

Firmware 2.0

#51
Marzo 2025

Embedded Microcontrollers

IN QUESTO NUMERO:

IL POWER COMPUTING

CHE COS'È ZEPHYR RTOS E QUANDO PREFERIRLO A FREERTOS

L'ARCHITETTURA ARM NEI SYSTEM-ON-CHIP

CORSO DI ELETTRONICA PER RAGAZZI - PUNTATA 25

E MOLTI ALTRI ARTICOLI E PROGETTI!

it.emcelettronica.com

COSA LEGGERAI NEL 2025?

<i>TOPICS</i>	<i>MAKERS ZONE</i>	<i>DATA DI PUBBLICAZIONE</i>
PCB Design	Power Management	1 Febbraio
Embedded	Microcontrollers	1 Marzo
Automotive	Sensors	1 Aprile
Artificial Intelligence	Edge AI	1 Maggio
Raspberry Pi	Wearable Projects	1 Giugno
Wireless/RF	Retrogaming	1 Luglio
Arduino	Open Source Projects	1 Settembre
IoT	Smart Monitoring	1 Ottobre
Industry 4.0	Automation Projects	1 Novembre
Test&Measurements	Connectors	1 Dicembre

Innovazione e potenza nei sistemi embedded

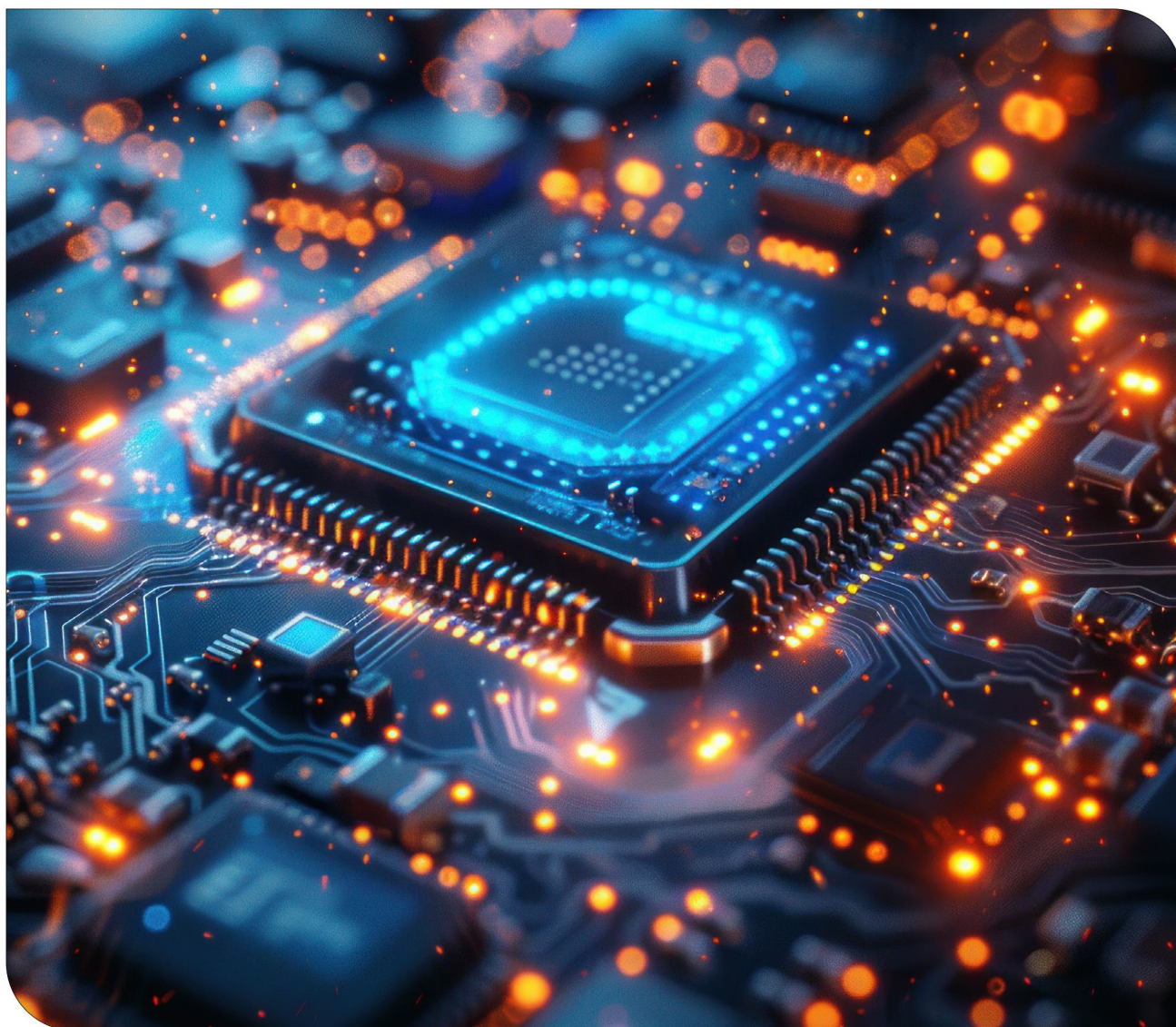
Efficienza, AI e connettività per l'elettronica del futuro

Cari lettori,
l'evoluzione della tecnologia embedded sta trasformando il mondo dell'elettronica e dell'ingegneria informatica con nuove soluzioni hardware e software. Le parole chiave da associare all'ecosistema embedded sono **sicurezza**, **efficienza energetica** e **potenza di calcolo**, parametri progettuali che spingono oggi gli ingegneri e gli sviluppatori a trovare soluzioni sempre più innovative e performanti. I sistemi embedded si trovano praticamente ovunque: nelle automobili, negli elettrodomestici, nei dispositivi medici, nei sistemi di automazione industriale, e persino nei giocattoli. Ciò che li rende così importanti è la loro capacità di eseguire compiti specifici in modo affidabile ed efficiente, con **consumi energetici ottimizzati e risorse hardware limitate**. Dai dispositivi IoT a bassa potenza alle soluzioni industriali più avanzate e complesse, i microcontrollori sono il motore che permette ai sistemi elettronici di diventare sempre più intelligenti ed efficienti. Il mercato embedded è in continua espansione grazie alla **crescente domanda di dispositivi connessi** e alla progressiva **miniaturizzazione dei componenti elettronici**. Microcontrollori sempre più potenti ed efficienti come quelli basati su architetture ARM Cortex-M o RISC-V, permettono di implementare funzioni avanzate mantenendo bassi i costi di produzione. Al contempo, lo sviluppo di sistemi embedded richiede ai progettisti di affrontare e risolvere diverse sfide ingegneristiche, una delle principali è il bilanciamento tra prestazioni e consumi; molti dispositivi devono infatti funzionare per anni con batterie di piccole dimensioni, il che impone l'uso di tecniche di gestione dell'energia e modalità di sleep avanzate. La progettazione di software per sistemi embedded è diventata nel tempo sempre più accessibile grazie a tool e framework, in tal senso ambienti di sviluppo come STM32CubeIDE e Zephyr RTOS semplificano la scrittura e il debugging del codice, mentre strumenti come FreeRTOS permettono di gestire più task in tempo reale con maggiore efficienza. Un aspetto fondamentale è rappresentato dalla sicurezza: con l'aumento della connettività, i sistemi sono sempre più esposti a vulnerabilità e attacchi informatici, per cui l'implementazione di soluzioni come la crittografia hardware o il secure boot sta diventando uno standard per proteggere i dati sensibili e garantire l'integrità del software. Suscita grande interesse anche l'integrazione di soluzioni basate su RISC-V, un'architettura open-source che sta guadagnando sempre più consensi grazie alla sua flessibilità e scalabilità. Guardando allo stato attuale, tra le maggiori tendenze vi è l'uso di linguaggi di programmazione di alto livello, come Python e MicroPython, che stanno guadagnando terreno rispetto ai tradizionali e più blasonati C e C++. La diffusione di questi linguaggi semplifica lo sviluppo e l'integrazione con tecnologie cloud, oltre alla creazione di dispositivi IoT intelligenti e connessi. Volgendo il nostro sguardo al futuro, possiamo dire che **l'elettronica embedded sarà sempre più orientata verso l'Intelligenza Artificiale e il Machine Learning**, con **microcontrollori dotati di acceleratori AI** che stanno creando nuove possibilità per il riconoscimento vocale, la visione artificiale, l'analisi predittiva in tempo reale e l'elaborazione dei dati. I microcontrollori con acceleratori AI vengono impiegati nell'elaborazione edge, si tratta sostanzialmente di **chip in grado di eseguire inferenze rapide con consumi ridotti e gestire reti neurali direttamente sull'hardware** senza la necessità di connessione costante al cloud, per una maggiore efficienza e sicurezza. L'integrazione di AI ed ML nei sistemi embedded sta rivoluzionando numerosi settori, apportando maggiore efficienza e autonomia nei dispositivi. Nei veicoli a guida autonoma i microcontrollori AI elaborano dati sensoriali in tempo reale per rilevare ostacoli e ottimizzare

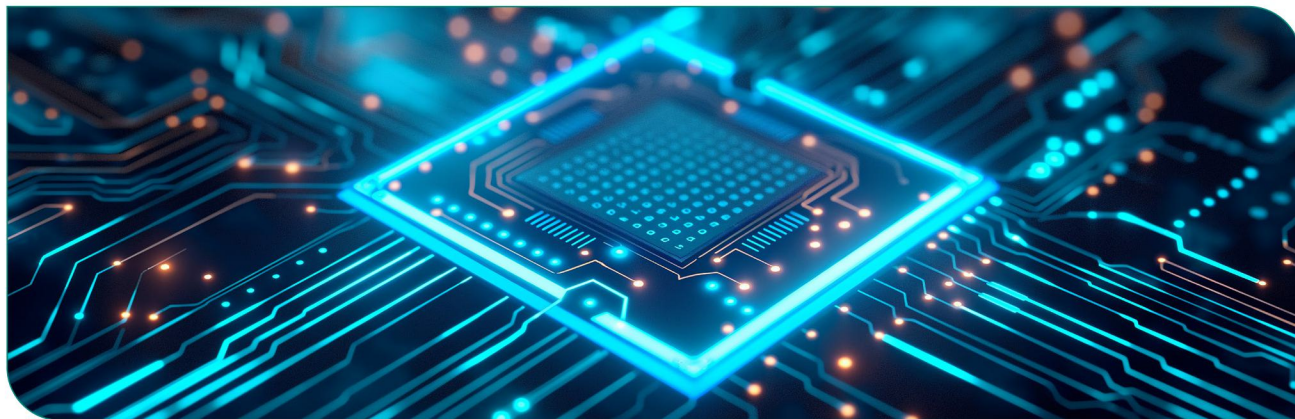
la navigazione, nell'industria manifatturiera la manutenzione predittiva sfrutta algoritmi embedded per identificare anomalie nei macchinari al fine di prevenire guasti, mentre nei dispositivi medici intelligenti l'AI monitora parametri vitali per prendere decisioni informate sulla base dell'analisi continua dei dati. Nel settore dell'IoT, telecamere smart basate su tecnologia AI con funzionalità di riconoscimento facciale garantiscono sicurezza, mentre gli assistenti vocali operano con reti neurali ottimizzate su hardware a basso consumo. Le applicazioni sembrano davvero infinite. In questa uscita di Firmware 2.0 exploreremo lo stato attuale del settore embedded ed analizzeremo le sfide progettuali e le soluzioni più innovative, per offrirvi una panoramica completa sulle tecnologie che stanno plasmando il futuro di questo strategico e affascinante settore.

Buona lettura!

Giordana Francesca Brescia



Embedded Microcontrollers



Founder&Editor
Emanuele Bonanni

CFO
Lidia Balica

Editorial Assistant
Maria Pisani

Maker in Chief
Giordana Francesca Brescia

Advertising & Marketing
Cristian Balica
cristian@contangosl.com

Graphic Designer
Marilde Mirra

Circulation

Users - 148.239
Social Network - 131.621

© Copyright

Tutti i diritti di riproduzione o di traduzione degli articoli pubblicati sono riservati. Manoscritti e disegni sono di proprietà di Contango SL.

E' vietata la riproduzione anche parziale degli articoli salvo espressa autorizzazione scritta dell'editore. I contenuti pubblicitari sono riportati senza responsabilità, a puro titolo informativo.

EDITORIALE

INNOVAZIONE E POTENZA
NEI SISTEMI EMBEDDED

3

LA PROGRAMMAZIONE
DEI MICROCONTROLLORI
NEI SISTEMI EMBEDDED

7

QUALE FUTURO PER
L'EMBEDDED
NEL 2025?

13

IL POWER COMPUTING

15

CHE COS'È ZEPHYR RTOS
E QUANDO PREFERIRLO A
FREERTOS

19

RISC-V E SICUREZZA NEI
MICROCONTROLLORI:
VERSO UN FUTURO
DELL'EMBEDDED
OPEN SOURCE

23

APACER ALL'EMBEDDED
WORLD 2025:
PADIGLIONE 1,
STAND 310 - ACCEDI ALLA
TRASFORMAZIONE INTELLIGENTE

27

LA PROGRAMMAZIONE
DEI MICROCONTROLLORI
ESP32

29

ACCELERARE LO
SVILUPPO EMBEDDED
CON CODEFUSION
STUDIO DI ANALOG DEVICES

34

SCOPRIAMO GLI
STANDARD SMARC,
QSEVEN E COM EXPRESS
PER APPLICAZIONI EMBEDDED

38

IL VENERABILE TIMER
555

42

CORSO DI ELETTRONICA
PER RAGAZZI - PUNTATA
25

46

RENESAS RA - FAMIGLIA
DI MICROCONTROLLORI
BASATA SU ARM CORE
CORTEX-M

53

LA PROGRAMMAZIONE ISP
DEI MICROCONTROLLORI
ATTINY E ATMEGA

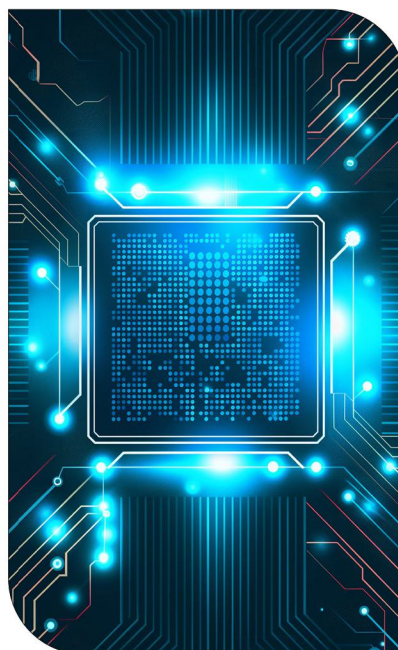
59

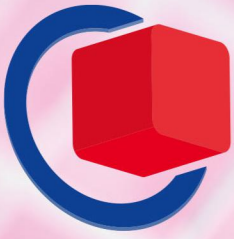
L'ARCHITETTURA ARM
NEI SYSTEM-ON-CHIP

68

CAPIRE GLI INTERRUPT
ED I TIMER

72





embeddedworld

Exhibition & Conference

CONNECTING THE
EMBEDDED COMMUNITY

11 – 13.3.2025

NUREMBERG, GERMANY



Get your
free ticket now!

embedded-world.de/codes

Use the voucher code **ew25GG**

Media partners

elektroniknet.de

Markt&Technik
DIE UNABHÄNGIGE NACHRICHTENFÜR ELEKTRONIK

Elektronik

Elektronik
automotive

Elektronik
•medical

connect
professional

NÜRNBERG / MESSE

IL POWER COMPUTING

di Daniele Valanzuolo

Siamo ormai abituati a utilizzare computer e smartphone per le più svariate funzioni: navigare in internet, giocare, scrivere testi, ascoltare musica e tanto altro. In realtà, la funzionalità chiave di ogni dispositivo dotato di CPU è quella di elaborare dati eseguendo specifiche istruzioni. Questo processo di elaborazione combina diversi meccanismi hardware e software al fine di eseguire il proprio compito con determinate prestazioni. Comprendere a pieno le potenzialità di calcolo del proprio elaboratore è un passo fondamentale per ogni progettista al fine di definire l'architettura del proprio progetto ottimizzandola per le prestazioni desiderate.

Avete presente il solito confronto che si ripropone tra CPU prodotte da Intel e quelle prodotte da AMD? Quali sono le migliori? Quali presentano una potenza di calcolo superiore? Quali consumano meno? Una risposta certa non esiste e, quando si parla di potenza computazionale, sono tantissimi gli aspetti che devono essere considerati, non da meno le funzionalità finali che devono svolgere.

Ogni tipologia di CPU, o per chiamarla in maniera ancora più generica, unità di elaborazione, è caratterizzata da diversi fattori e proprietà che influenzano le prestazioni del sistema. Tra le principali caratteristiche possiamo sicuramente individuare le seguenti:

- **l'architettura:** caratteristiche come numero di core e di thread, nonché l'architettura effettiva della CPU, sono aspetti fondamentali di un confronto. In realtà, anche le soluzioni tecnologiche di come è realizzata possono essere già un elemento di paragone per il potere computazionale;
- **la velocità di clock:** con l'avanzare del progresso tecnologico è aumentata a dismisura anche la velocità di clock delle unità di elaborazione. Tuttavia, come vedremo nel dettaglio nel corso di questo articolo, non sempre un valore più alto della velocità di clock corrisponde a prestazioni migliori;
- **la memoria cache:** oltre alla memoria centrale di una architettura standard di un elaboratore, sempre di più la memoria cache sta assumendo un ruolo importante per l'incremento delle prestazioni di elaborazione;
- **consumo energetico:** alcune volte il consumo energetico ha un ruolo chiave per quelle applicazioni dove si deve risparmiare energia;
- **dissipazione di calore:** è un altro aspetto fondamentale in quanto la necessità di gestire correttamente il calore generato può influenzare le prestazioni del processore.

L'ARCHITETTURA

L'architettura del processore è una delle caratteristiche fondamentali che, anche per uno stesso fornitore, evolve nelle diverse generazioni di prodotti. Parlare di architettura significa tanto e nulla. Ci sono tantissimi aspetti quali ad esempio la **microarchitettura** (ossia il core utilizzato), il numero di core, il numero di thread, l'esistenza di co-processor, la tecnologia di fabbricazione e il set di istruzioni. Ogni tipologia di architettura presenta vantaggi e svantaggi che la possono favorire o sfavorire per determinate funzionalità. Ad esempio, l'architettura di una GPU è ben diversa da quella di una CPU ed è orientata principalmente ad operazioni grafiche, quindi vettoriali e in maniera parallela.

Troviamo larga diffusione ultimamente dei core ARM in dispositivi mobile a basso consumo grazie alla loro efficienza e personalizzazione rispetto alle CPU AMD o Intel, concepite per applicazioni desktop. Tuttavia, dove vediamo due differenti architetture è sicuramente tra unità di elaborazione per applicazioni grafiche (le bene note GPU) e le classiche CPU per applicazioni generaliste. Infatti, possiamo identificare le due aree di utilizzo delle stesse come segue:

- La **CPU** (Central Processor Unit) gestisce attività informatiche del tutto generali e le operazioni di base del sistema come l'interfacciamento con le periferiche, la memoria e le altre risorse. In combinazione con sistemi operativi, la **CPU consente di svolgere funzioni e applicazioni di varia natura** come ad esempio la navigazione web, la videoscrittura e altro.
- La **GPU** (Graphical Processor Unit) è concepita per svolgere in maniera ottimizzata determinate tipologie di **elaborazione dei dati**, che trovano principale applicazione nella grafica vettoriale e nelle elaborazioni matematiche avanzate. Infatti, in linea generale, le GPU sono adatte per eseguire molteplici calcoli in parallelo su dati più o meno complessi o di tipo matriciale.

Dunque, mentre le CPU sono progettate per la versatilità e il controllo di funzioni complesse e diversificate, le GPU sono progettate per gestire attività semplici e ripetitive in modo efficiente e in massa.

LA VELOCITÀ DI CLOCK

La velocità di clock del nostro processore è sicuramente un parametro che ci indica quanto veloce possa andare. In realtà, la teoria ci dice che all'aumentare della velocità aumentano anche i consumi energetici in quanto la potenza dissipata è dinamica e strettamente legata alla frequenza di commutazione dei circuiti. Tuttavia, la frequenza di clock non è un parametro affidabile per indicare le prestazioni della CPU ma bisogna far riferimento ad altre metriche che sono:

- istruzioni per ciclo IPC (ossia quante istruzioni possono essere eseguite ogni colpo di clock)
- istruzioni al secondo IPS indicano quante operazioni possono essere eseguite nell'unità di tempo
- FLOPS: quante operazioni a virgola mobile possono essere eseguite
- Latenza: tempo impiegato dalla CPU per rispondere ad una richiesta

LA MEMORIA CACHE

La CPU utilizza una cache per ridurre il tempo e la potenza necessari per recuperare i dati dalla memoria. La cache è pensata per essere più veloce, più compatta e vicina alle altre parti della CPU rispetto alla memoria principale. Ci sono molti livelli nella cache della CPU. Tutti i core della CPU condividono il livello più lontano dal core, mentre il livello più vicino al core viene utilizzato solo da quel core. Le CPU contemporanee eseguono automaticamente la gestione della cache. Ciascun livello determina se il frammento di memoria deve essere conservato o rimosso in base alla frequenza di utilizzo.

cessori. Ecco alcuni dei principali sviluppi e tendenze:

1. **Aumento delle dimensioni della cache:** una tendenza evidente è stata l'aumento delle dimensioni della cache, soprattutto per i livelli di cache L2 e L3. Le moderne CPU possono avere cache di grandi dimensioni, che possono variare da diversi megabyte a decine di megabyte. Questo aumento delle dimensioni della cache consente una maggiore capacità di memorizzazione dei dati e delle istruzioni, riducendo la necessità di accedere alla RAM esterna e migliorando le prestazioni complessive.
2. **Miglioramento dell'architettura della cache:** le CPU moderne adottano architetture di cache avanzate, come la cache adattiva o predittiva, che utilizzano algoritmi sofisticati per anticipare e pre-caricare i dati e le istruzioni più probabili che saranno richiesti dalla CPU. Questo contribuisce a ridurre i tempi di attesa della CPU e ad aumentare l'efficienza complessiva del sistema.
3. **Cache gerarchica:** le moderne CPU spesso implementano una struttura di cache gerarchica con più livelli di cache (L1, L2, L3), ciascuno con caratteristiche diverse e dimensioni variabili. Questo approccio consente di ottimizzare l'accesso alla memoria a seconda delle esigenze del processore e di minimizzare i tempi di latenza.
4. **Cache inclusiva vs esclusiva:** un'altra tendenza è rappresentata dalla scelta dell'architettura della cache inclusiva o esclusiva. Nella cache inclusiva, i dati presenti nella cache L1 sono anche presenti nella cache L2 o L3, riducendo il rischio di cache miss. D'altra parte, nella cache esclusiva, i dati non sono duplicati tra i livelli di cache, consentendo una maggiore flessibilità nell'utilizzo della memoria cache.

QUELLO CHE HAI LETTO E' UN ESTRATTO, L'ARTICOLO COMPLETO E' RISERVATO AGLI ABBONATI AD ELETTRONICA OPEN SOURCE.

PERCHE' ABBONARSI A PLATINUM 2.0?

UN ANNO DI **FIRMWARE 2.0**
TUTTI GLI ARTICOLI TECNICI RISERVATI
CONTEST E PROMOZIONI RISERVATI



 **VOGLIO ABBONARMI!**

CHE COS'È ZEPHYR RTOS E QUANDO PREFERIRLO A FREERTOS

di **Andrea Garrapa**

Zephyr OS è un sistema operativo in tempo reale (RTOS) open source, progettato per supportare i dispositivi IoT con il minimo ingombro. Come molti sapranno, esiste già un RTOS largamente utilizzato il cui nome è FreeRTOS. Ci si potrebbe allora chiedere perché si dovrebbe perdere tempo ad imparare un altro sistema operativo. Questo articolo ha proprio l'obiettivo di rispondere a questa domanda, descrivendo le caratteristiche di Zephyr e guidando il lettore nella scelta più opportuna tra i due competitori a seconda dei casi d'uso.

INTRODUZIONE

I dispositivi IoT sono generalmente piccoli, quindi, a differenza del PC, non hanno spazio quasi infinito, ecco perché spesso si affidano a qualcosa chiamato **sistema operativo in tempo reale (RTOS)** invece che ad un sistema operativo tradizionale. Un RTOS esegue una singola attività in tempo reale anziché più attività lentamente come un sistema operativo tradizionale, ad esempio Linux, Windows o iOS. In realtà, oltre a RTOS esistono i dispositivi **bare metal**. Un dispositivo bare metal esegue una singola applicazione anziché un sistema operativo completo. Per comprendere la differenza tra questo ed un RTOS o un sistema operativo, pensiamo a come il computer può eseguire varie applicazioni diverse nel corso di una giornata. Ad esempio, Windows potrebbe essere il sistema operativo, ma è possibile eseguire varie applicazioni come Microsoft Word, un'applicazione di fotoritocco, ecc. Un sistema operativo può eseguire diverse di queste applicazioni contemporaneamente. Un RTOS passerà da un'applicazione all'altra per eseguirle una alla volta molto più velocemente. Al contrario, un dispositivo bare metal viene creato appositamente per un'applicazione. Molti dispositivi IoT, ad esempio gli orologi intelligenti, sono piccoli ma potrebbero dover eseguire diverse applicazioni nel corso della giornata, come applicazioni di monitoraggio della salute, applicazioni GPS e altro ancora. Per queste esigenze, un RTOS è l'opzione più comune.

ZEPHYR

Zephyr è un RTOS scalabile, il che lo rende adatto a un'ampia gamma di dispositivi con diversi vincoli di risorse. La scalabilità è ottenuta attraverso un'architettura modulare che consente agli sviluppatori di includere solo i componenti di cui hanno bisogno, riducendo al minimo l'ingombro del sistema. Essendo un sistema

operativo in tempo reale (RTOS), Zephyr offre funzionalità come il multitasking preventivo, la comunicazione inter-thread e il supporto del clock in tempo reale. Il sistema operativo è inoltre dotato di tecnologie e protocolli di rete come TCP/IP, Bluetooth e IEEE 802.15.4 (utilizzato ad esempio in Zigbee), **MQTT**, NFS e LoRaWAN. Insieme alle sue capacità di rete, le funzionalità di gestione energetica integrate in Zephyr lo rendono adatto per applicazioni IoT ad alta efficienza energetica e dispositivi alimentati a batteria.

Una serie di librerie e middleware semplificano attività comuni, come protocolli di comunicazione, file system e driver di dispositivo. Zephyr è progettato anche per essere compatibile con le certificazioni di sicurezza come ISO 26262, rendendolo adatto ad applicazioni critiche per la sicurezza.

Zephyr non è Linux, ma si avvale di concetti, tecniche e strumenti utilizzati da Linux. Ad esempio, *Kconfig* viene utilizzato per configurare il sistema operativo e le proprietà, e le configurazioni dell'hardware vengono descritte utilizzando *Device Tree Specific (DTS)*. Pertanto, gli sviluppatori Linux si sentiranno subito a loro agio durante la codifica per Zephyr. Ultimo ma non meno importante, Zephyr è rilasciato sotto la licenza open source Apache 2.0, che ne consente l'uso sia commerciale che non commerciale. La sua comunità di utenti fornisce supporto e documentazione.

SICUREZZA NEI DISPOSITIVI IOT

Nel regno dei dispositivi IoT, la sicurezza è una necessità assoluta. Immaginiamo una rete di termostati intelligenti installati in una città, che controllano i sistemi di riscaldamento e raffreddamento di innumerevoli edifici. Se questi dispositivi non dispongono di solide misure

di sicurezza, potrebbero diventare i bersagli principali dei criminali informatici. Un attacco riuscito potrebbe provocare qualsiasi cosa, da un lieve disagio a gravi interruzioni.

È qui che il sistema operativo Zephyr brilla. **Zephyr fornisce agli sviluppatori una serie di funzionalità di sicurezza progettate specificamente per i dispositivi IoT.** Queste includono l'avvio sicuro (per garantire che solo il codice autenticato venga eseguito sul dispositivo), il supporto della crittografia hardware (per la protezione dei dati) e protocolli di sicurezza a livello di rete come DTLS e TLS (per salvaguardare la trasmissione dei dati). In sostanza, Zephyr fornisce gli strumenti necessari per costruire dispositivi IoT sicuri in grado di resistere agli attacchi e proteggere i dati sensibili.

FUNZIONALITÀ DI CONNETTIVITÀ DI ZEPHYR

Zephyr supporta un'ampia gamma di protocolli di rete tra cui Bluetooth Low Energy (BLE), Wi-Fi, Ethernet e persino 6LoWPAN (un protocollo che consente ai dispositivi a basso consumo di connettersi direttamente a Internet). Ciò significa che è possibile progettare dispositivi IoT in grado di comunicare perfettamente con altri dispositivi e reti.

Ma approfondiamo un esempio particolare: BLE o Bluetooth Low Energy. Questa tecnologia è ideale per le applicazioni IoT, poiché consuma meno energia mantenendo una comunicazione solida. Immaginiamo una rete di sensori intelligenti in un grande magazzino che monitora i livelli di temperatura e umidità. Con il supporto BLE nel sistema operativo Zephyr, questi sensori possono inviare dati in tempo reale ad un sistema centrale senza scaricare le batterie troppo velocemente.

sarà necessario clonare il progetto Zephyr dal suo repository GitHub:

<https://github.com/zephyrproject-rtos/zephyr>

Passaggio 3: configurare il proprio dispositivo. Il dispositivo va configurato utilizzando *Device Tree* e file *Kconfig*. Questi file consentono di specificare i dettagli hardware e selezionare funzionalità software specifiche per il dispositivo IoT che si intende utilizzare. Per visualizzare i file DTS in un modo un po' più comodo, si può utilizzare il plugin DeviceTree per Visual Studio Code. Questo plugin fornisce l'evidenziazione della sintassi e il ripiegamento del codice, rendendo i file più facili da leggere (i file DTS utilizzano la sintassi in stile linguaggio C). La configurazione del progetto è gestita da Kconfig, il sistema di configurazione del kernel in fase di compilazione utilizzato anche da Linux. Esistono diversi modi per interagire con esso, e uno di questi è un'interfaccia utente grafica (GUI) che si può aprire attraverso un semplice comando Linux.

Passaggio 4: creazione e flashing. Infine, la compilazione della propria applicazione con tutte le sue dipendenze in un file binario che può essere flashato nella memoria del proprio dispositivo. Il flashing potrebbe richiedere uno strumento di programmazione esterno, come un adattatore J-Link o un altro programmatore (JTAG o SWD compatibile).

Nel caso in cui si volesse provare Zephyr anche senza la disponibilità di un dispositivo IoT, esso presenta un supporto integrato nell'emulatore QEMU (solo su Linux/macOS). Ciò consente di eseguire e testare virtualmente le applicazioni.

RISORSE E STRUMENTI PER GLI SVILUPPATORI

QUELLO CHE HAI LETTO È UN ESTRATTO, L'ARTICOLO COMPLETO È RISERVATO AGLI ABBONATI AD ELETTRONICA OPEN SOURCE.

PERCHÉ ABBONARSI A PLATINUM 2.0?

UN ANNO DI **FIRMWARE 2.0**
TUTTI GLI ARTICOLI TECNICI RISERVATI
CONTEST E PROMOZIONI RISERVATI



VOGLIO ABBONARMI!

APACER ALL'EMBEDDED WORLD 2025: PADIGLIONE 1, STAND 310 – ACCEDI ALLA TRASFORMAZIONE INTELLIGENTE

di **Apacer**

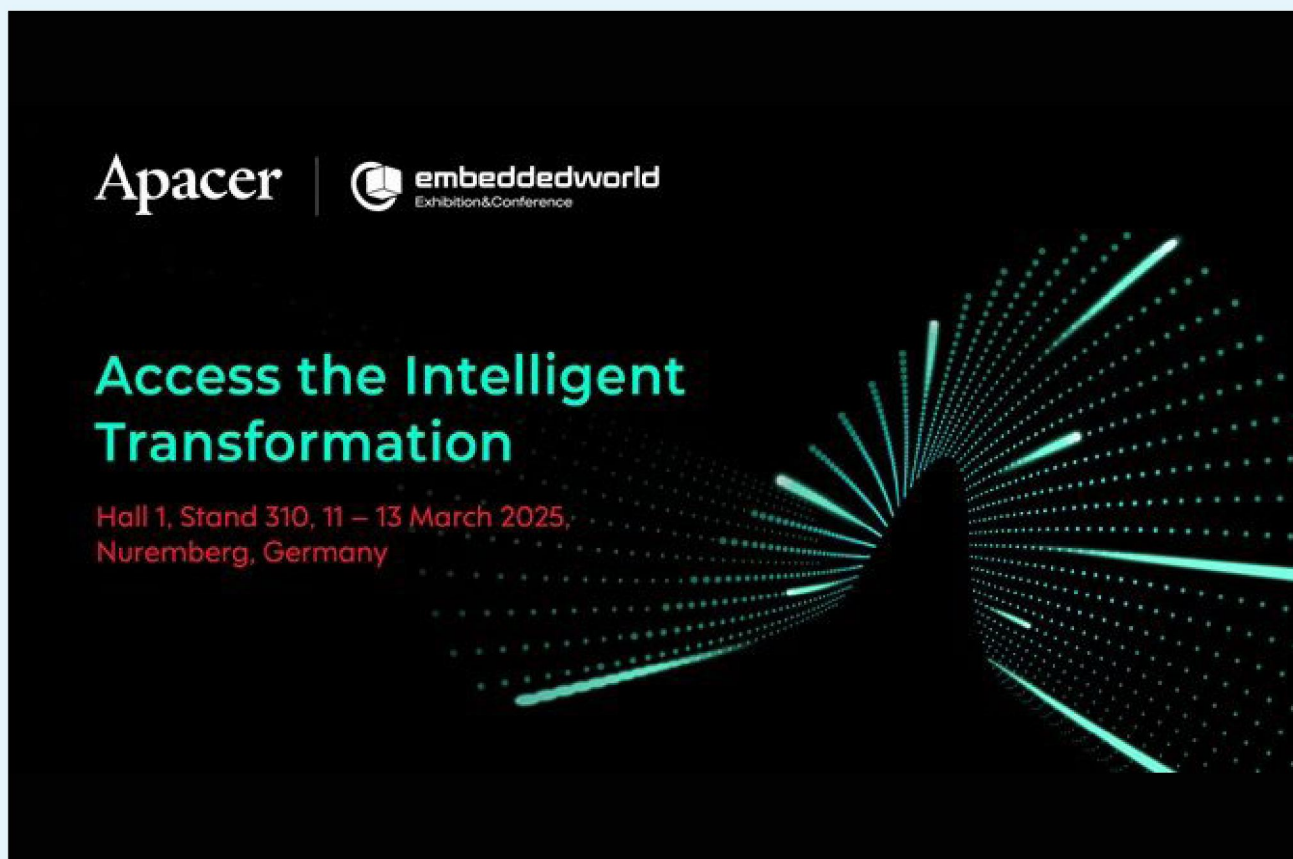
A *embedded world 2025*, **Apacer** presenterà le sue ultime soluzioni di storage industriale al Padiglione 1, Stand 310. Queste soluzioni all'avanguardia offrono prestazioni di elaborazione migliorate, efficienza energetica ottimizzata e design sostenibili su misura per un'ampia gamma di applicazioni. Inoltre, il Dott. Gibson Chen, Presidente di Apacer, terrà una presentazione intitolata "Advanced Storage for Industry Transformation" al forum degli espositori martedì 11 marzo alle 10:30 nella Hall 3. Partecipa per esplorare i progressi all'avanguardia che plasmano il futuro dello storage industriale.

Apacer presenta la **serie Enterprise SSD PCIe** con nuovi modelli nei formati Gen5 E3S e 2,5" con una capacità fino a 30 TB. Grazie alla loro elevata capacità, prestazioni e affidabilità, nonché alla bassa latenza, sono perfetti per server edge o data center.

Con un innovativo **SSD PCIe Gen4** personalizzato a **basso consumo**, Apacer risolve il problema del calore

che gli SSD PCIe hanno in genere a causa della loro elevata velocità. Grazie al suo basso consumo energetico, consente sistemi senza ventole con operazioni di scrittura stabili.

Apacer presenta anche il **primo SSD completamente privo di piombo** e la **serie SSD CET (Commercial Extended Temperature)**. Con un intervallo di temperatura da -25 °C a 85 °C, offre un equilibrio ottimale tra



The graphic features the Apacer logo and the Embedded World Exhibition & Conference logo at the top left. The main text reads "Access the Intelligent Transformation" in a large, bold, teal font. Below this, the event details are provided: "Hall 1, Stand 310, 11 – 13 March 2025, Nuremberg, Germany". The background is dark with a dynamic, glowing teal pattern of dots and lines that creates a sense of depth and movement, resembling a data tunnel or a futuristic interface.

costi e prestazioni per molte applicazioni.

I nuovi **DDR5 6400 CUDIMM** e **CSODIMM**, nonché **LPDDR5X CAMM2**, il nuovo standard JEDEC per SO-DIMM, garantiscono un funzionamento senza interruzioni per l'analisi dei dati in tempo reale, l'elaborazione di big data e l'elaborazione ad alto carico, rendendoli la scelta ideale per applicazioni di elaborazione ad alte prestazioni (HPC) e AI. Come tutte le soluzioni di memoria Apacer DDR5, sono dotate di un design del resistore completamente privo di piombo, eliminando la necessità di esenzioni ai sensi della direttiva RoHS dell'UE.

La comprovata tecnologia **CorePower** collega l'alimentazione in caso di interruzione di corrente, garantendo un trasferimento dati stabile senza perdita di dati. Apacer ha ora ampliato il portafoglio di prodotti CorePower per includere SLC-liteX con cicli P/E fino a 100.000.

Apacer ha aggiunto la nuova tecnologia **CoreRescue**, un altro vincitore del Taiwan Excellence Award 2025, alla sua popolare suite di backup e ripristino CoreSnapshot. CoreRescue esegue automaticamente il processo di ripristino e ripristina il sistema al normale funzionamento in modo rapido e semplice. I clienti possono scegliere la soluzione cloud o on-premise più appropriata in base alle loro esigenze tra quattro soluzioni principali: CoreRescue Button, CoreRescue USB (USB Self Recovery), CoreRescue OOB e CoreRescue ASR (Auto Self Recovery).

La **tecnologia di backup e ripristino CoreSnapshot 3** supporta una funzione multi-backup per sistemi che devono funzionare senza problemi ed a prova di errore. Consente di ripetere backup e ripristini senza reinstallazione, in modo che gli utenti possano eseguire il backup dei propri dati in qualsiasi momento. Apacer presenterà gli SSD SATA III in vari formati con tecnologia Core-

INFORMAZIONI SU APACER

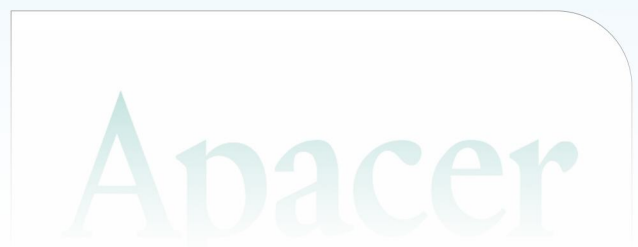
Fondata nel 1997, Apacer (TWSE:8271) è un marchio leader mondiale di storage digitale con capacità complete di R&S, progettazione, produzione e marketing.

Con anni di tecnologia di storage digitale brevettata accumulata e una profonda esperienza di successo in R&S, Apacer offre una gamma competitiva di prodotti e servizi personalizzati.

Le nostre linee di prodotti sono diversificate e comprendono soluzioni per moduli di memoria, SSD industriali, prodotti di storage digitale consumer e applicazioni integrate Internet of Things.

Apacer si impegna a implementare il nostro valore fondamentale "Becoming Better Partners": manteniamo le nostre promesse, ci impegniamo per un miglioramento costante e sviluppiamo soluzioni che siano reciprocamente vantaggiose per noi e per i nostri clienti. Creiamo continuamente soluzioni di storage innovative e diversificate e servizi di integrazione hardware/software per vari settori.

Ci impegniamo a diventare un partner migliore nell'ecosistema industriale e ad offrire vantaggi sostanziali a tutte le parti interessate. Scopri di più sui prodotti Apacer per applicazioni industriali su: <https://www.apacer.com/>.



**QUELLO CHE HAI LETTO E' UN ESTRATTO, L'ARTICOLO
COMPLETO E' RISERVATO AGLI ABBONATI
AD ELETTRONICA OPEN SOURCE.**

PERCHE' ABBONARSI A PLATINUM 2.0?

**UN ANNO DI FIRMWARE 2.0
TUTTI GLI ARTICOLI TECNICI RISERVATI
CONTEST E PROMOZIONI RISERVATI**



VOGLIO ABBONARMI!

CORSO DI ELETTRONICA PER RAGAZZI – PUNTATA 25

di Fulvio De Santis

Nella **precedente puntata** del corso abbiamo continuato a trattare il diodo a semiconduttore, descrivendo il funzionamento della giunzione p-n a circuito aperto, il processo di diffusione dei portatori maggioritari e minoritari, e la formazione della regione di carica spaziale. Inoltre, abbiamo spiegato il funzionamento in polarizzazione diretta e in polarizzazione inversa della giunzione p-n a circuito chiuso, dimostrando che la corrente può scorrere solo in un verso, dalla zona p alla zona n, quando la giunzione è polarizzata direttamente, mentre se la giunzione è polarizzata inversamente scorre solo una piccola corrente nella direzione opposta, dalla zona n alla zona p. In questa nuova puntata del corso, approfondiremo lo studio del diodo iniziando con un breve riepilogo della giunzione p-n a circuito chiuso, poi tratteremo la relazione corrente-tensione del diodo e, infine, vedremo come determinare il punto di lavoro del diodo graficamente mediante la retta di carico.

INTRODUZIONE: UN RIEPILOGO SULLA POLARIZZAZIONE DELLA GIUNZIONE P-N

Per meglio comprendere il nuovo argomento sul diodo a giunzione, riteniamo utile riepilogare sinteticamente il funzionamento della giunzione p-n a circuito chiuso polarizzata direttamente e inversamente, argomento che abbiamo affrontato nella precedente puntata 24 del corso nel paragrafo “**Funzionamento della giunzione p-n a circuito chiuso**”.

Polarizzazione diretta della giunzione p-n

Della giunzione p-n polarizzata direttamente, il cui schema è riportato in **Figura 1**, analizziamo le correnti che l'attraversano indicate con il loro verso:

La corrente totale I che circola nel circuito è formata dalla corrente di diffusione ID e dalla corrente di saturazione inversa IR di direzione contraria alla ID . Sono indicati anche i versi del campo elettrico E presente sulla giunzione, e della tensione V della batteria. Nella polarizzazione diretta della giunzione, la tensione V della batteria influisce sul potenziale della giunzione ove agisce anche il campo elettrico E . Infatti, dato che i versi della tensione della batteria V e del campo elettrico E sono discordi, la tensione totale che interessa la giunzione è la differenza fra la tensione V e la tensione E , ovvero $V_{tot}=E-V$.

Quando il valore della tensione V supera il valore della tensione del campo elettrico E , ovvero della barriera di potenziale, la tensione risultante applicata agli estremi della regione di carica spaziale favorisce l'incremento

del flusso della corrente di diffusione ID dei portatori maggioritari di tipo p diretti verso la zona n della giunzione. In questa situazione, ciò che è importante sottolineare è che se viene applicata una tensione V della batteria di valore tale da superare il potenziale del campo elettrico E , la barriera di potenziale si annulla completamente consentendo il passaggio della massima corrente ID attraverso la giunzione, di conseguenza la giunzione p-n si comporta come un conduttore di resistenza nulla (idealmente).

Abbiamo detto sopra che la corrente I che scorre nel circuito, oltre alla corrente di diffusione ID , è costituita anche dalla corrente di saturazione inversa IR , di verso opposto alla ID . Generata da energia termica, la IR è una corrente molto bassa formata da portatori minoritari, lacune dirette dalla zona n alla zona p ed elettroni diretti dalla zona p alla zona n. La corrente I che scorre nel circuito può essere espressa dalla relazione $I=ID-IR\approx ID$, considerando che ID è di diversi ordini di grandezza maggiore della corrente di saturazione inversa IR .

Polarizzazione inversa della giunzione p-n

La **Figura 2** mostra la giunzione a circuito chiuso polarizzata inversamente.

Notiamo che in questo circuito in cui la giunzione è polarizzata inversamente, la tensione V della batteria ha verso concorde al verso del campo elettrico E presente agli estremi della regione di carica spaziale. Quindi, la tensione totale che interessa la giunzione corrisponde

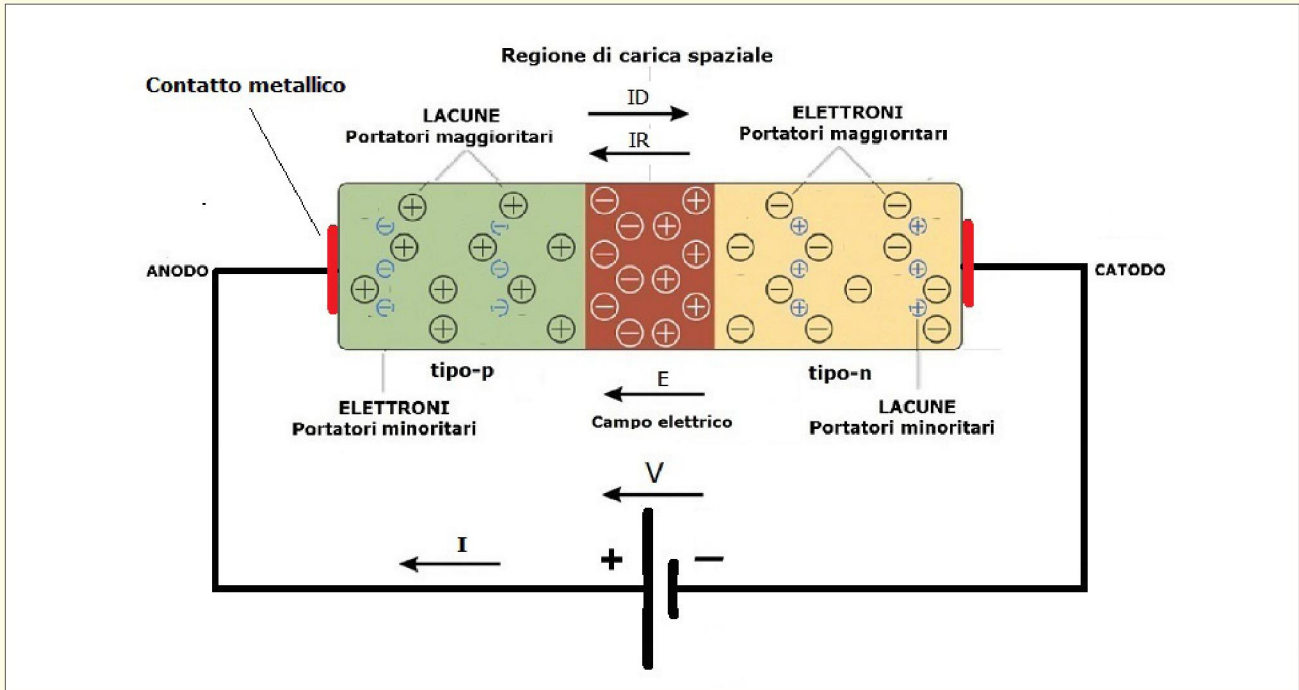
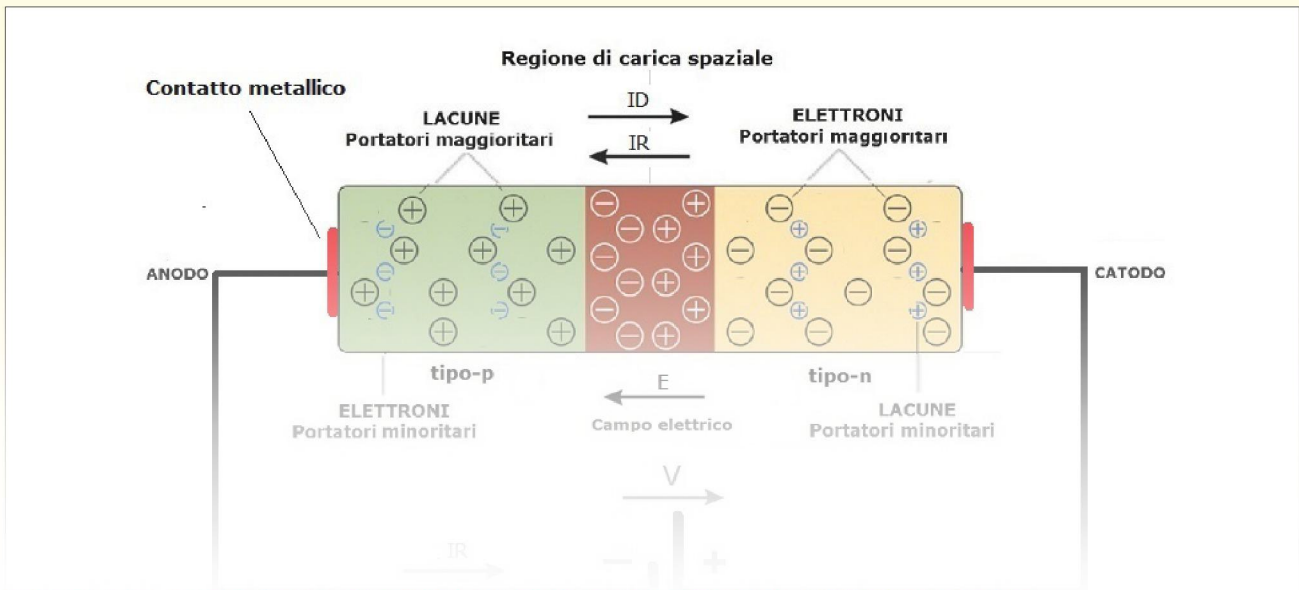


Figura 1: Polarizzazione diretta della giunzione p-n



QUELLO CHE HAI LETTO E' UN ESTRATTO, L'ARTICOLO COMPLETO E' RISERVATO AGLI ABBONATI AD ELETTRONICA OPEN SOURCE.

PERCHE' ABBONARSI A PLATINUM 2.0?

**UN ANNO DI FIRMWARE 2.0
TUTTI GLI ARTICOLI TECNICI RISERVATI
CONTEST E PROMOZIONI RISERVATI**



VOGLIO ABBONARMI!!

L'ARCHITETTURA ARM NEI SYSTEM-ON-CHIP

di Daniele Valanzuolo

Nel contesto dei microcontrollori, ARM ha rivoluzionato il settore offrendo soluzioni che combinano alte prestazioni e basso consumo energetico. La loro architettura avanzata e la capacità di adattarsi a diverse applicazioni li rendono essenziali per una vasta gamma di dispositivi, dai microcontrollori nei piccoli gadget IoT ai complessi sistemi integrati nei dispositivi industriali. Dunque, nel panorama della tecnologia embedded, i core processore ARM rappresentano ormai un punto di riferimento fondamentale per i progettisti elettronici e informatici. In questo articolo, andremo ad esplorare in dettaglio i principali core ARM Cortex, evidenziando le loro caratteristiche tecniche e le performance.

INTRODUZIONE

I microcontrollori basati su ARM Cortex sono progettati per soddisfare esigenze diverse, variando dalla gestione di semplici dispositivi IoT fino al controllo di sistemi industriali avanzati. La famiglia Cortex include diverse tipologie di core, ognuno ottimizzato per specifici compiti e prestazioni. Ad esempio, il Cortex-M è focalizzato su applicazioni a basso consumo energetico, mentre il Cortex-A è destinato a sistemi con requisiti di elaborazione più intensivi. Una delle chiavi del successo dell'architettura ARM è la sua capacità di offrire elevate prestazioni mantenendo un consumo energetico ridotto, caratteristica essenziale **per i sistemi embedded**, dove l'efficienza energetica può determinare la longevità di un dispositivo alimentato a batteria. I core ARM implementano tecniche avanzate di gestione dell'alimentazione e ottimizzazioni del percorso dati che riducono il consumo energetico senza compromettere la performance.

LE TIPOLOGIE DI CORTEX

Quando si parla di microcontrollori ARM, è fondamentale distinguere tra le diverse famiglie di core che ARM offre, ovvero Cortex-A, Cortex-R e Cortex-M. Ogni famiglia è concepita e progettata con caratteristiche specifiche che le rendono adatte a una varietà di applicazioni nei campi dell'embedded, delle alte prestazioni e della gestione in tempo reale.

Partendo dai core **ARM Cortex-A**, questi sono progettati per offrire alte prestazioni e sono comunemente impiegati in applicazioni che richiedono capacità di calcolo elevate, come smartphone, tablet, e dispositivi di infotainment. I core sono caratterizzati da una pipeline avanzata, spesso superscalare, e includono sofisticate tecniche di gestione della cache per ottimizzare le prestazioni. La famiglia Cortex-A supporta sistemi operativi complessi come Linux e Android, grazie alla loro capa-

bilità di gestire multitasking e di offrire ampie capacità di memoria. Le varianti più recenti, come il Cortex-A76, integrano anche supporto per l'architettura ARMv8-A, che include istruzioni a 64 bit, permettendo un significativo aumento delle prestazioni per applicazioni intensive.

Passando ai core **ARM Cortex-R**, ci troviamo di fronte ad una famiglia progettata appositamente per applicazioni real-time che richiedono alta affidabilità e prestazioni deterministiche. Tali core sono comunemente utilizzati in sistemi di controllo industriale, automotive e dispositivi medici, dove la prevedibilità del comportamento del sistema è determinante. I core Cortex-R offrono un set di istruzioni ottimizzato per il real-time e sono dotati di caratteristiche come la protezione della memoria e la correzione degli errori (ECC) per garantire l'integrità dei dati. Inoltre, supportano la tecnologia di interrupt a bassa latenza, che è essenziale per rispondere rapidamente agli eventi esterni.

Infine, i core **ARM Cortex-M** rappresentano la scelta ideale per applicazioni embedded che richiedono bassa potenza e un'alta efficienza energetica. Si tratta, infatti, di core ampiamente utilizzati in dispositivi come sensori, controllori di dispositivi IoT e piccoli elettrodomestici. La famiglia Cortex-M include core come il Cortex-M0, M3, M4 e M7, ognuno dei quali offre un diverso equilibrio tra potenza di elaborazione e consumo energetico.

Il set di istruzioni Thumb-2, utilizzato dai core Cortex-M, permette di ottimizzare il codice per ottenere alta densità di istruzioni e migliorare l'efficienza della memoria. Inoltre, questi core supportano funzionalità avanzate come il **Digital Signal Processing (DSP)** nei modelli Cortex-M4 e M7, rendendoli ideali per applicazioni che richiedono elaborazione di segnali in tempo reale.

Architettura di Core ARM	Caratteristiche Principali	Microcontrollori ARM
ARM Cortex-M0	Ultra low power, 32-bit, frequenza fino a 50 MHz, set di istruzioni Thumb-2	STM32F0, NXP LPC1100, Atmel SAM D20
ARM Cortex-M3	Efficienza energetica, 32-bit, frequenza fino a 100 MHz, set di istruzioni Thumb-2, gestione di eccezioni avanzata	STM32F1, STM32L1, NXP LPC1300
ARM Cortex-M4	Performance elevata, 32-bit, frequenza fino a 200 MHz, set di istruzioni Thumb-2, DSP (Digital Signal Processing)	STM32F4, NXP Kinetis K, Texas Instruments TM4C
ARM Cortex-M7	Alte prestazioni, 32-bit, frequenza fino a 600 MHz, set di istruzioni Thumb-2, DSP e floating point unit (FPU)	STM32F7, NXP i.MX RT
ARM Cortex-A7	Efficienza energetica, 32-bit, frequenza fino a 1.5 GHz, supporto per sistemi operativi complessi come Linux	Raspberry Pi 2, Freescale i.MX 6
ARM Cortex-A9	Performance bilanciate, 32-bit, frequenza fino a 2 GHz, cache L1 e L2, pipeline superscalare	NVIDIA Tegra 2, Freescale i.MX 6
ARM Cortex-A53	Efficienza energetica, 64-bit, frequenza fino a 2.5 GHz, ARMv8-A architecture, supporto per virtualizzazione	Raspberry Pi 3, Qualcomm Snapdragon 410
ARM Cortex-A57	Alte prestazioni, 64-bit, frequenza fino a 2.3 GHz, ARMv8-A architecture, pipeline out-of-order	NVIDIA Tegra X1, Samsung Exynos 7420

QUELLO CHE HAI LETTO E' UN ESTRATTO, L'ARTICOLO COMPLETO E' RISERVATO AGLI ABBONATI AD ELETTRONICA OPEN SOURCE.

PERCHE' ABBONARSI A PLATINUM 2.0?

**UN ANNO DI FIRMWARE 2.0
TUTTI GLI ARTICOLI TECNICI RISERVATI
CONTEST E PROMOZIONI RISERVATI**



VOGLIO ABBONARMI!

ABBONATI A

Firmware 2.0

PER AVERE **TUTTA L'ELETTRONICA A PORTATA DI CLICK** E RESTARE SEMPRE AGGIORNATO SULL'ELETTRONICA EMBEDDED, I MICROCONTROLLORI E L'INNOVAZIONE TECNOLOGICA



 Elettronica Open Source

+ 145.000

REGISTERED USERS

7.414

 AVERAGE DAILY PAGEVIEWS (FEB2020)

830.610

 2020 ANNUAL VISITORS

THE BIGGEST EMBEDDED COMMUNITY IN ITALY

SOCIAL CONNECTIONS

 + 83.000

 + 23.000

CATEGORIES

PROFESSIONALS

53 %

ACADEMICS/STUDENTS

25 %

MAKERS/HOBBYISTS

22 %

