

AKw10 Architektura mikroprocesora typu ARM

ARM = Advanced RISC Machines

Międzynarodowa firma, mająca główną siedzibę w Cambridge w Wielkiej Brytanii.

Projektuje i sprzedaje licencje na rdzenie ARM - nie produkuje *krzemu (układów)*.

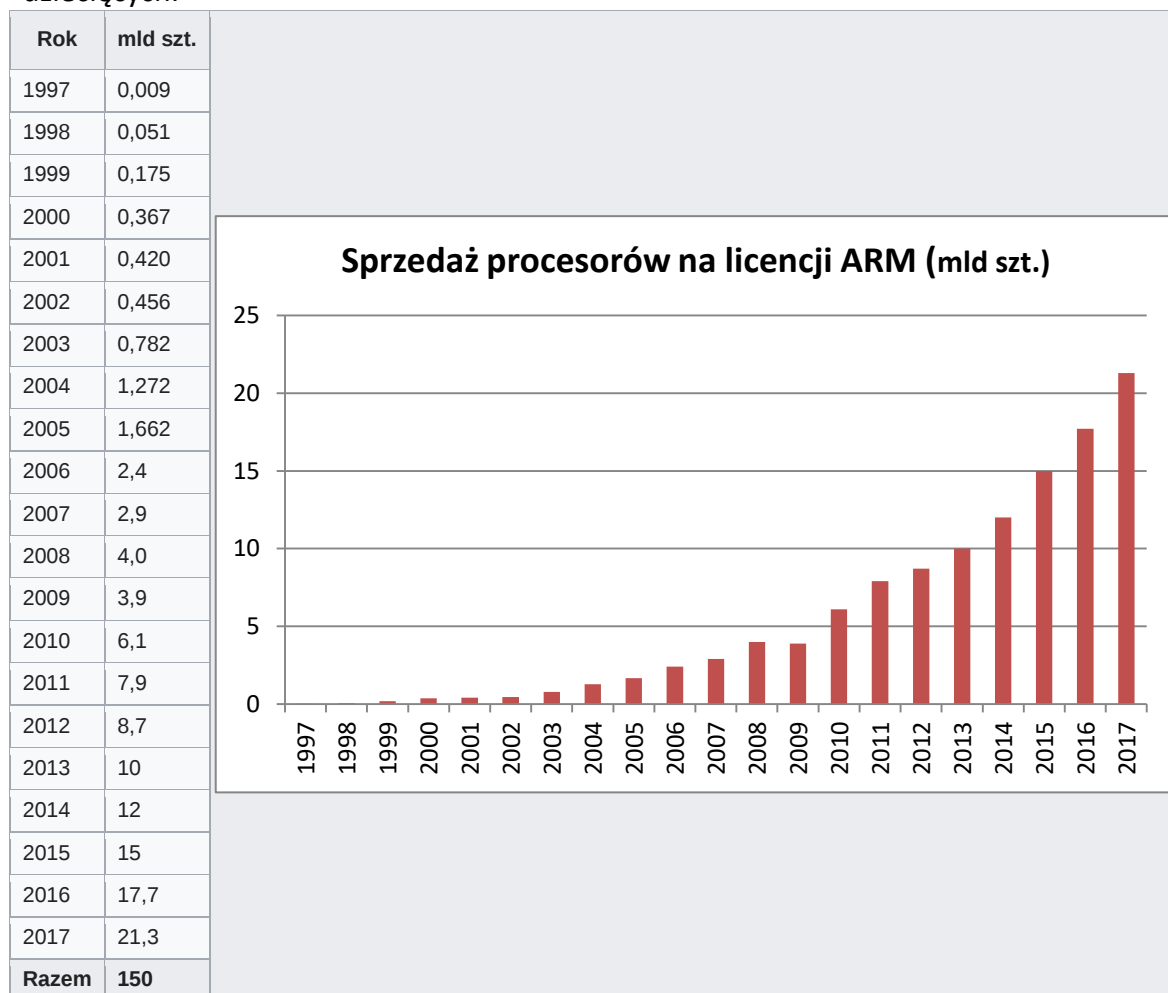
Większość urządzeń mobilnych, w tym głównie telefonów komórkowych, i systemów wbudowanych zawiera rdzenie ARM

Architektura ARM (Advanced RISC Machine, pierwotnie Acorn RISC Machine) – 32-bitowa oraz 64-bitowa (Apple A7, 2013 r.) architektura (model programowy) procesorów typu RISC.

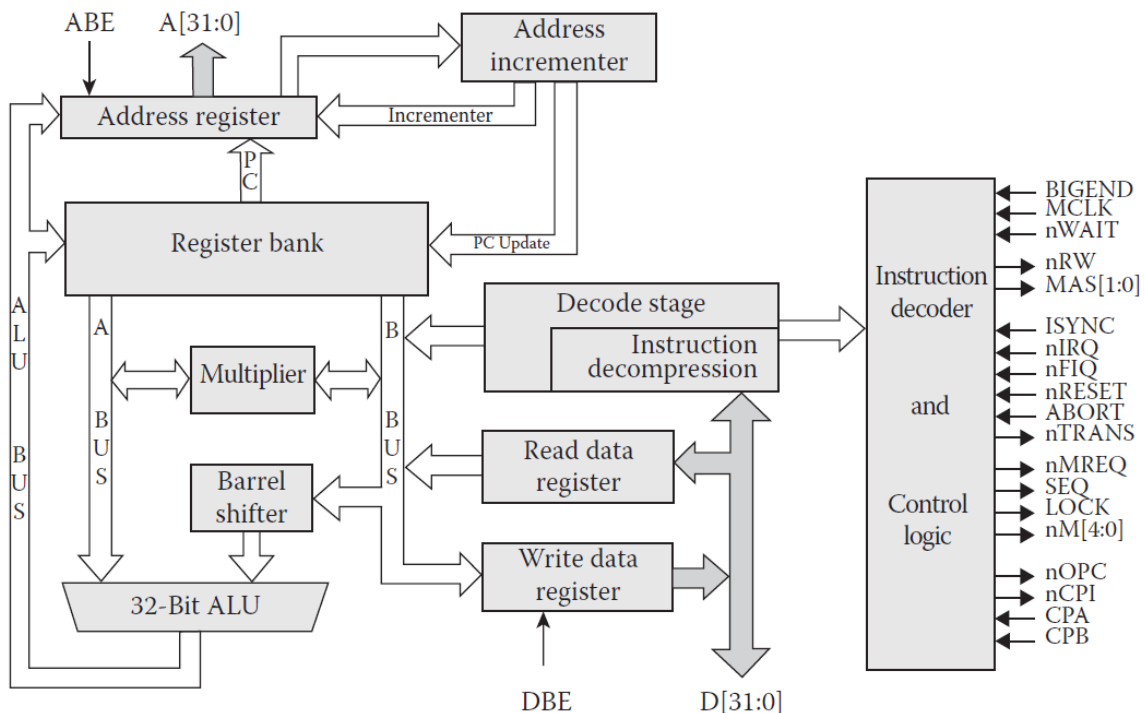
Różne wersje rdzeni ARM są szeroko stosowane w systemach wbudowanych i systemach o niskim poborze mocy, ze względu na ich energooszczędną architekturę.

Architektura ARM

Obecnie ponad 75% rynku 32-bitowych mikroprocesorów stosowanych w systemach wbudowanych stanowią procesory z architekturą ARM. Używa się ich między innymi w telefonach komórkowych, routerach, systemach przemysłowych, a nawet w zabawkach dziecięcych.



https://en.wikipedia.org/wiki/Arm_Holdings



Schemat blokowy procesora ARM7TDMI

T – skompresowany zestaw instrukcji THUMB

D – debug – sprzęt do obsługi zewnętrznego debugera

M – wbudowany układ mnożący (do realizacji algorytmów DSP)

I – In-Circuit Emulation

Barrel shifter – cykliczny rejestr przesuwany

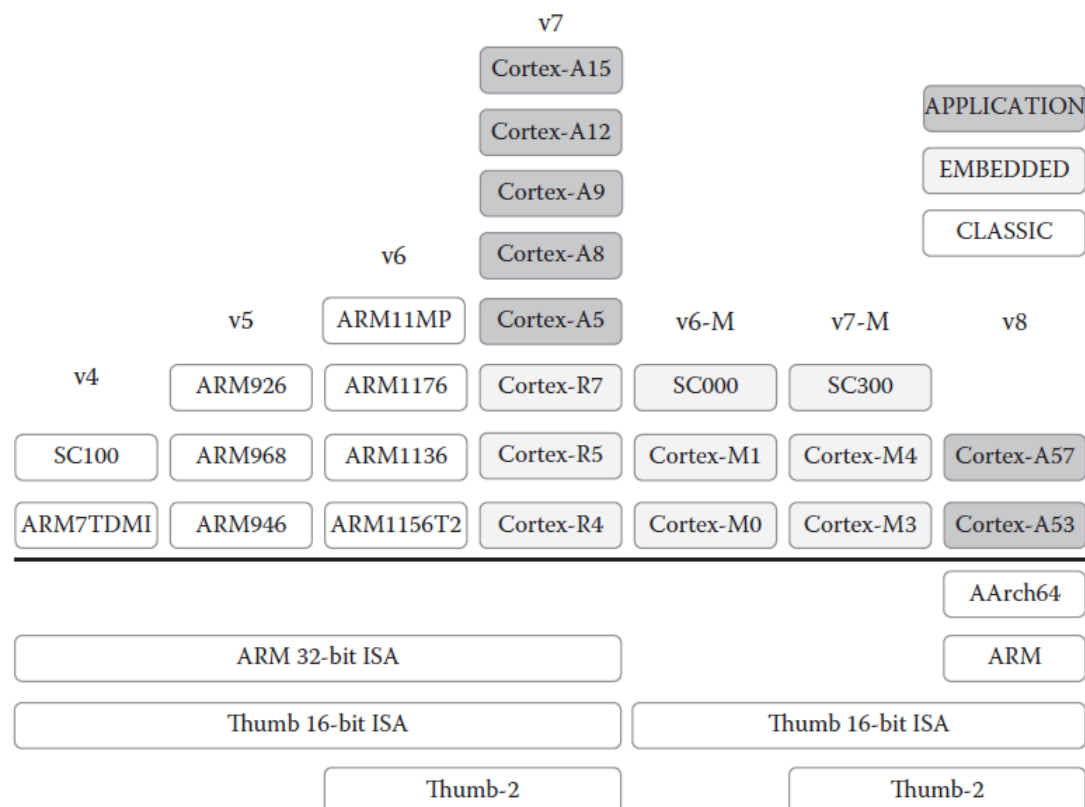
ARM1136JF-S

implementacja ARM11

zbiór instrukcji ISA 6

obsługa sprzętowa środowiska Java oraz operacji zmiennoprzecinkowych

Wersje architektury ARM https://en.wikipedia.org/wiki/List_of_ARM_microarchitectures



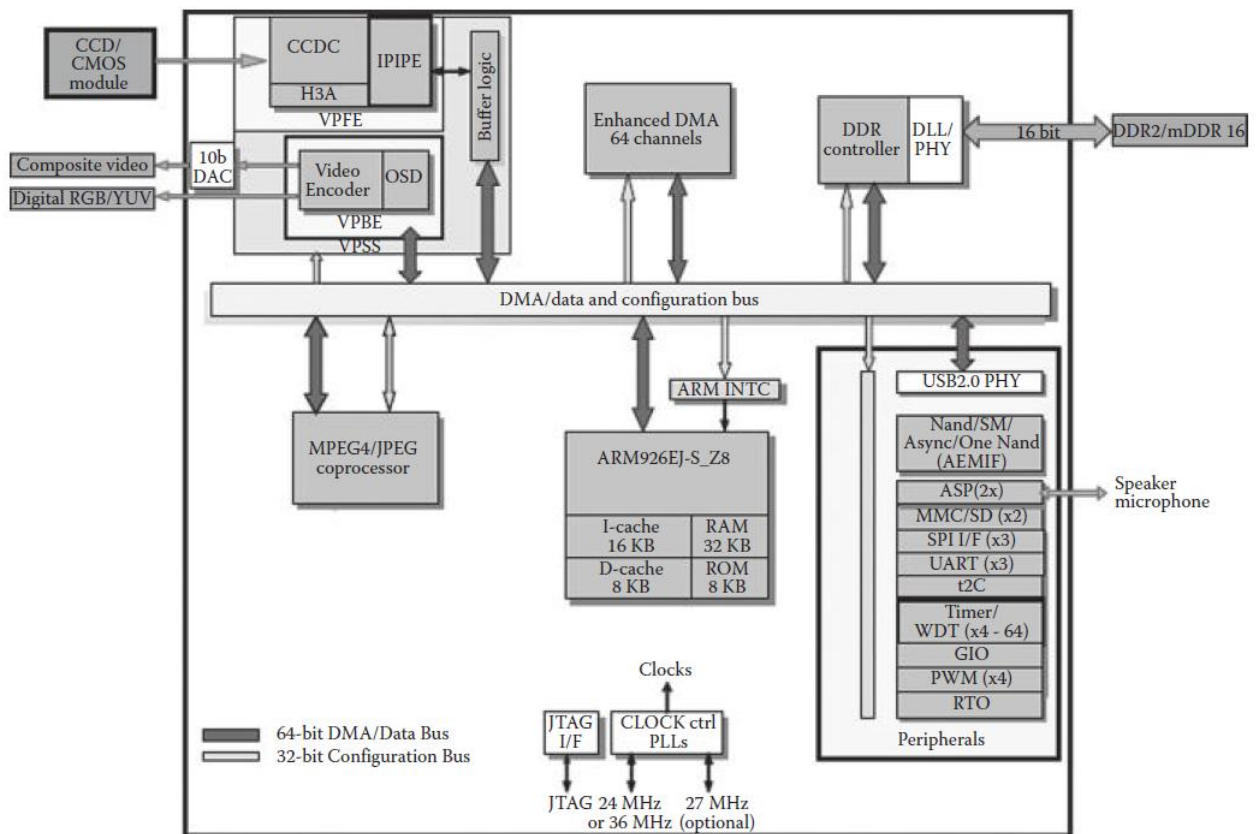
Wersje architektury ARM

Procesory ARM wyposażone w rdzenie **Cortex** zostały podzielone na trzy grupy:

- **A** przeznaczona do implementacji aplikacji z systemami operacyjnymi, wymagających dużych mocy obliczeniowych i obsługi pamięci wirtualnej z MMU (Memory Management Unit),
- **R** stworzona do systemów czasu rzeczywistego, w których czas reakcji jest krytyczny (np. ABS),
- **M** oferująca bardzo korzystny stosunek ceny do możliwości, przeznaczona do zastosowań konsumenckich i przemysłowych.

Procesory ARM wspierają cztery różne zestawy instrukcji:

- Thumb stworzony w celu zwiększenia gęstości kodu, stosuje instrukcje 16-bitowe, ograniczoną różnorodność operandów oraz ograniczony zakres dostępnych rejestrów,
- ARM32 zawiera instrukcje 32-bitowe, większy zakres wartości operandów i zakres skoków relatywnych oraz dostęp do wszystkich rejestrów,
- Thumb-2. stanowi kompromisem pomiędzy Thumb i ARM32, zawiera cały zestaw Thumb i wybrane instrukcje ARM32 (celem jest duża gęstość, zbliżona do Thumb i wydajność porównywalna z ARM 32).
- ARM64 zawiera zestaw instrukcji 64-bitowych wykorzystywany w procesorach o największych mocach obliczeniowych, stosowanych np. w serwerach.



System-on-Chip TMS320DM355 Texas Instruments

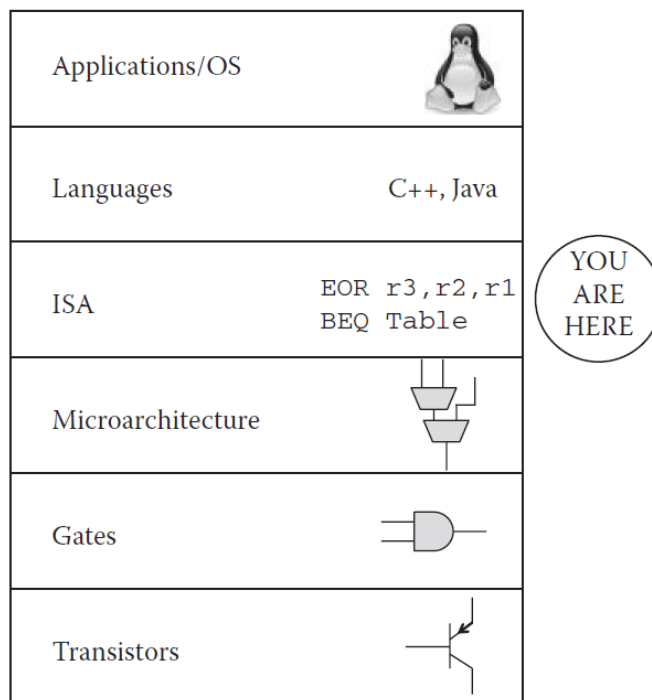
Komputer SoC (ang. System-on-Chip)

Układ scalony wielkiej integracji zawierający **kompletny system elektroniczny zintegrowany z układami analogowymi, cyfrowo-analogowymi oraz radiowymi**. Poszczególne moduły tego systemu, ze względu na ich złożoność, pochodzą zwykle od różnych dostawców, np. rdzeń procesora od jednego producenta, układy peryferyjne od innego, interfejsy od jeszcze innego, itd.

Typowym obszarem zastosowań SoC są systemy wbudowane, a najbardziej rozpowszechnionym przedstawicielem tego rozwiązania są systemy oparte na procesorach ARM.

W przypadku, gdy nie jest możliwa integracja wszystkich układów na jednym podłożu półprzewodnikowym, poszczególne moduły wykonuje się na osobnych krysztalach, a całość zamyka się w jednej obudowie, SiP (ang. System-in-a-package).

SoC różni się od mikrokontrolerów **znacznie wydajniejszą jednostką obliczeniową CPU** (pozwalającą uruchamiać systemy operacyjne, np. Linux, Windows) oraz są zwykle **wyposażone w specjalizowane układy peryferyjne**.



Hierarchia opisu systemu komputerowego

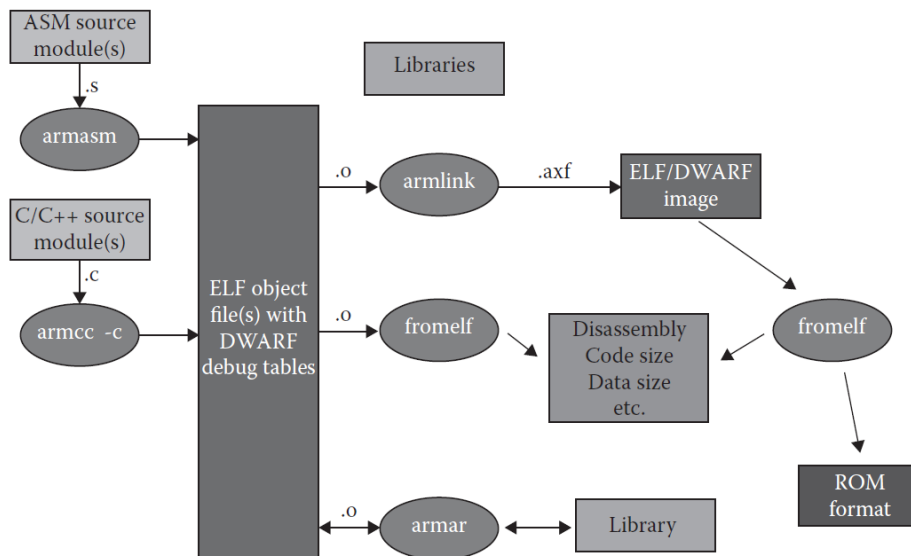


FIGURE 1.13 Tools flow.

Etapy przetwarzania kodu

Typy danych w procesorze ARM:

- słowo – 32 bity
- półsłowo – 16 bitów
- bajt – 8 bitów

Tryby pracy procesora

Wersja 4T obsługuje 7 trybów procesora: User, FIQ, IRQ, Supervisor, Abort, Undefined oraz System.

Tryby wyjątków	Tryb	Opis	
	Supervisor (SVC)	Używany po resecie oraz po wykonaniu instrukcji przerwania programowego (SWI)	Tryby uprzywilejowane
	FIQ	Używany po zgłoszeniu przerwania o wysokim priorytecie (szybkiego)	
	IRQ	Używany po zgłoszeniu przerwania o niskim priorytecie (zwykłego)	
	Abort	Używany do obsługi nieprawidłowego dostępu do pamięci i pamięci wirtualnej	
	Undef	Używany do obsługi nieprawidłowej instrukcji (lub instrukcji zmiennoprzecinkowej jeśli procesor takich nie obsługuje)	
	System	Tryb uprzywilejowany korzystający z tych samych rejestrów co tryb User	
	User	Tryb, w którym pracuje większość aplikacji i systemów operacyjnych	Tryb nieuprzywilejowany

Tryby pracy procesora

Mode					
User/System	Supervisor	Abort	Undefined	Interrupt	Fast interrupt
R0	R0	R0	R0	R0	R0
R1	R1	R1	R1	R1	R1
R2	R2	R2	R2	R2	R2
R3	R3	R3	R3	R3	R3
R4	R4	R4	R4	R4	R4
R5	R5	R5	R5	R5	R5
R6	R6	R6	R6	R6	R6
R7	R7	R7	R7	R7	R7
R8	R8	R8	R8	R8	R8_FIQ
R9	R9	R9	R9	R9	R9_FIQ
R10	R10	R10	R10	R10	R10_FIQ
R11	R11	R11	R11	R11	R11_FIQ
R12	R12	R12	R12	R12	R12_FIQ
R13	R13_SVC	R13_ABORT	R13_UNDEF	R13_IRQ	R13_FIQ
R14	R14_SVC	R14_ABORT	R14_UNDEF	R14_IRQ	R14_FIQ
PC	PC	PC	PC	PC	PC

CPSR	CPSR	CPSR	CPSR	CPSR	CPSR
	SPSR_SVC	SPSR_ABORT	SPSR_UNDEF	SPSR_IRQ	SPSR_FIQ

 = *banked register*

Organizacja rejestrów

Procesor ARM7TDMI ma 37 rejestrów:

- 30 rejestrów ogólnego zastosowania,
- 6 rejestrów stanu,
- licznik programu.

Rejestry zorganizowane są w banki zależne od trybu pracy procesora. Pozwala to szybko zapamiętać stan procesora.

R13 - wskaźnik stosu SP

R14 – rejestr łączący lub LR – przechowuje adres powrotny procedury

Architektura procesora jest potokowa, co oznacza, że w trakcie pobierania jednej instrukcji inna jest dekodowana, a jeszcze inna jest wykonywana. Adres pobieranej instrukcji znajduje się w liczniku programu.

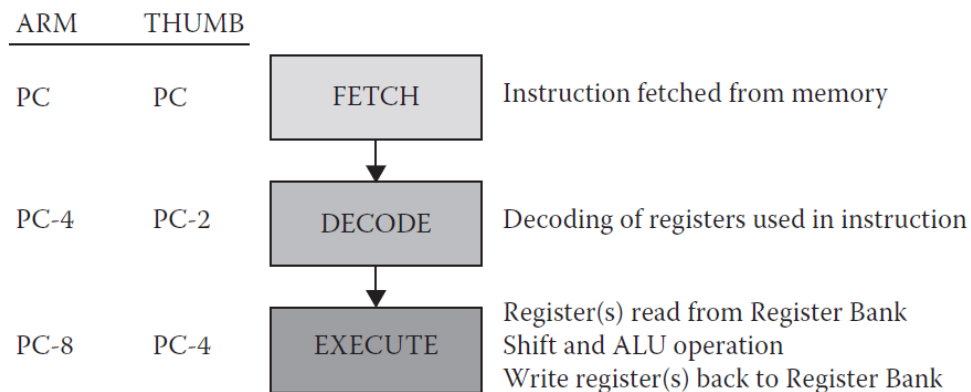
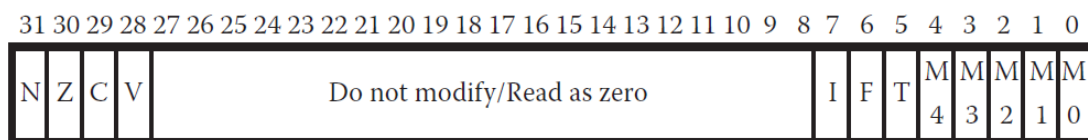


Diagram potoku procesora ARM7TDMI

Rejestry stanu programu

Są to rejestry **Current Program Status Register** oraz **Saved Program Status Register**



Format rejestrów statusowych

N, Z, C, V – bity znaczników warunków kodu – modyfikowane przez instrukcje arytmetyczne i logiczne. Mogą być używane przez wszystkie instrukcje procesora do warunkowego wykonywania kodu co zwiększa gęstość kodu i szybkość działania.

I, F – bity zezwalające na przerwania IRQ oraz FIQ

T – oznacza wykonywanie kodu THUMB, złożonego z instrukcji 16 bitowych; ustawiany specjalną instrukcją procesora

Bity trybu:

The Mode Bits

xPSR[4:0]	Mode
10000	User mode
10001	FIQ mode
10010	IRQ mode
10011	Supervisor mode
10111	Abort mode
11011	Undefined mode
11111	System mode

Tabela wektorów wyjątków.

Składa się z wybranych adresów pamięci , które przechowują informacje niezbędne do obsługi danego wyjątku, przerwania lub innego nietypowego zdarzenia, np. gdy pojawi się przerwanie IRQ procesor zmienia licznik programu na 0x18 i zaczyna pobierać instrukcje z tego adresu.

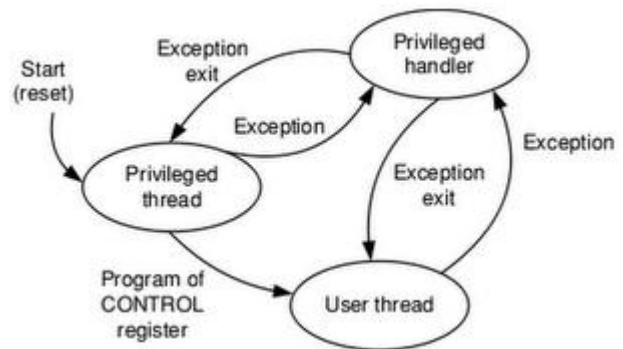
ARM7TDMI Exception Vectors

Exception Type	Mode	Vector Address
Reset	SVC	0x00000000
Undefined instruction	UNDEF	0x00000004
Software Interrupt (SVC)	SVC	0x00000008
Prefetch abort (instruction fetch memory abort)	ABORT	0x0000000C
Data abort (data access memory abort)	ABORT	0x00000010
IRQ (interrupt)	IRQ	0x00000018
FIQ (fast interrupt)	FIQ	0x0000001C

Procesor Cortex M4

	Privileged	User
Handler mode	Use: Exception handling Stack: Main	
Thread mode	Use: Applications Stack: Main or Process	Use: Applications Stack: Main or Process

Cortex-M4 modes.



Tryby pracy rdzenia Cortex-M

Cortex-M4 posiada tylko dwa tryby pracy: **Handler mode** i **Thread mode**. Jak pokazano na rysunku powyżej, istnieją również dwa poziomy dostępu do przełączania między tymi trybami, Uprzywilejowany (Privileged) i Użytkownika (User), i w zależności od tego co system robi, przełącza te tryby przez ustawienie odpowiedniego bitu w rejestrze sterującym. Dla bardzo prostych aplikacji procesor pozostaje na jednym poziomie dostępu – może nigdy nie przechodzić w tryb użytkownika. W przypadku kiedy używany jest wbudowany system operacyjny, taki jak SYS/BIOS, sterujący wszystkim, bezpieczeństwo ma duże znaczenie i oddzielany jest obszar rdzenia od obszaru użytkownika.

Rejestry procesora Cortex-M4.

Procesor ten posiada mniejszą liczbę rejestrów niż ARM7TDMI, lecz te same 16 rejestrów występują w trybie użytkownika jak w ARM7TDMI. Cortex-M4 zawierający jednostkę zmiennoprzecinkową, oprócz rejestrów związanych z układami peryferyjnymi, posiada następujące rejestry:

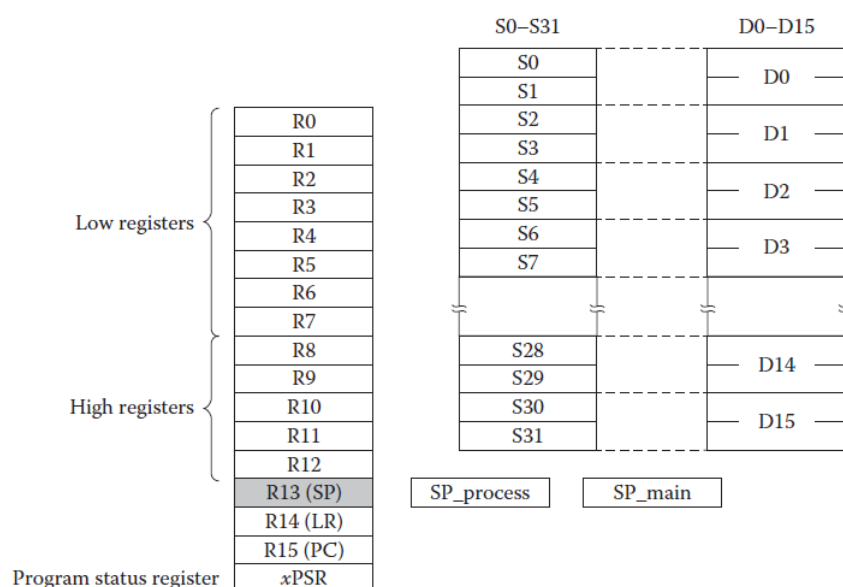


FIGURE 2.6 Cortex-M4 with floating-point register organization.

- 17 rejestrów ogólnego przeznaczenia, tj., przechowujących dowolną wartość
- Rejestr Statusowy (status register)
- 3 rejestry maskowania przerwań
- Rejestr sterujący (control register)
- 32 rejestry zmiennoprzecinkowe pojedynczej precyzji (s0–s31) lub 16 rejestrów podwójnej precyzji (d0–d15) albo ich kombinację
- 4 rejestry sterujące operacji zmiennoprzecinkowych (znajdujące się w pamięci, nie w fizycznych rejestrach)

Rejestry r0 do r12 są rejestrami ogólnego przeznaczenia, przechowują dane 32 bitowe takie jak adresy, dane, dane spakowane, dane ułamkowe itp.

Niektóre rejestry mają specjalne przeznaczenie, takie jak r13, wskaźnik wierzchołka stosu (dwa rejestry pozwalające pracować z dwoma niezależnymi stosami); rejestr r14, rejestr łączący (Link Register) i rejestr r15, licznik programu (Program Counter).

Podobnie jak w ARM7TDMI, rejestr r13 (wskaźnik stosu, SP) przechowuje adres stosu w pamięci operacyjnej, lecz tutaj są dwa takie rejestry: Main Stack Pointer (MSP) i Process Stack Pointer (PSP).

Są dwa stosy:

- MSP dla trybów uprzywilejowanych (ang. *privileged thread, privileged handler*),
- PSP dla trybu użytkownika (ang. *user thread*).

Rejestr r14 (Link Register, LR) służy do przechowywania adresów powrotu z procedury lub obsługi wyjątku. W przeciwieństwie do ARM7TDMI, występuje tutaj tylko jeden Link Register.

Rejestr r15, Program Counter lub PC, wskazuje instrukcję, która będzie pobrana.

Rejestr statusowy.

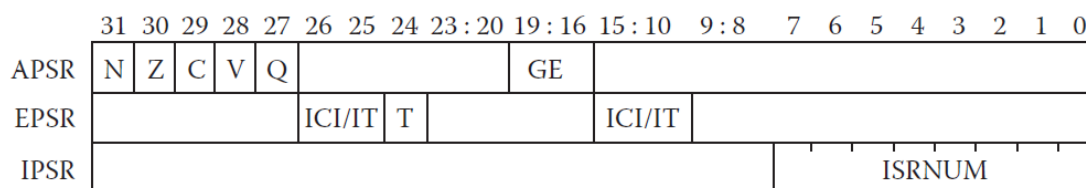


FIGURE 2.7 Program status registers on the Cortex-M4.

Program Status Register (xPSR), pełni tą samą rolę co rejestr CPSR w ARM7TDMI lecz posiada inne pola. Rejestr ten może być dostępny w całości lub testowany na trzy różne sposoby.

Application Program Status Register (APSR), Interrupt Program Status Register (IPSR) i Execution Program Status Register (EPSR) są trzema szczegółowymi widokami tego rejestru.

APSR zawiera znaczniki statusowe (N, C, V i Z), znacznik Greater Than lub Equal (używane przez instrukcję SEL), i dodatkowy znacznik „sticky” Q używany w arytmetyce z nasyceniem, gdzie „sticky” oznacza, że bit ten może być wyzerowany tylko przez jawne wpisanie zera do niego.

IPSR zawiera numer wyjątku wykorzystywany do obsługi błędów i innych typów wyjątków.

Dwa pola EPSR określają status instrukcji IF-THEN w złożeniu z bitami Interrupt-Continuable Instruction (ICI), zaś bit T określa tryb Thumb (T).

Rejestry maskowania przerwań, PRIMASK, FAULTMASK, and BASEPRI są używane do maskowania różnego rodzaju przerwań i wyjątków. PRIMASK i FAULTMASK są rejestrami jednobitowymi. BASEPRI ma szerokość 8 bitów i zawiera priorytet obsługiwanego przerwania.

Ostatnim rejestrem specjalnego przeznaczenia jest rejestr CONTROL zawierający tylko trzy bity. Bit najmniej znaczący, CONTROL[0], zmienia poziom dostępu w trybie Thread mode na Privileged lub User level. Następny bit, CONTROL[1], wybiera jeden ze wskaźników wierzchołka stosu: Main Stack Pointer (MSP) lub Process Stack Pointer (PSP). Najbardziej znaczący bit, CONTROL[2], wskazuje czy zachowywać stan operacji zmiennoprzecinkowych podczas przetwarzania wyjątków.

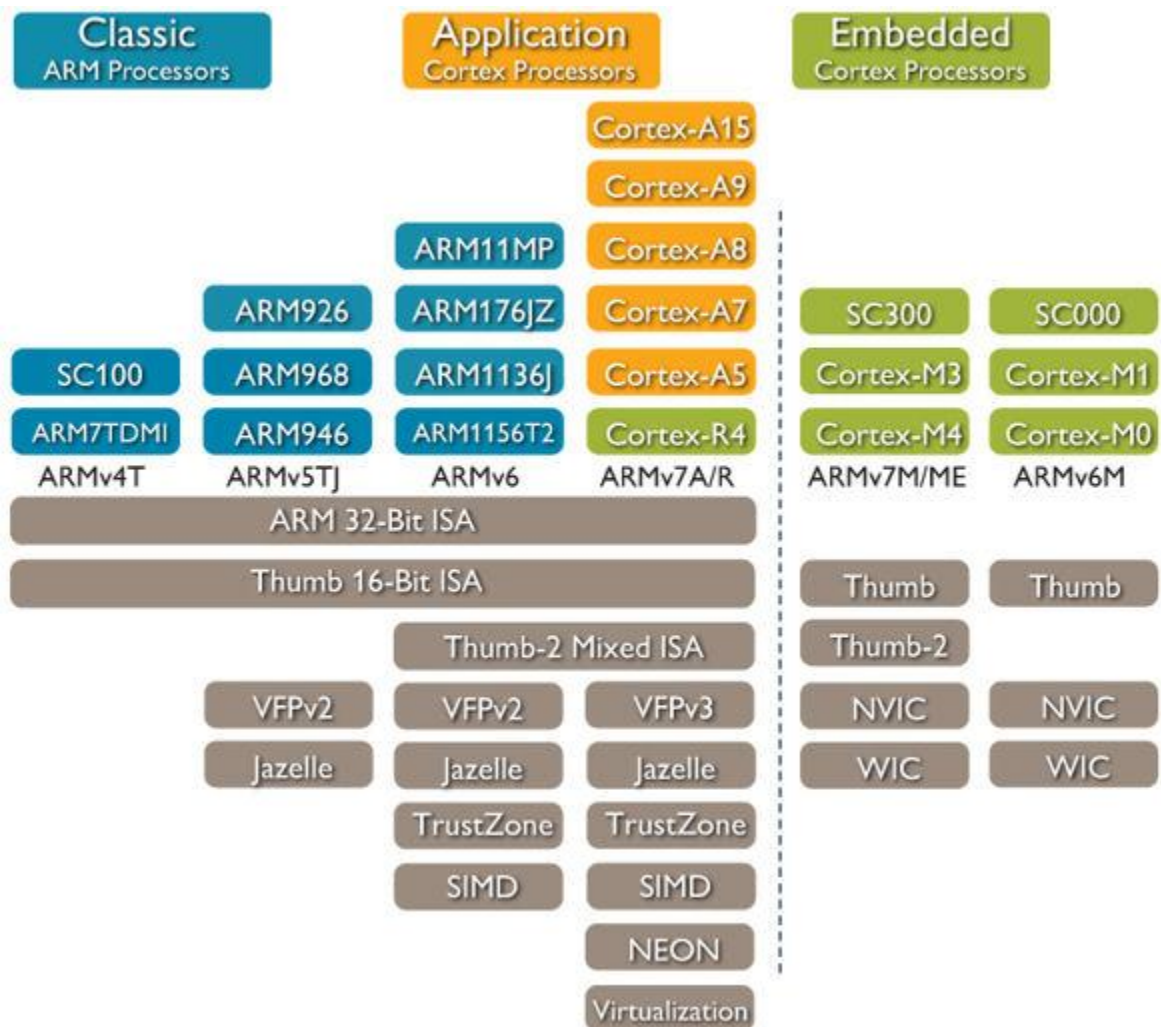
Tablica wektorów przerwań

Najbardziej znaczącą zmianą w procesorze Cortex-M4 w porównaniu z wcześniejszymi procesorami ARM jest powrót do stosowanej w mikroprocesorach 8051 koncepcji przechowywania w tablicy wektorów przerwań adresów procedur obsługi przerwań a nie instrukcji wykonywanej po przerwaniu.

TABLE 2.3
Cortex-M4 Exception Vectors

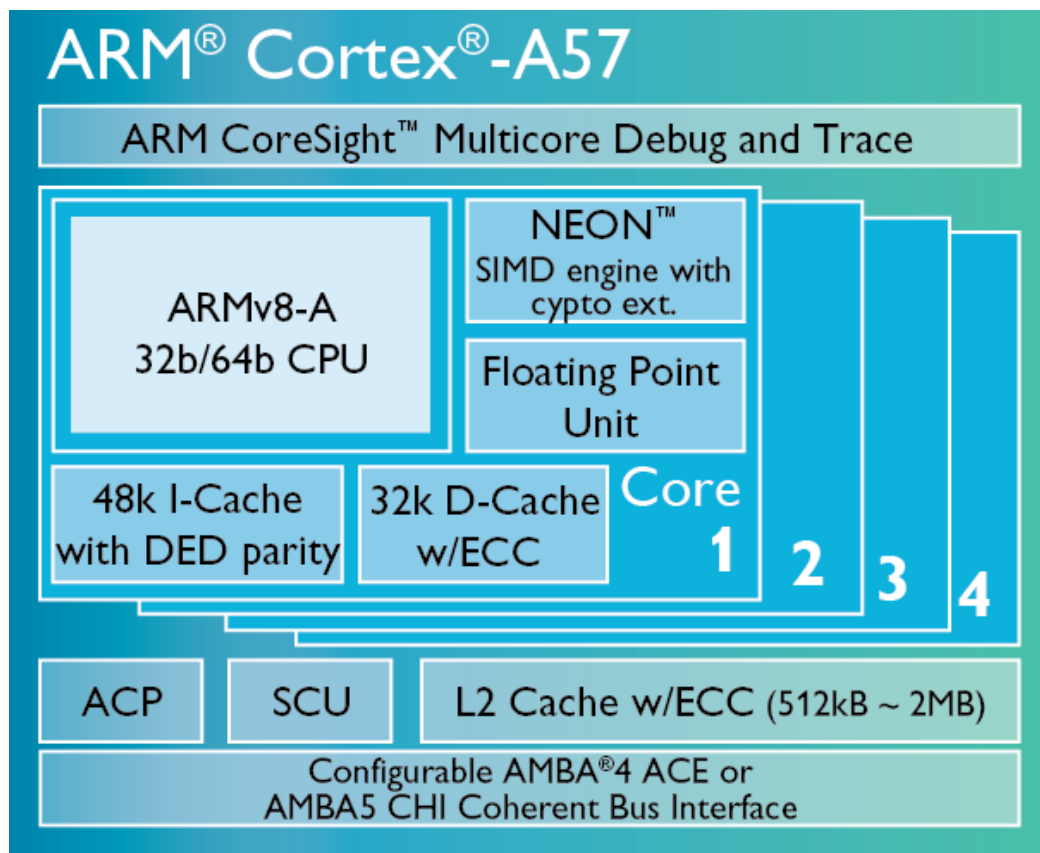
Exception Type	Exception No.	Vector Address
(Top of Stack)	—	0x00000000
Reset	1	0x00000004
NMI	2	0x00000008
Hard fault	3	0x0000000C
Memory management fault	4	0x00000010
Bus fault	5	0x00000014
Usage fault	6	0x00000018
SVcall	11	0x0000002C
Debug monitor	12	0x00000030
PendSV	14	0x00000038
SysTick	15	0x0000003C
Interrupts	16 and above	0x00000040 and above

Rozwój 32-bitowej architektury ARM



ARM Cortex-A50

Procesory Cortex-A57 i Cortex-A53



ACP: Accelerator Coherency Port – układ zapewniający spójność pamięci (w tym przypadku podręcznej L2)

SCU: Snoop Control Unit – układ zapewniający spójność pamięci (w tym przypadku podręcznej L2)

ECC: Error Correction Code – kod korygujący błędy w pamięci półprzewodnikowej

SEC: Single Error Correct – możliwość skorygowania przekłamania jednego bitu

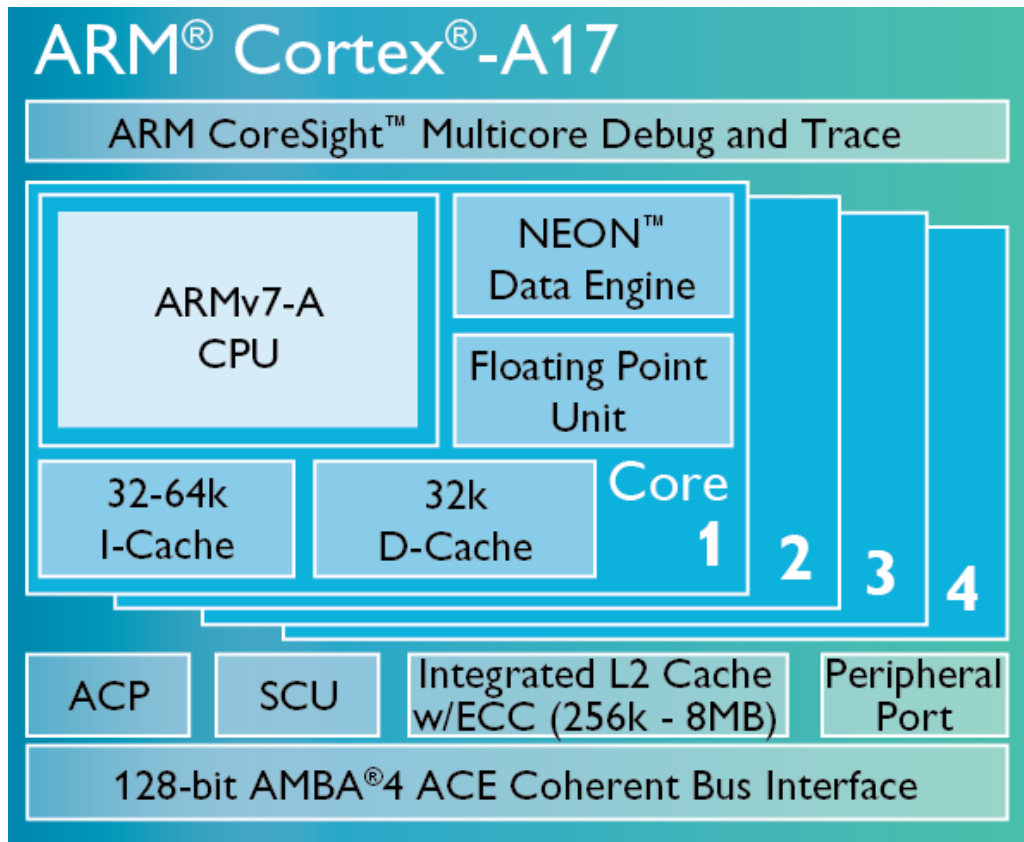
DED: Double Error Detect – możliwość wykrycia dwóch przekłamanych bitów

AMBA 5: Advanced Microcontroller Bus Architecture – szyna wraz z zestawem protokołów do komunikacji między układami cyfrowymi, piąta wersja specyfikacji

CHI: Coherent Hub Interface

ARM Cortex-A

Procesory Cortex-A17, Cortex-A15, Cortex-A12, Cortex-A9, Cortex-A8, Cortex-A7, Cortex-A5



AMBA 4: czwarta wersja specyfikacji

ACE: AXI Coherency Extensions

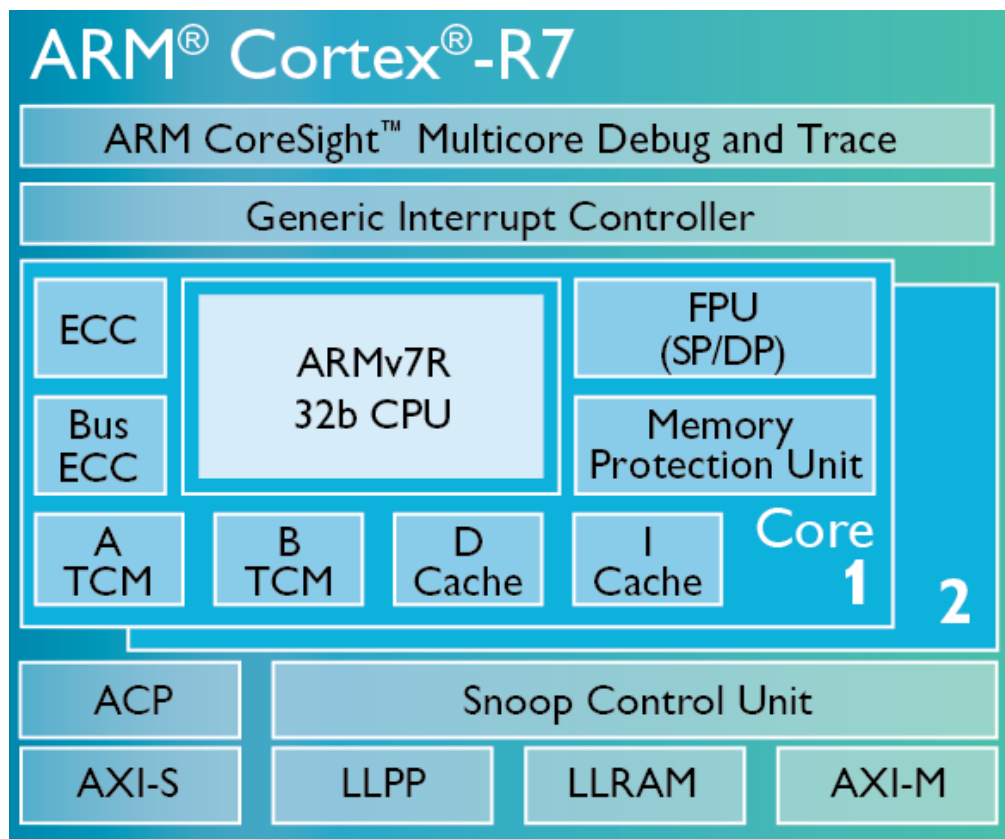
AXI: Advanced eXtensible Interface

Rdzenie Cortex-A przeznaczone są do stosowania w urządzeniach wymagających dużych mocy obliczeniowych i umiarkowanego zużycia energii

- smartfony
- tablety

ARM Cortex-R

Procesory Cortex-R7, Cortex-R5, Cortex-R4



FPU: Floating Point Unit – jednostka zmiennoprzecinkowa

SP: Single Precision – pojedyncza precyzja (32 bity)

DP: Dual Precision – podwójna precyzja (64 bity)

MPU: Memory Protection Unit – układ umożliwiający separację procesów w pamięci operacyjnej

TCM: Tightly Coupled Memory – pamięć umieszczona blisko procesora, przyspieszająca obsługę przerw

AMBA 3: trzecia wersja specyfikacji

AXI-S: Advanced eXtensible Interface Slave

AXI-M: Advanced eXtensible Interface Master

LLPP: Low Latency Peripheral Port – port wejścia-wyjścia o krótkim czasie dostępu

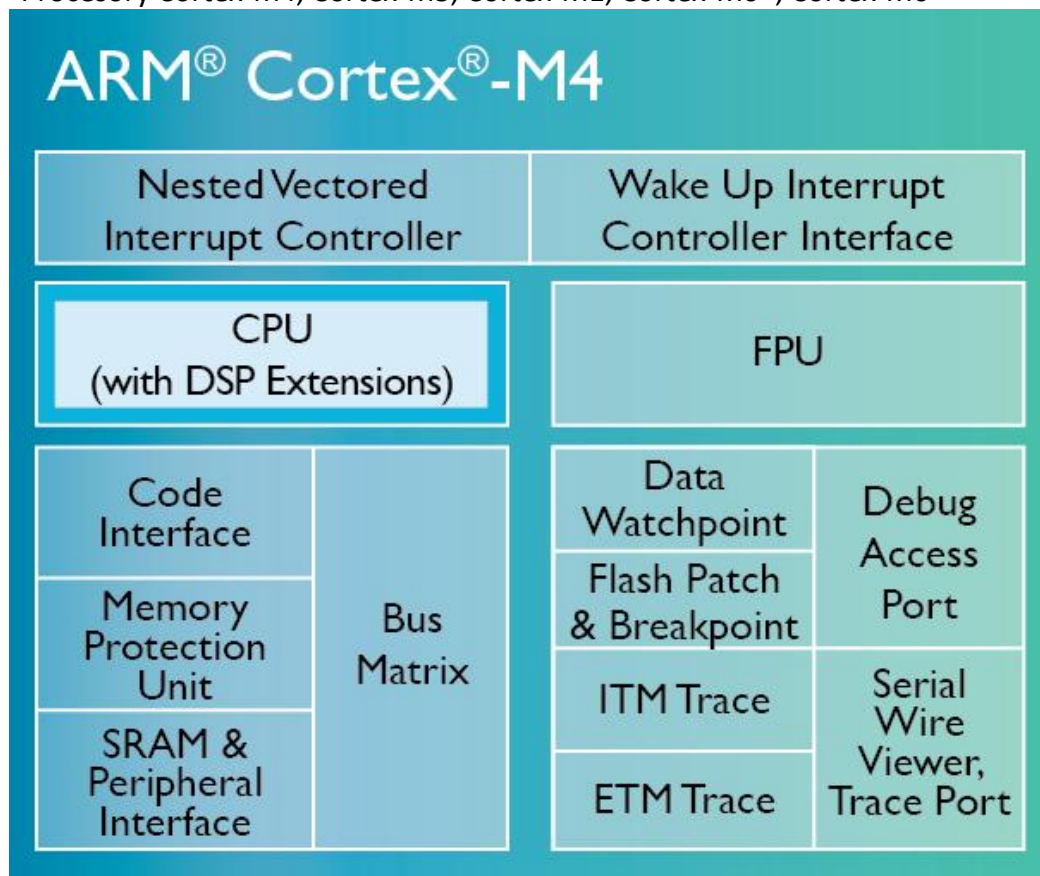
LLRAM: Low Latency RAM – pamięć RAM o krótkim czasie dostępu

Uwagi:

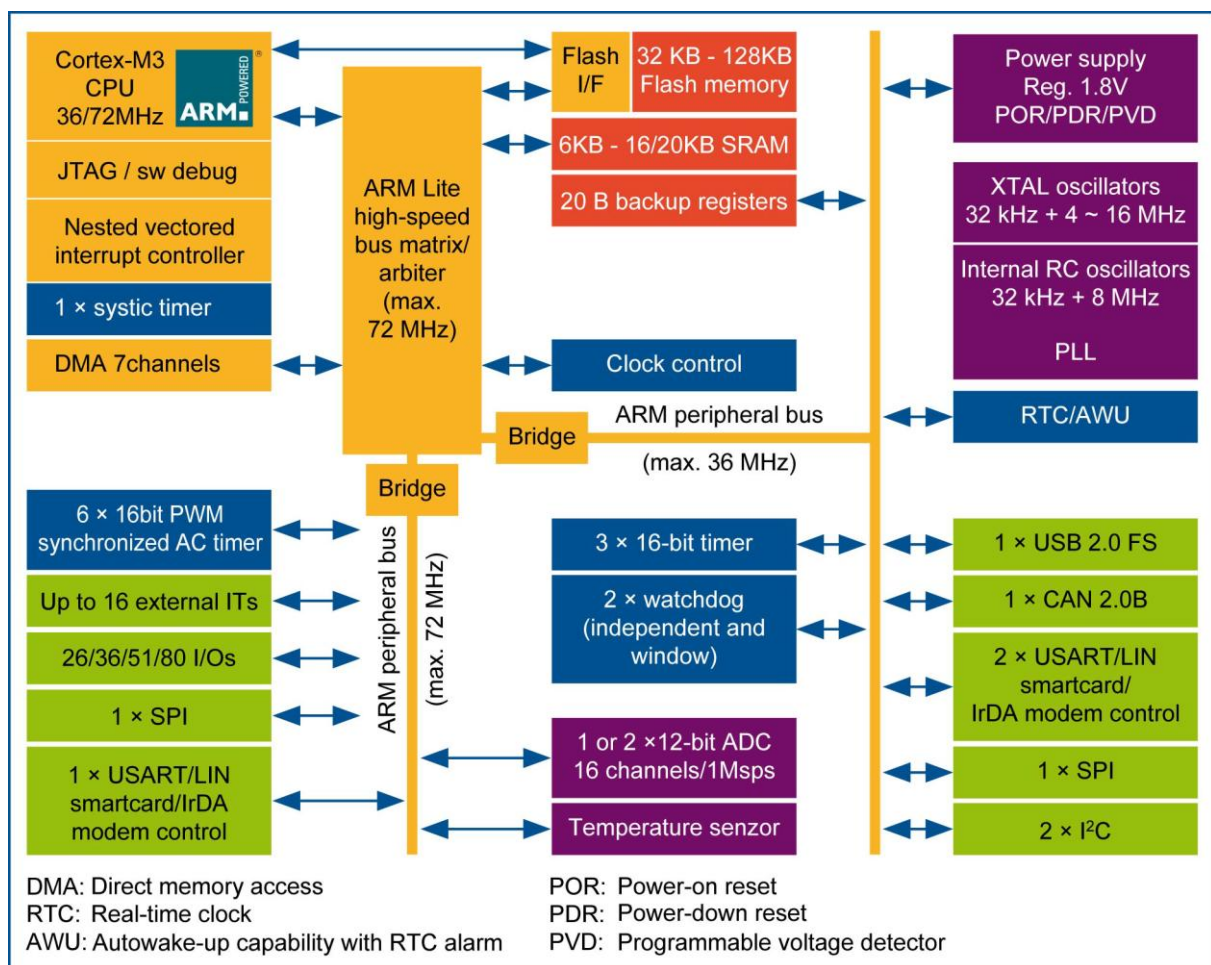
Rdzenie Cortex-R przeznaczone są do stosowania w systemach czasu rzeczywistego (ang. *real time*), gdzie wymagana jest szybka reakcja na zdarzenia, wysoka niezawodność, spora moc obliczeniowa (motoryzacja, lotnictwo, energetyka)

ARM Cortex-M

- Processor Cortex-M4, Cortex-M3, Cortex-M1, Cortex-M0+, Cortex-M0



Przykład mikrokontrolera z rdzeniem Cortex-M3



DSP: Digital Signal Processing – sprzętowe wsparcie dla cyfrowego przetwarzania sygnałów, np. operacje z nasyceniem, dodawanie z akumulacją itp.

ITM: Instrumentation Trace Macrocell – sprzętowe wsparcie odpluskwiania

ETM: Embedded Trace Macrocell – sprzętowe wsparcie odpluskwiania

AMBA 2: druga wersja specyfikacji

AHB: Advanced High-performance Bus

APB: Advanced Peripheral Bus

Uwagi

Cortex-M4 przeznaczony do systemów wbudowanych wymagających dużej wydajności

FPU SP i DSP tylko w tym rdzeniu

Cortex M3 przeznaczony do szerokiego stosowania w systemach wbudowanych

Cortex-M1 przeznaczony do implementacji w FPGA (ang. *Field Programmable Gate Array*)

ASIC (ang. *Application Specific Integrated Circuit*)

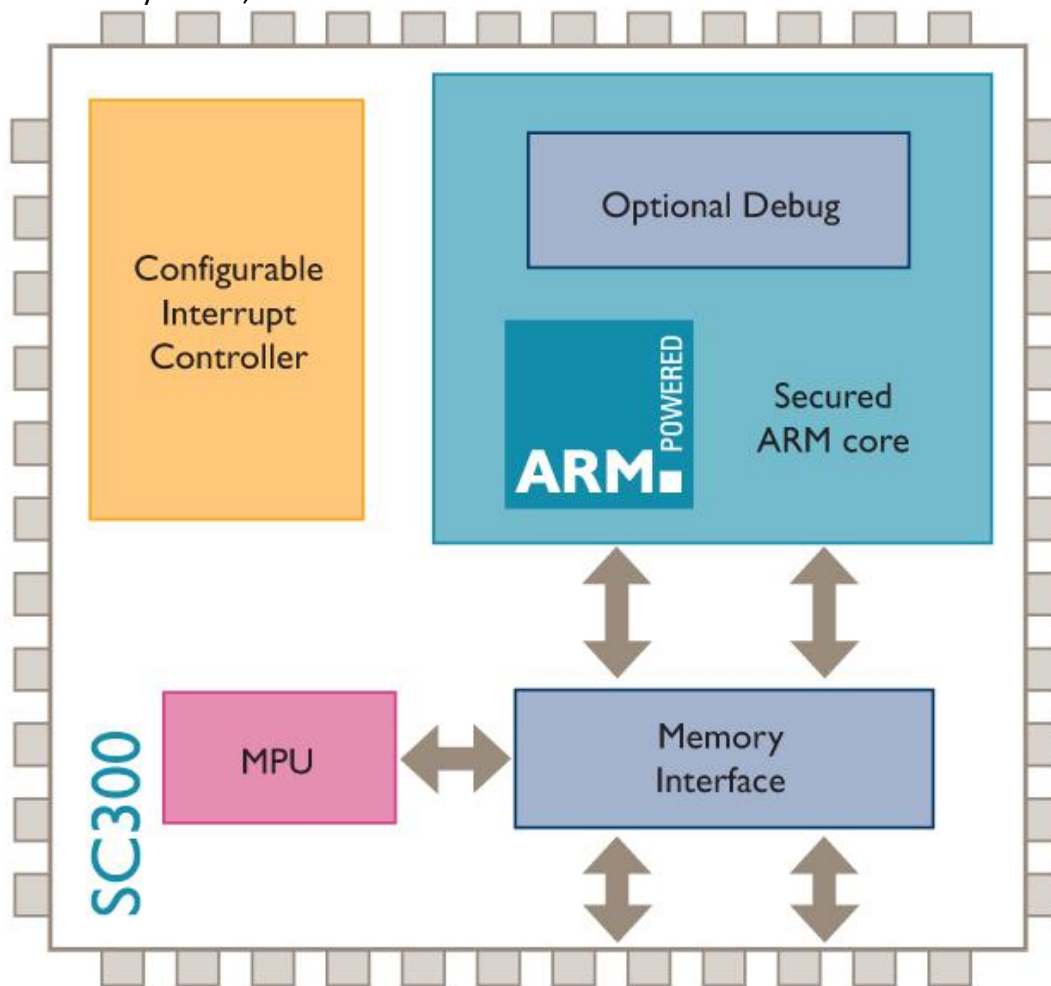
ASSP (ang. *Application Specific Standard Product*)

Cortex-M0+ przeznaczony do urządzeń o małym poborze energii, ograniczony zestaw instrukcji, uproszczona architektura, alternatywa dla architektur 8- i 16-bitowych

Cortex-M0 to poprzednia (mniej udana) wersja Cortex-M0+

ARM SecureCore

- Procesory SC300, SC100 i SC000



Uwagi

SC300 – rdzeń Cortex-M3

SC100 – rdzeń ARM7TDMI

SC000 – rdzeń Cortex-M0

Główny obszar zastosowań to karty czipowe

- SIM
- płatnicze
- do dekodérów TV
- identyfikacyjne

Raspberry Pi

Family	Model	Form Factor	Ethernet	Wireless	GPIO	Released	Discontinued
Raspberry Pi 1	B	Standard	Yes	No	26-pin	2012	Yes
	A	(85.60 × 56.5 mm)	No			2013	Yes
	B+		Yes		40-pin	2014	
	A+	Compact (65 × 56.5 mm)	No			2014	
Raspberry Pi 2	B	Standard	Yes	No		2015	
Raspberry Pi Zero	Zero	Zero (65 × 30 mm)	No	No		2015	
	W/WH		Yes	2017			
Raspberry Pi 3	B	Standard	Yes	Yes		2016	
	A+	Compact	No			2018	
	B+	Standard	Yes			2018	
Raspberry Pi 4	B (1GB)	Standard	Yes	Yes		2019	
	B (2GB)						
	B (4GB)						

Model Raspberry Pi	SoC	CPU
Model A	Broadcom BCM2835 (CPU + GPU + DSP + SDRAM)	700 MHz ARM1176JZF-S core (ARM11 family)
Model B		
Model B+		
Model 2B	Broadcom BCM2836 (CPU + GPU + DSP + SDRAM + jeden port USB)	900 MHz <u>quad-core</u> ARM Cortex-A7
Model Zero	Broadcom BCM2835	1 GHz ARM1176JZF-S
Model 3B	Broadcom BCM2837	1,2 GHz <u>quad-core</u> ARM-8 Cortex-A53 (64-bit)
Model 3B+	Broadcom BCM2837	1,4 GHz <u>quad-core</u> ARM-8 Cortex-A53 (64-bit)
Model 4B	Broadcom BCM2711	1,5 GHz <u>quad-core</u> ARM-8 Cortex-A72 (64-bit)

Raspberry Pi 4 B (najnowszy)



Typ Komputer jednopłytkowy SBC

Premiera	Czerwiec 2019 r.
Koniec produkcji	Styczeń 2026
Procesor	Broadcom BCM2711 quad-core 64-bitowy ARM-8 Cortex-A72 1,5 GHz
Procesor graficzny	Broadcom VideoCore VI
Pamięć operacyjna	1, 2 lub 4 GB
Nośnik danych	Złącze SD Card (karty SD lub SDHC) oraz dowolne urządzenia podłączane za pomocą portu USB
System operacyjny	Linux (Raspbian, Debian GNU/Linux, Ubuntu Mate, Fedora lub Arch Linux), FreeBSD, Windows 10 dla ARM64
poprzednik: Raspberry Pi 3 B+	następca: <i>brak</i>

BCM2711

This is the Broadcom chip used in the Raspberry Pi 4 Model B. The architecture of the BCM2711 is a considerable upgrade on that used by the SoCs in earlier Pi models. It continues the quad-core CPU design of the BCM2837, but uses the more powerful ARM A72 core. It has a greatly improved GPU feature set with much faster input/output, due to the incorporation of a PCIe link that connects the USB 2 and USB 3 ports, and a natively attached Ethernet controller. It is also capable of addressing more memory than the SoCs used before.

The ARM cores are capable of running at up to 1.5 GHz, making the Pi 4 about 50% faster than the Raspberry Pi 3B+. The new VideoCore VI 3D unit now runs at up to 500 MHz. The ARM cores are 64-bit, and while the VideoCore is 32-bit, there is a new Memory Management Unit, which means it can access more memory than previous versions.

The BCM2711 chip continues to use the heat spreading technology started with the BCM2837B0, which provides better thermal management.

A preliminary datasheet for the BCM2711 can be found [here](#).

Some technical details

Processor: Quad-core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5 GHz

Memory: Accesses up to 4GB LPDDR4-2400 SDRAM (depending on model)

Multimedia: H.265 (4Kp60 decode); H.264 (1080p60 decode, 1080p30 encode); OpenGL ES, 3.0 graphics

I/O: PCIe bus, onboard Ethernet port, 2 × DSI ports (only one exposed on Raspberry Pi 4B), 2 × CSI ports (only one exposed on Raspberry Pi 4B), up to 6 × I2C, up to 6 × UART (muxed with I2C), up to 6 × SPI (only five exposed on Raspberry Pi 4B), dual HDMI video output, composite video output.

https://en.wikipedia.org/wiki/Comparison_of_single-board_computers