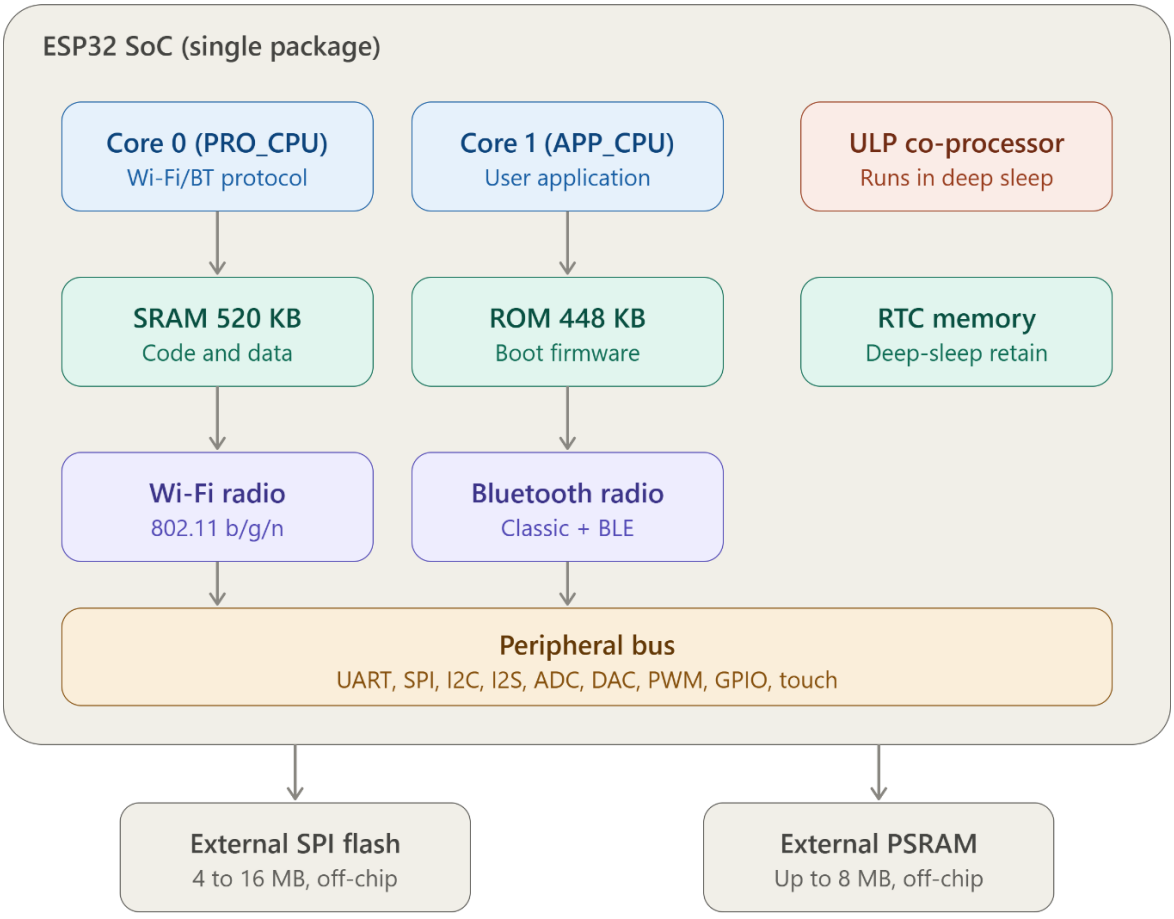
24EC504 – Unit I – Part 2: ESP32, STM32, Memory and Peripherals

1. ESP32 Microcontroller

ESP32 is a low-cost, low-power System-on-Chip from Espressif Systems that integrates a processor, Wi-Fi and Bluetooth radios, and many peripherals on one chip. Built-in wireless removes the need for an external module, so it is popular in IoT, home automation, wearables, industrial monitoring, robotics and biomedical sensor nodes.



1.1 Processor Architecture

- **Core:** 32-bit dual-core microprocessor, 80–240 MHz (adjustable with Dynamic Frequency Scaling). Some variants differ: ESP32-S2 is single-core and ESP32-C3 uses a RISC-V core.
- **Core 0 (PRO_CPU, Protocol CPU):** handles the Wi-Fi/Bluetooth stack and low-level radio work.

- **Core 1 (APP_CPU, Application CPU):** left free for user code, enabling parallel task execution with FreeRTOS (the ESP-IDF default). Tasks can be pinned to a core through FreeRTOS APIs.
- Each core has its own registers; both share the memory bus through arbitration.
- 7-stage pipeline and a Floating Point Unit (FPU) for signal-processing work.
- **ULP co-processor:** a small 16-bit RISC core that keeps running in deep sleep on a few microamps. It samples sensors (temperature, touch, ADC/RTC signals), evaluates simple conditions, and wakes the main CPU only on a significant event. This greatly extends battery life.

1.2 Power Management

- Power modes: Active, Modem-sleep, Light-sleep, Deep-sleep and Hibernation.
- In deep sleep only the RTC controller and ULP co-processor stay active, drawing low microamp-range current.

1.3 Memory

- **ROM:** 448 KB internal, for booting and core functions; cannot be modified.
- **SRAM:** 520 KB on-chip, in SRAM0, SRAM1 and SRAM2 regions, for instructions and data.
- **RTC memory:** small RTC FAST and RTC SLOW areas retain limited data in deep sleep, used by the ULP and for wake-up context.
- **eFuse:** one-time-programmable memory for MAC address, chip ID and security keys.
- **External SPI Flash:** the ESP32 has no built-in program Flash; it uses an external chip, typically 4 MB and up to 16 MB.
- **External PSRAM:** up to 8 MB for extra RAM, for example camera image buffers.
- **Cache controller:** a small instruction/data cache speeds access to the slower external Flash and PSRAM.
- Harvard-style memory map: instruction and data buses are addressed separately, though some SRAM areas are reachable by both.

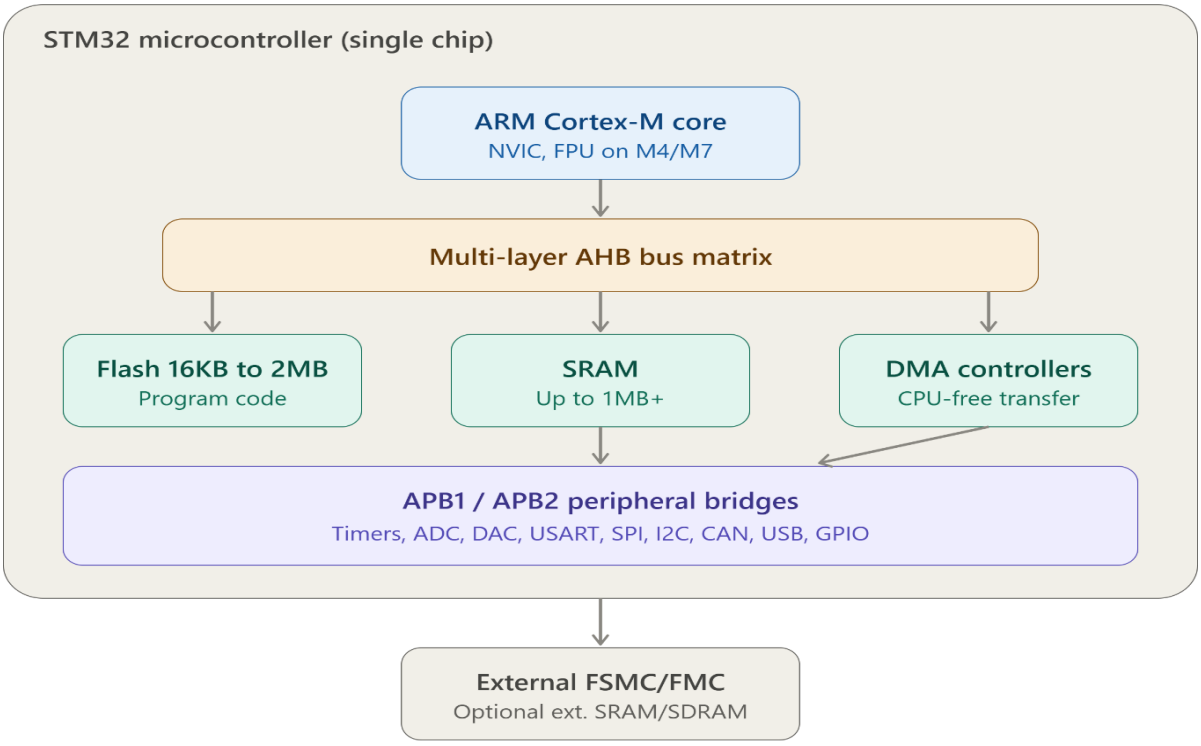
1.4 Peripherals

Group	Details
Wi-Fi	IEEE 802.11 b/g/n, 2.4 GHz; Station, SoftAP and Wi-Fi Direct
Bluetooth	Dual-mode: Classic (BR/EDR) and BLE
UART	3 hardware controllers
SPI	4 interfaces (Flash/PSRAM, displays)
I2C	2 controllers
I2S	2 interfaces for digital audio

Group	Details
CAN (TWAI)	Built-in controller for automotive/industrial use
SDIO	SD card interface
GPIO	Up to 36 pins, multiplexed with peripheral functions
ADC	18 channels, 12-bit SAR
DAC	2 channels, 8-bit
Touch sensor	10 capacitive touch pins
PWM	LED control / MCPWM channels for motors and dimming
Built-in sensors	Hall effect sensor and internal temperature sensor
Security	Hardware AES, SHA, RSA, ECC; Secure Boot; Flash Encryption

2. STM32 Microcontroller

STM32 is a family of 32-bit microcontrollers from STMicroelectronics built around ARM Cortex-M cores. Unlike ESP32 it has no built-in wireless (except a few parts such as STM32WB). It focuses on processing performance, rich peripherals, real-time control and industrial-grade reliability, so it is used in industrial automation, motor control, robotics, automotive electronics and medical instrumentation.



2.1 Processor Architecture

- **Core variants:** Cortex-M0/M0+ (entry level, F0/L0), Cortex-M3 (F1/F2), Cortex-M4 (F4/L4, with FPU and DSP instructions), Cortex-M7 (F7/H7, highest performance).
- **Clock:** a few MHz (ultra-low-power L-series) up to 480 MHz (H7, dual-core in some variants).
- RISC load-store architecture with a 3-stage pipeline (M3/M4) or superscalar 6-stage pipeline (M7).
- Harvard architecture with separate instruction and data buses and a bus matrix, so fetch and data access happen together.
- **NVIC:** up to 240 interrupt sources, configurable priorities, latency as low as 12 clock cycles.
- FPU and DSP extensions on M4/M7 for filtering, useful in biomedical signal conditioning.
- MPU in many variants for safety-critical and RTOS-based applications.

2.2 Power Management

- Low-power modes: Sleep, Stop and Standby, trading wake-up time against power.
- Ultra-low-power L-series parts target battery-operated portable medical and wearable devices.

2.3 Memory

- **On-chip Flash:** 16 KB to 2 MB for program code. Supports in-system programming (ISP) and in-application programming (IAP), so firmware can be updated without removing the chip. Some series have dual-bank Flash: one bank runs the application while the other is updated.
- **On-chip SRAM:** a few KB up to 1 MB or more (H7). Some variants add CCM (Core-Coupled Memory), directly coupled to the CPU for zero-wait-state access in time-critical DSP routines.
- **Memory map and bus matrix:** the standard Cortex-M memory map; a multi-layer AHB bus matrix connects the CPU, DMA, Flash and the APB1/APB2 peripheral buses so several masters can work without contention.
- **External memory:** FSMC/FMC on many variants for external SRAM, NOR/NAND Flash or SDRAM (for example display buffers).
- **Other:** system ROM with a factory bootloader for serial/USB programming, option bytes for read/write protection and boot configuration, and OTP memory for device IDs and calibration data.

2.4 Peripherals

- **Communication:** USART/UART (often 3–8), SPI, I2C, CAN/CAN-FD, USB (device, host, OTG), Ethernet MAC on higher-end series, SAI and SDIO.
- **Timers:** advanced-control, general-purpose and basic timers (commonly 8–17) for PWM, input capture, output compare and encoder interfacing. Advanced timers add dead-time insertion and break inputs for three-phase motor control.

- **RTC and watchdogs:** RTC with calendar on an independent power domain; Independent and Window watchdog timers.
- **ADC:** multiple 12-bit channels (up to 16-bit on some H7), very high sampling rates, scan mode, injected/regular groups, well suited to ECG/EEG acquisition.
- **DAC and analog:** one or more 12-bit DACs; comparators and op-amps built into some series.
- **DMA:** multiple controllers/streams move data between peripherals and memory without the CPU.
- **GPIO:** typically 40–140+ pins, each configurable as input, output, alternate function or analog.
- **Other:** CRC unit, random number generator, crypto accelerators on security-focused variants.

3. SRAM vs Flash in STM32

Aspect	SRAM	Flash
Volatility	Volatile: contents lost at power-off	Non-volatile: retained without power
Speed	High-speed working memory	Used mainly for code storage
Holds	Variables, stack, heap	Compiled firmware and constant data
Capacity	Tens to hundreds of KB	Larger

An STM32 boots by executing code directly from Flash while using SRAM as fast working memory at runtime.

4. GPIO

General Purpose Input/Output pins are programmable pins not tied to one fixed function. As an **input**, a pin reads a digital signal such as a push-button or sensor output. As an **output**, the processor drives it high or low to switch an LED or trigger a relay. GPIO is the most fundamental interface between a microcontroller and the physical world.

5. Watchdog Timer (COP)

A Watchdog Timer, also called a Computer Operating Properly (COP) timer, is a hardware timer that counts down and resets the whole microcontroller if it reaches zero. Healthy software must periodically "feed" (reset) it. If the program hangs because of a bug, infinite loop or glitch, the timer expires and forces a reset, so the device recovers without manual intervention. This is a critical safety feature for unattended or remote systems.

6. ESP32 vs STM32 Comparison

Parameter	ESP32	STM32
Core	Xtensa LX6 dual-core	ARM Cortex-M0/M3/M4/M7

Parameter	ESP32	STM32
	(RISC-V in some variants)	(dual-core in some H7)
Clock	80–240 MHz	A few MHz to 480 MHz
Wireless	Built-in Wi-Fi and Bluetooth/BLE	Not built in (except STM32WB); external module needed
Program memory	External SPI Flash, typically 4–16 MB	Internal Flash, 16 KB to 2 MB
RAM	520 KB SRAM + optional PSRAM	Up to 1 MB+ SRAM, varies by series
Real-time performance	Moderate; RTOS-based, less deterministic interrupt latency	Excellent; NVIC gives very low, deterministic latency
Timers / PWM	Basic LEDC/MCPWM channels	Extensive advanced, general-purpose and basic timers; ideal for motor control
ADC	12-bit, 18 channels	12-bit (up to 16-bit on H7), multi-channel with injected/regular groups
DMA	Limited	Extensive multi-stream DMA
Power modes	Active, Modem-sleep, Light-sleep, Deep-sleep, Hibernation	Sleep, Stop, Standby (ultra-low-power L-series)
Security	Hardware AES/SHA/RSA, Secure Boot, Flash Encryption	CRC, RNG and crypto accelerators on select variants
Typical applications	IoT nodes, home automation, wearables, wireless sensor networks	Industrial control, motor drives, automotive ECUs, medical instrumentation
Ecosystem	ESP-IDF (FreeRTOS-based), Arduino IDE, MicroPython	STM32CubeIDE/CubeMX, HAL/LL drivers, Keil, IAR

Conclusion: ESP32 is built around wireless connectivity and IoT integration; STM32 is built around deterministic real-time control, peripheral richness and processing performance. The right choice depends on the application.