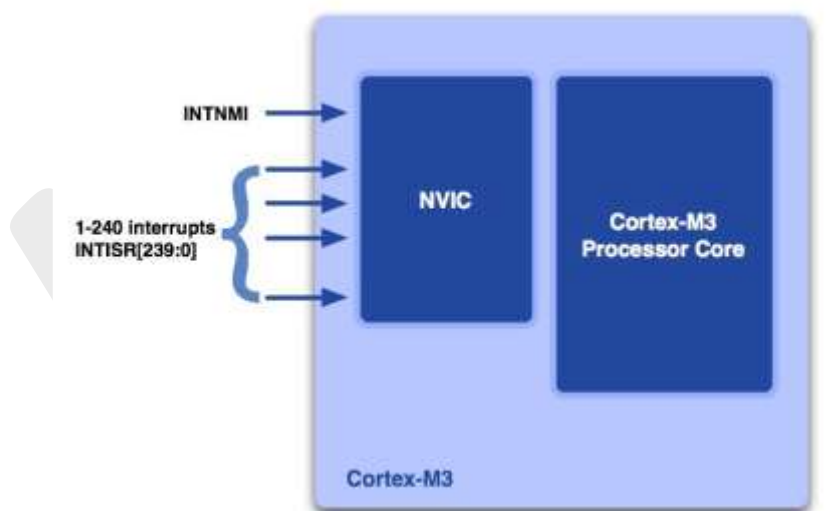


# ARM – Cortex M

## 2.5. Prerušovací systém

Požiadavka na prerušenie je požiadavka externého zariadenia na procesor, aby procesor prerušil vykonávanie aktuálneho programu – postupnosť inštrukcií, vykonal podprogram určený na obsluhu prerušenia (t.j. spracoval požiadavku na prerušenie) a následne pokračoval v prerušenom programe. Súbor takýchto požiadaviek na prerušenie, ich spôsob spracovania a priorít nazývame prerušovací systém.

Prerušovací systém v CORTEX-M jadre je realizovaný štandardnou jednotkou nazývanou radič vnorených prerušení - **Nested Vector Interrupt Controller** –NVIC. To znamená, že mikropočítačové systémy na báze CORTEX-M jadra majú rovnakú štruktúru prerušenia bez ohľadu na výrobcu. Ako už názov napovedá, NVIC je navrhnutý na podporu vnorených prerušení a pre STM32 systém podporuje 16 úrovní priority. NVIC je plne programovateľný a svojou architektúrou spĺňa požiadavky na real-time použitie (t.j. použitie v aplikáciách s presne definovanou odozvou na vznik prerušenia).



NVIC obsahuje:

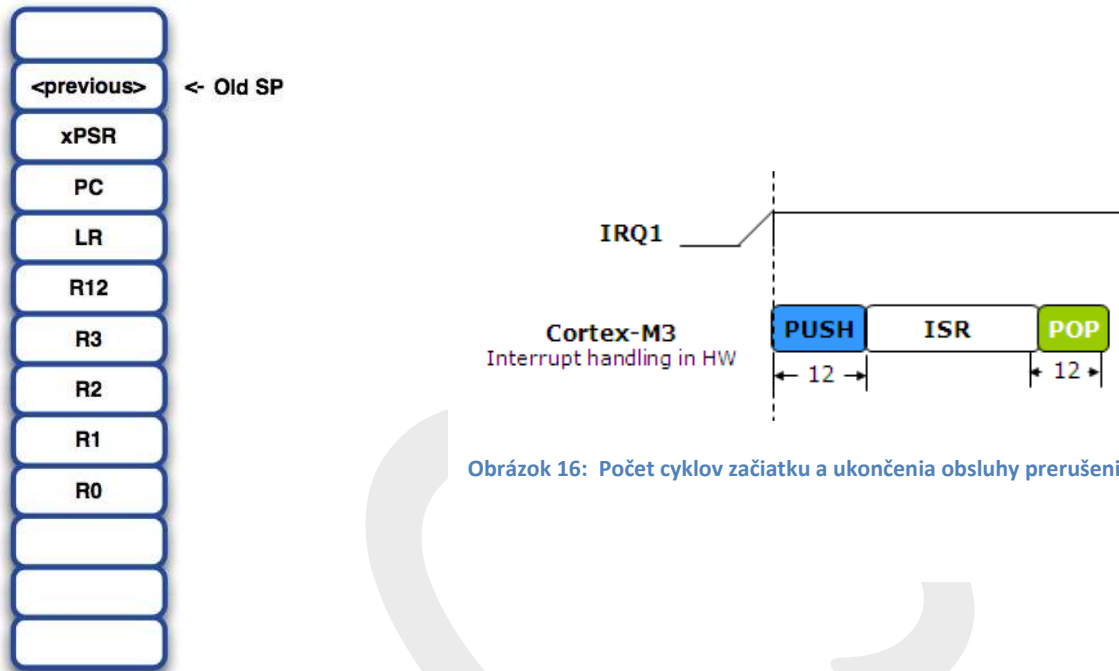
- 1 nemaskovateľné prerušenie (non-maskable interrupt – NMI, pin na puzdre procesora),
- 15 zdrojov prerušenia v CORTEX jadre, ktoré sa používajú na spracovanie výnimiek v rámci samotného jadra, na riadenie pri použití RTOS, atď,
- 1-240 externých prerušení, na ktoré môžu byť pripojené užívateľské periférie.

### 2.5.1. Začiatok a koniec obsluhy prerušenia

Keď príde požiadavka na prerušenie od periférie, NVIC naštartuje CORTEX-M CPU na proces obsluhy prerušenia. Z dôvodu, že CORTEX-M CPU prechádza do režimu prerušenia, je potrebné uchovať informácie o prerušenom programe. Obsah určených registrov sa uloží do zásobníka (PSP alebo MSP). Keď je obsah registrov uložený do zásobníka, vyberie sa cez inštrukčnú zbernicu adresa podprogramu na obsluhu prerušenia.

# ARM – Cortex M

Čas od požiadavky na prerušenie po prvú inštrukciu obslužného podprogramu prerušenia je **12 cyklov**, počas ktorých procesor zrealizuje vyššie uvedené úkony.



Obrázok 16: Počet cyklov začiatku a ukončenia obsluhy prerušenia

Obrázok 17: Registre, ktoré sa pri obsluhu prerušenia ukladajú do zásobníka

## 2.5.2. Pokročilý mód spracovania prerušenia

Rovnako, ako je schopný rýchlo spracovať jedno prerušenie, NVIC je navrhnutý, aby efektívne spracovával viac prerušení v reálnom čase. Má metódy na spracovanie viacerých zdrojov prerušenia s minimálnym oneskorením medzi prerušeniami a na zabezpečenie, aby požiadavky s najvyššou prioritou boli spracované ako prvé.

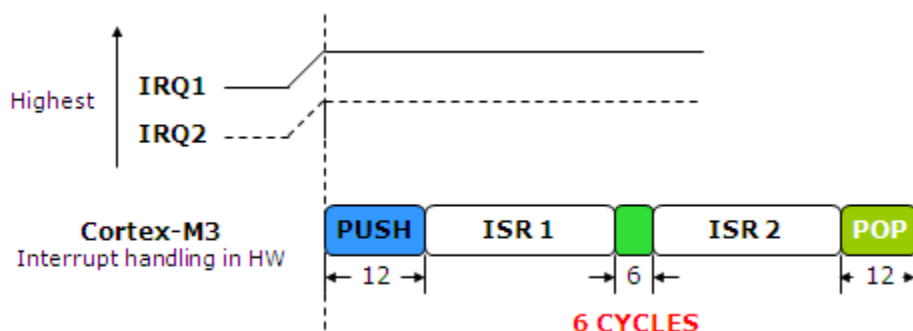
### 2.5.2.1. Prerušenie prerušenia (Interrupt Pre-emption)

Ak príde požiadavka na prerušenie s vyššou prioritou v čase obsluhy prerušenia s nižšou prioritou, obsluha prerušenia s nižšou prioritou sa zastaví. Vybrané registre sa uložia v 12 cykloch do zásobníka, načíta sa adresa obslužného podprogramu prerušenia s vyššou prioritou. Po ukončení obsluhy prerušenia s vyššou prioritou sa obnovia registre zo zásobníka a pokračuje obsluha prerušenia s nižšou prioritou.

## ARM – Cortex M

### 2.5.2.2. Metóda koncového zret'azenia (Tail Chaining)

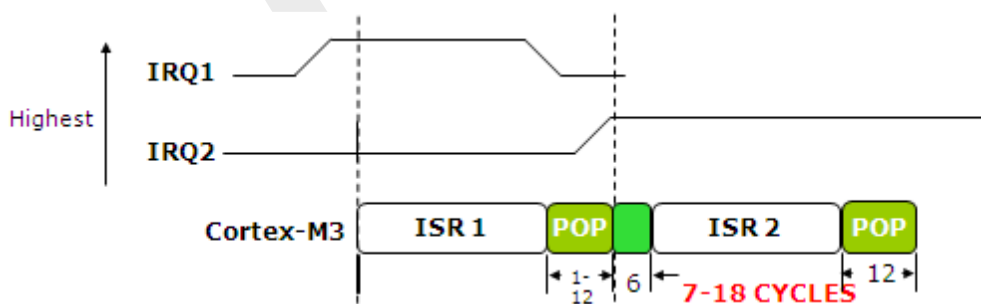
Ak beží obsluha prerušenia s vyššou prioritou a príde požiadavka na prerušenie s nižšou prioritou, použije sa metóda Tail Chaining, aby sa zabezpečilo minimálne oneskorenie medzi obsluhými podprogramami.



Obrázok 18: Tail Chaining

Ako vidieť z obrázka, počas 12 cyklov sa uložia registre do zásobníka, vyberie sa adresa obslužného podprogramu. Po ukončení obslužného podprogramu sa neobnovia registre zo zásobníka, ale načíta sa adresa obslužného podprogramu ďalšieho prerušenia podľa priority – 6 cyklov. Po ukončení obsluhy prerušenia sa obnovia registre zo zásobníka – 12 cyklov.

Ak počas obsluhy prerušenia príde požiadavka na prerušenie s nižšou prioritou, od obnovovania registrov bude upustené, ukazovateľ zásobníka sa obnoví na pôvodnú hodnotu, pridá sa 6 cyklov na vybratie adresy obslužného podprogramu prerušenia s nižšou prioritou t.j. oneskorenie je 7 – 18 cyklov pred obsluhou nového prerušenia. (viď. nasledujúci obrázok)

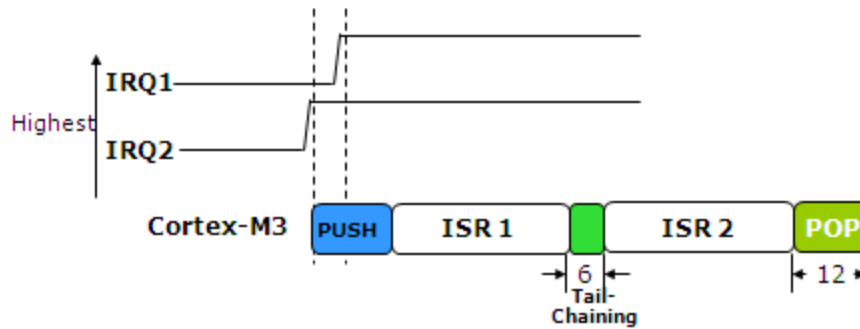


Obrázok 19: Ret'azenie obsluhy prerušení

### 2.5.2.3. Oneskorený príchod požiadavky s vyššou prioritou

V realite sa často stáva, že začne obsluha požiadavky na prerušenie s nižšou prioritou a príde následne požiadavka s vyššou prioritou. Pokiaľ sa tak stane v inicializačnom ukladaní registrov do zásobníka (PUSH), prepne sa na obsluhu prerušenia s vyššou prioritou, po jej ukončení nasleduje Tail Chaining.

## ARM – Cortex M



Obrázok 20: Oneskorený príchod požiadavky na prerušenie s vyššou prioritou

### 2.5.3. Konfigurácia a použitie NVIC

Aby sme mohli použiť NVIC, musíme:

1. Nakonfigurovať vektorovú tabuľku pre zdroje prerušenia, ktoré chceme používať.
2. Nakonfigurovať NVIC registre na povolenie a nastavenie priorít prerušení.
3. Nakonfigurovať periférie a povoliť ich podporu prerušení.

#### 2.5.3.1. Vektorová tabuľka výnimiek

V CORTEX-M sa vektorová tabuľka výnimiek (prerušení) nachádza v pamäti v dolnej časti pamäťového modelu, konkrétne začína na adrese 0x00000004. Prvé 4B sú určené pre inicializačnú hodnotu zásobníka - „stack pointer“.

Každý vektor prerušenia vo vektorovej tabuľke má 4B a obsahuje adresu začiatku obslužného podprogramu určeného na obsluhu príslušného prerušenia. Prvých 15 položiek tabuľky sú výnimky, ktoré môžu nastať v jadre procesora (resetovací vektor, NMI, poruchy a riadenie chýb, výnimky debuggera, prerušenie od SysTick Timer prerušenie, ...). Perifériám sú vyhradené prerušenia od 16 po max. počet určený výrobcom (označované ako Interrupt\_0 .. Interrupt\_240).

Tabuľku adries vektorov prerušenia je možné presunúť z pôvodnej preddefinovanej adresy (0x00000004) na ľubovoľnú inú adresu v rozsahu adresného priestoru procesora.

## ARM – Cortex M

No.	Exception Type	Priority	Type of Priority	Descriptions
1	Reset	-3 (Highest)	fixed	Reset
2	NMI	-2	fixed	Non-Maskable Interrupt
3	Hard Fault	-1	fixed	Default fault if other handler not implemented
4	MemManage Fault	0	settable	MPU violation or access to illegal locations
5	Bus Fault	1	settable	Fault if AHB interface receives error
6	Usage Fault	2	settable	Exceptions due to program errors
7-10	Reserved	N.A.	N.A.	
11	SVCall	3	settable	System Service call
12	Debug Monitor	4	settable	Break points, watch points, external debug
13	Reserved	N.A.	N.A.	
14	PendSV	5	settable	Pendable request for System Device
15	SYSTICK	6	settable	System Tick Timer
16	Interrupt #0	7	settable	External Interrupt #0
*****	*****	*****	settable	*****
256	Interrupt# 240	247	settable	External Interrupt #240

Obrázok 21: Tabuľka výnimiek – prerušenie



Registre NVIC sú sprístupnené len ak CPU pracuje v privilegovanom režime.

Interné CORTEX výnimky sú konfigurované v System Control a System Priority registroch.

Vonkajšie prerušenia užívateľských periférií sú konfigurované cez IRQ registre.

Obrázok 22: Registre NVIC

Ako vidieť z tabuľky Reset, NMI a HardFault majú fixne danú prioritu, ostatné vnútorné výnimky majú nastaviteľnú prioritu (v System Priority) – každá výnimka má 8 bitov na nastavenie priority (keďže má 16 úrovní priority = využívajú sa len 4 bity).