Perifera och externa avbrott

Ur innehållet:

- NVIC - Nested Vectored Interrupt Controller
- Avbrott från interna periferienheter, (”Timers”)
- EXTI - External Interrupt Configuration
- Avbrottsvippa med programstyrd återställning
- Pulsgenerator för periodiska avbrott

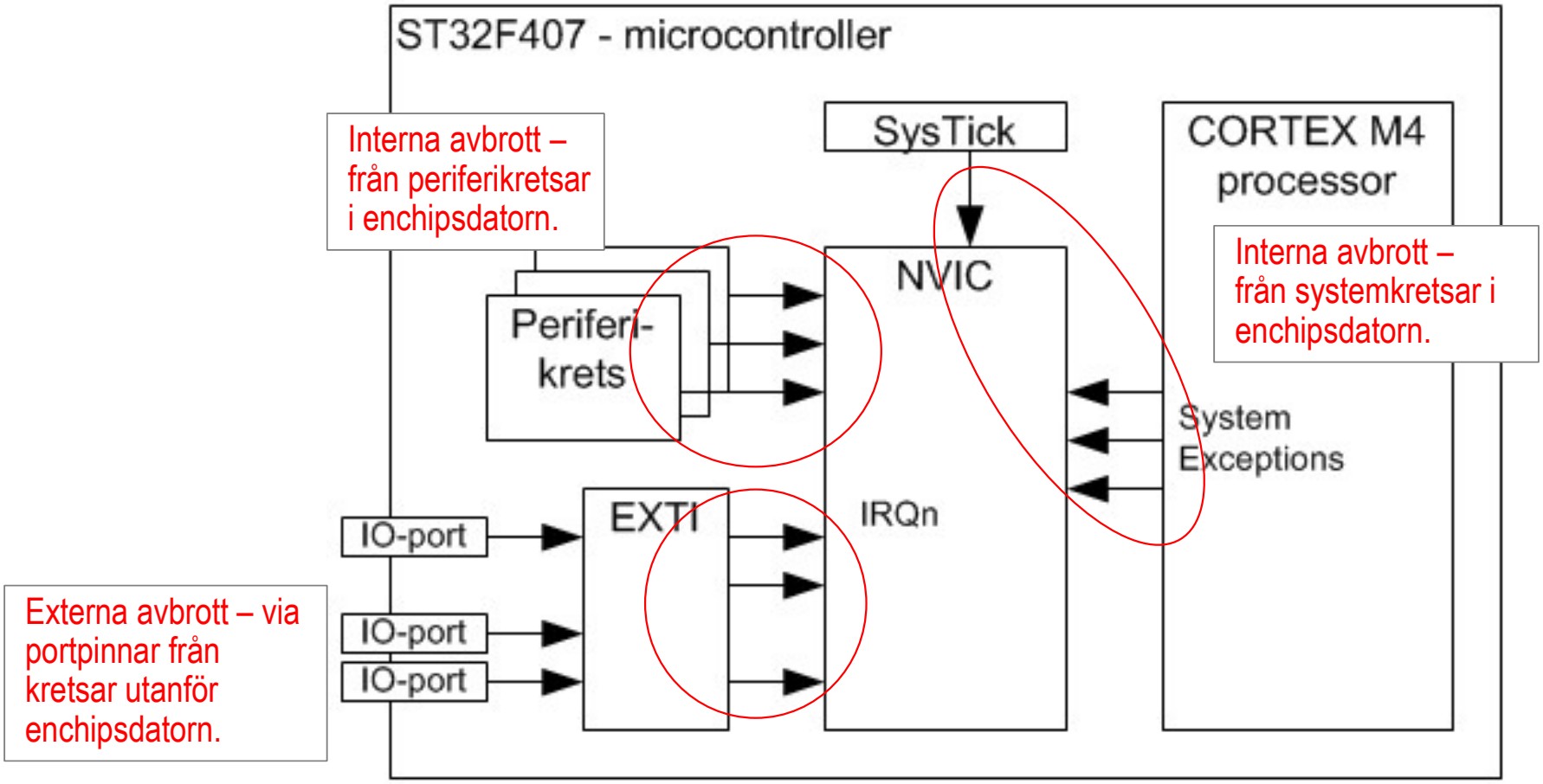
Läsanvisningar:

- Arbetsbok kap 6.3, 6.6, 6.7

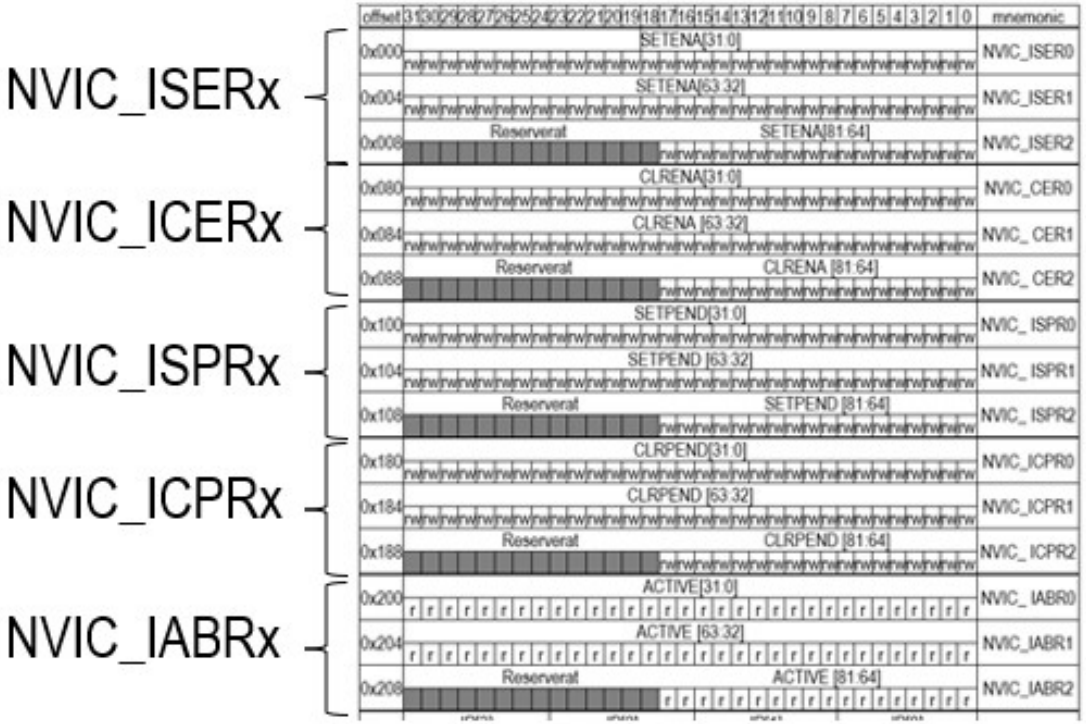
Målsättningar:

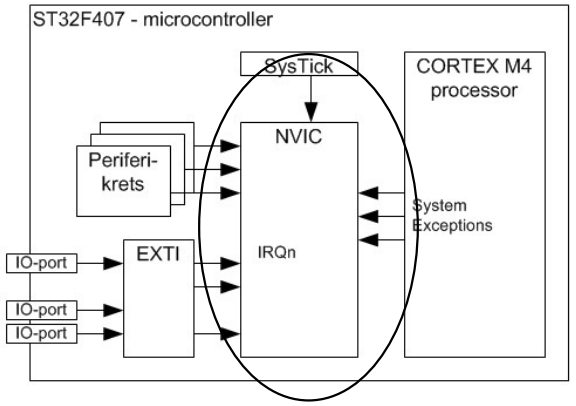
- Konfigurera NVIC för interna och externa periferikretsar
- Konfigurera EXTI för portpinne som en avbrottsingång (extern periferikrets)
- Programmera extern avbrottsvippa, och hantera vektoriserade avbrott

Tre grupper av avbrottskällor



NVIC – Kontrollerar alla avbrott





Fem uppsättningar register för olika funktioner, men med samma struktur. Vår processor har 82 avbrottsgenererande enheter, alltså har varje registeruppsättning 82 bitar.

- Interrupt Set Enable Registers, NVIC_ISERx
- Interrupt Clear Enable Registers, NVIC_ICERx
- Interrupt Set Pendig Registers, NVIC_ISEPRx
- Interrupt Clear Pending Registers, NVIC_ICPRx
- Interrupt Active Bit Registers, NVIC_IABRx

Bestäm register **x** och bitnummer **bit** i NVIC

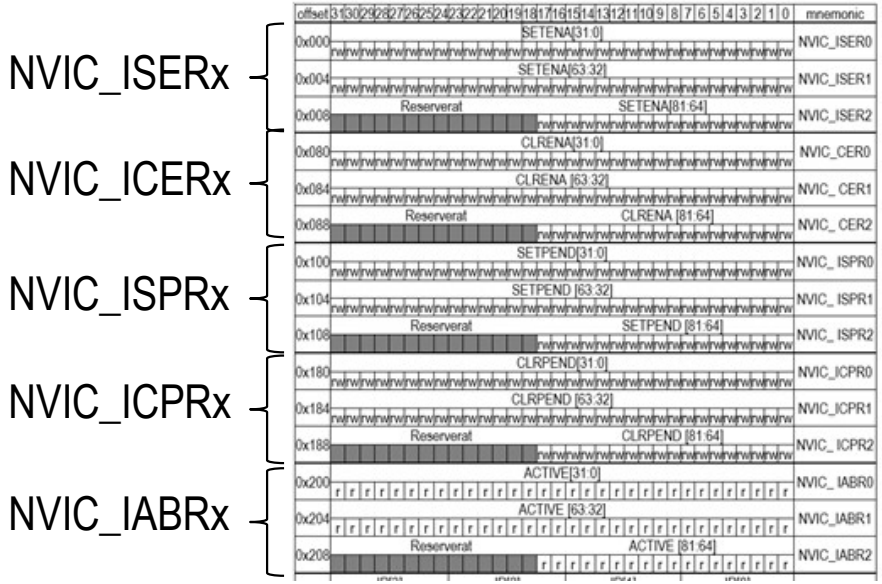
För vektor IRQ=n (n = 0..81):

`x` = n/32

dvs. `x` kan vara 0,1,2

`bit` = n – (32*`x`),

dvs. bit kan vara 0..31



offset	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0	mnemonic
0x000	SETENA[31:0]	NVIC_ISER0
0x004	SETENA[63:32]	NVIC_ISER1
0x008	Reserverat / SETENA[81:64]	NVIC_ISER2

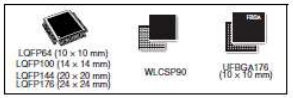
Interna avbrott från periferikretsar

life.augmented

STM32F405xx
STM32F407xx

ARM Cortex-M4 32b MCU+FPU, 210DMIPS, up to 1MB Flash/192+4KB RAM, USB OTG HS/FS, Ethernet, 17 TIMs, 3 ADCs, 15 comm. interfaces & camera

Datasheet - production data

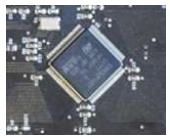


Features

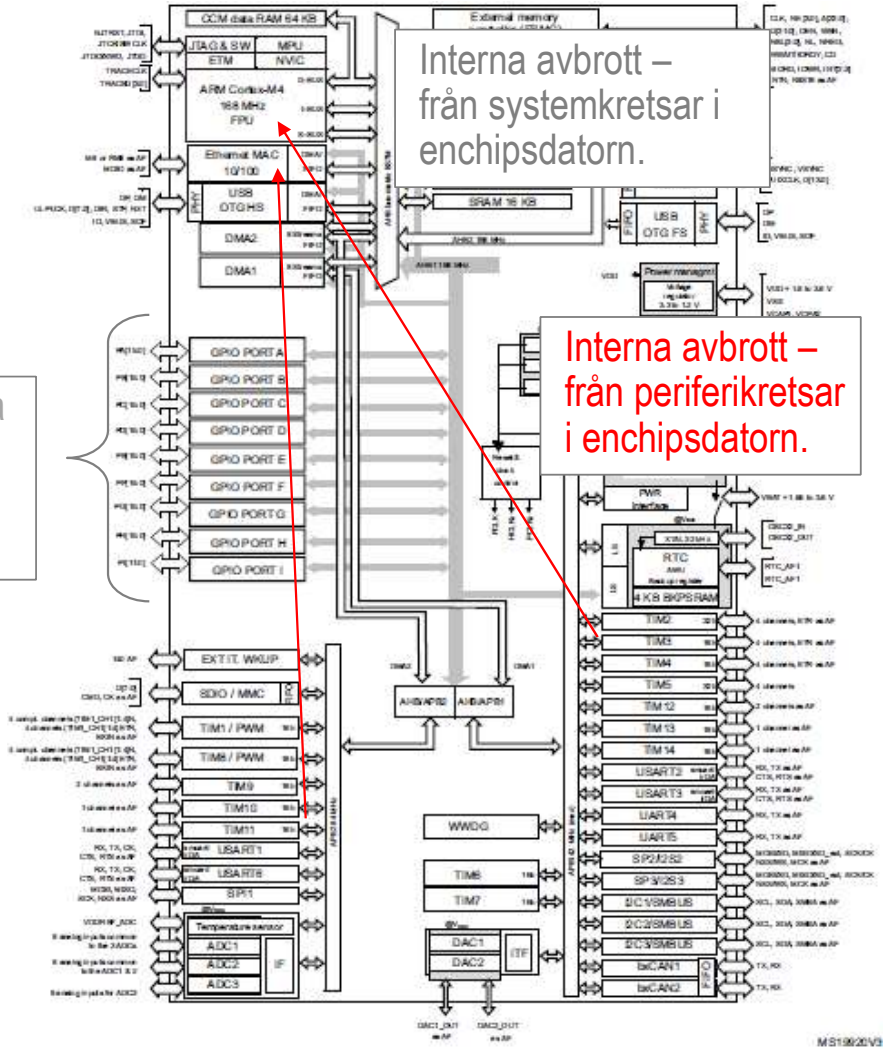
- Core: ARM 32-bit Cortex™-M4 CPU with FPU, Adaptive real-time accelerator (ART Accelerator™) allowing 0-wait state execution from Flash memory, frequency up to 168 MHz, memory protection unit, 210 DMIPS/ 1.25 DMIPS/MHz (Dhrystone 2.1), and DSP instructions
- Memories
  - Up to 1 Mbyte of Flash memory
  - Up to 192+4 Kbytes of SRAM including 64-Kbyte of CCM (core coupled memory) data RAM
  - Flexible static memory controller supporting Compact Flash, SRAM, PSRAM, NOR and NAND memories
- LCD parallel interface, 8080/6800 modes
- Clock, reset and supply management
  - 1.8 V to 3.6 V application supply and I/Os
  - POR, PDR, PVD and BOR
  - 4-to-26 MHz crystal oscillator
  - Internal 16 MHz factory-trimmed RC (1% accuracy)
  - 32 kHz oscillator for RTC with calibration
  - Internal 32 kHz RC with calibration
- Low power
  - Sleep, Stop and Standby modes
  - V_{BAT} supply for RTC, 20×32 bit backup registers + optional 4 KB backup SRAM
- 3×12-bit, 2.4 MSPS A/D converters: up to 24 channels and 7.2 MSPS in triple interleaved mode
- 2×12-bit D/A converters
- General-purpose DMA: 16-stream DMA controller with FIFOs and burst support
- Up to 17 timers: up to twelve 16-bit and two 32-bit timers up to 168 MHz, each with up to 4 IC/OC/PWM or pulse counter and quadrature (incremental) encoder input
- Debug mode
  - Serial wire debug (SWD) & JTAG interfaces
  - Cortex-M4 Embedded Trace Macrocell™
- Up to 140 I/O ports with interrupt capability
  - Up to 136 fast I/Os up to 84 MHz
  - Up to 138 5 V-tolerant I/Os
- Up to 15 communication interfaces
  - Up to 3 × I^2C interfaces (SMBus/PMBus)
  - Up to 4 USARTs/2 UARTs (10.5 Mbit/s, ISO 7816 interface, LIN, IrDA, modem control)
  - Up to 3 SPIs (42 Mbits/s), 2 with muxed full-duplex I^2S to achieve audio class accuracy via internal audio PLL or external clock
  - 2 × CAN interfaces (2.0B Active)
  - SDIO interface
- Advanced connectivity
  - USB 2.0 full-speed device/host/OTG controller with on-chip PHY
  - USB 2.0 high-speed/full-speed device/host/OTG controller with dedicated DMA, on-chip full-speed PHY and ULPI
  - 10/100 Ethernet MAC with dedicated DMA: supports IEEE 1588v2 hardware, MII/RMII
- 8- to 14-bit parallel camera interface up to 54 Mbytes/s
- True random number generator
- CRC calculation unit
- 96-bit unique ID
- RTC: subsecond accuracy, hardware calendar

Table 1. Device summary

Reference	Part number
STM32F405xx	STM32F405RG, STM32F405VG, STM32F405ZG, STM32F405OG, STM32F405OE
STM32F407xx	STM32F407VG, STM32F407IG, STM32F407ZG, STM32F407VE, STM32F407ZE, STM32F407IE



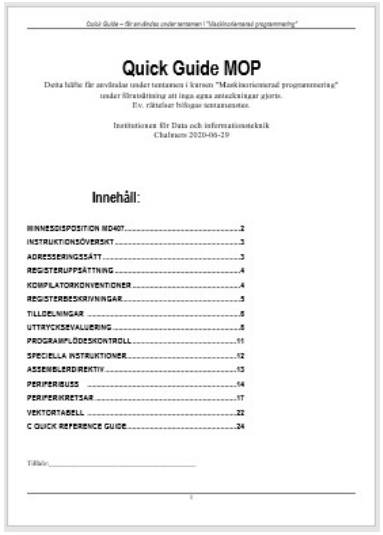
Externa avbrott – via portpinnar från kretsar utanför enchipsdatorn.



Exempel: Avbrottsvektor och bitnummer i NVIC för Timer 6

Vi letar upp TIM6 i vektortabellen. Den första kolumnen anger avbrottets nummer i NVIC, dvs. 54. Kolumnen längst till höger anger offset för avbrottsvektorns placering i vektortabellen:

52	settable	UART4	UART4 global interrupt	0x0000 0110
53	settable	UART5	UART5 global interrupt	0x0000 0114
54	settable	TIM6_DAC	TIM6 global interrupt, DAC1 and DAC2 underrun error interrupts	0x0000 0118
55	settable	TIM7	TIM7 global interrupt	0x0000 011C
56	settable	DMA2_Stream0	DMA2_Stream0 global interrupt	0x0000 0120



Quick Guide MOP

Detta hjälp för användas under tentamen i kursen "Maskinorienterad programmering" under förutsättning att inga egna anteckningar gjorts. Ev. tillåtet bilagor tentamenstes.

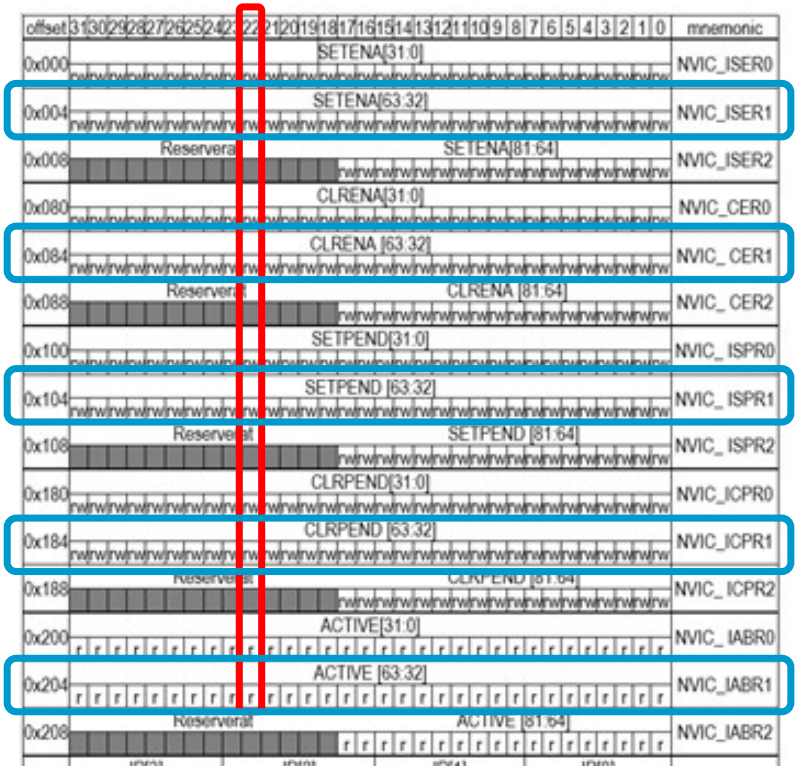
Institutionen för Data och informationsteknik
Chalmers 2020-08-29

Innehåll:

Tillbör: _____

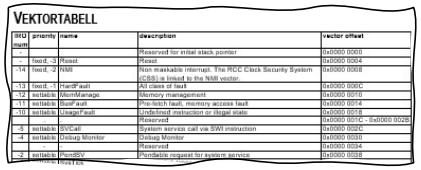
Vi har alltså avbrottsnummer 54 och får:
x = 54/32 = 1, dvs. register NVIC_ xxxx**1**

Bit = 54 -32 = 22
Dvs: bit 22 i registren NVIC_ xxxx**1**



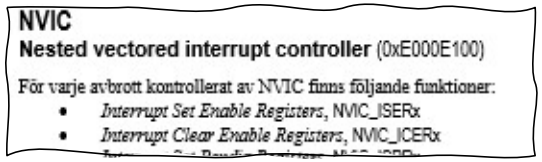
Exempel:

Vi samanfattar i makrodefinitioner för avbrottsvektor, registerdefinitioner och bitnummer:



VEKTORTABELL

IRQ num	priority	name	description	vector offset
-			Reserved for initial stack pointer	0x0000 0000
-	fixed, -3	Reset	Reset	0x0000 0004
-14	fixed, -2	NMI	Non maskable interrupt. The RCC Clock Security System (CSS) is linked to the NMI vector.	0x0000 0008
-13	fixed, -1	HardFault	All class of fault	0x0000 000C
-12	settable	MemManage	Memory management	0x0000 0010
-11	settable	BusFault	Pre-fetch fault, memory access fault	0x0000 0014
-10	settable	UsageFault	Undefined instruction or illegal state	0x0000 0018
			Reserved	0x0000 001C - 0x0000 0028
-5	settable	SVCall	System service call via SWI instruction	0x0000 002C
-4	settable	Debug Monitor	Debug Monitor	0x0000 0030
			Reserved	0x0000 0034
-2	settable	PendSV	Pendable request for system service	0x0000 0038



NVIC
Nested vectored interrupt controller (0xE000E100)

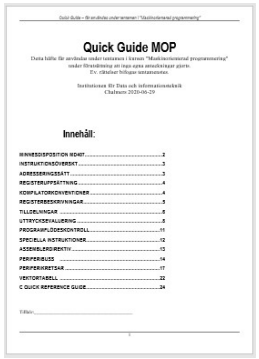
För varje avbrott kontrollerat av NVIC finns följande funktioner:

- *Interrupt Set Enable Registers*, NVIC_ISERx
- *Interrupt Clear Enable Registers*, NVIC_ICERx

Relokerad avbrottsvektor: `0x2001C000 + 0x118 = 0x2001C118`

Bitposition:

```
#define NVIC_TIM6_IRQ_BPOS  (1<<22)
```



Quick Guide MOP

Innehåll:

NVIC_ISER: `0xE000E100 + x*4`

```
#define NVIC_TIM6_ISER ((volatile unsigned int *) 0xE000E104)
```

NVIC_ICER: `0xE000E180 + x*4`

```
#define NVIC_TIM6_ICER ((volatile unsigned int *) 0xE000E184)
```

NVIC_ISPR: `0xE000E200 + x*4`

```
#define NVIC_TIM6_ISPR ((volatile unsigned int *) 0xE000E204)
```

NVIC_ICPR: `0xE000E280 + x*4`

```
#define NVIC_TIM6_ICPR ((volatile unsigned int *) 0xE000E284)
```

NVIC_IABR: `0xE000E300 + x*4`

```
#define NVIC_TIM6_IABR ((volatile unsigned int *) 0xE000E304)
```

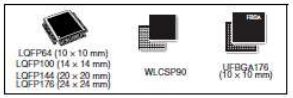
Externa avbrott från periferikretsar

life.augmented

STM32F405xx
STM32F407xx

ARM Cortex-M4 32b MCU+FPU, 210DMIPS, up to 1MB Flash/192+4KB RAM, USB OTG HS/FS, Ethernet, 17 TIMs, 3 ADCs, 15 comm. interfaces & camera

Datasheet - production data

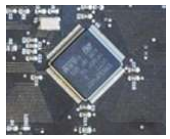


Features

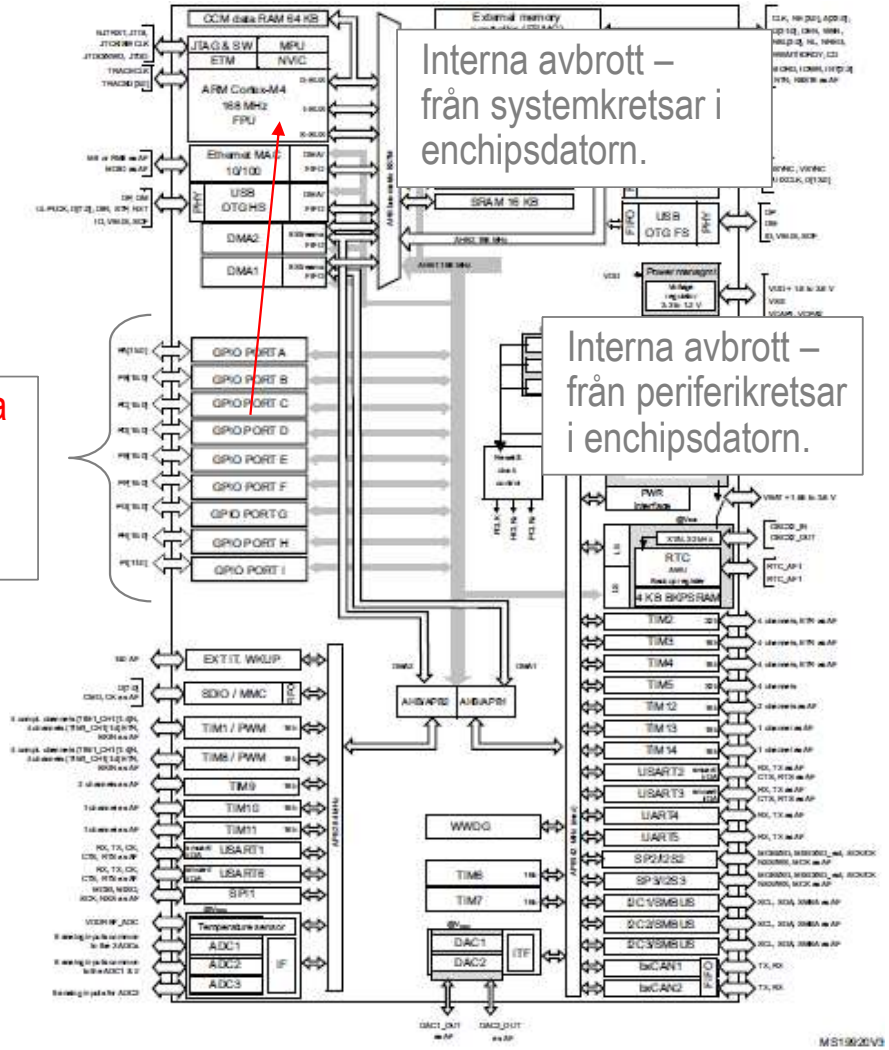
- Core: ARM 32-bit Cortex™-M4 CPU with FPU, Adaptive real-time accelerator (ART Accelerator™) allowing 0-wait state execution from Flash memory, frequency up to 168 MHz, memory protection unit, 210 DMIPS/ 1.25 DMIPS/MHz (Dhrystone 2.1), and DSP instructions
- Memories
  - Up to 1 Mbyte of Flash memory
  - Up to 192+4 Kbytes of SRAM including 64-Kbyte of CCM (core coupled memory) data RAM
  - Flexible static memory controller supporting Compact Flash, SRAM, PSRAM, NOR and NAND memories
- LCD parallel interface, 8080/6800 modes
- Clock, reset and supply management
  - 1.8 V to 3.6 V application supply and I/Os
  - POR, PDR, PVD and BOR
  - 4-to-26 MHz crystal oscillator
  - Internal 16 MHz factory-trimmed RC (1% accuracy)
  - 32 kHz oscillator for RTC with calibration
  - Internal 32 kHz RC with calibration
- Low power
  - Sleep, Stop and Standby modes
  - V_{BAT} supply for RTC, 20×32 bit backup registers + optional 4 KB backup SRAM
- 3×12-bit, 2.4 MSPS A/D converters: up to 24 channels and 7.2 MSPS in triple interleaved mode
- 2×12-bit D/A converters
- General-purpose DMA: 16-stream DMA controller with FIFOs and burst support
- Up to 17 timers: up to twelve 16-bit and two 32-bit timers up to 168 MHz, each with up to 4 IC/OC/PWM or pulse counter and quadrature (incremental) encoder input
- Debug mode
  - Serial wire debug (SWD) & JTAG interfaces
  - Cortex-M4 Embedded Trace Macrocell™
- Up to 140 I/O ports with interrupt capability
  - Up to 136 fast I/Os up to 84 MHz
  - Up to 138 5 V-tolerant I/Os
- Up to 15 communication interfaces
  - Up to 3 × I^2C interfaces (SMBus/PMBus)
  - Up to 4 USARTs/2 UARTs (10.5 Mbit/s, ISO 7816 interface, LIN, IrDA, modem control)
  - Up to 3 SPIs (42 Mbits/s), 2 with muxed full-duplex I^2S to achieve audio class accuracy via internal audio PLL or external clock
  - 2 × CAN interfaces (2.0B Active)
  - SDIO interface
- Advanced connectivity
  - USB 2.0 full-speed device/host/OTG controller with on-chip PHY
  - USB 2.0 high-speed/full-speed device/host/OTG controller with dedicated DMA, on-chip full-speed PHY and ULPI
  - 10/100 Ethernet MAC with dedicated DMA: supports IEEE 1588v2 hardware, MII/RMII
- 8- to 14-bit parallel camera interface up to 54 Mbytes/s
- True random number generator
- CRC calculation unit
- 96-bit unique ID
- RTC: subsecond accuracy, hardware calendar

Table 1. Device summary

Reference	Part number
STM32F405xx	STM32F405RG, STM32F405VG, STM32F405ZG, STM32F405OG, STM32F405OE
STM32F407xx	STM32F407VG, STM32F407IG, STM32F407ZG, STM32F407VE, STM32F407ZE, STM32F407IE



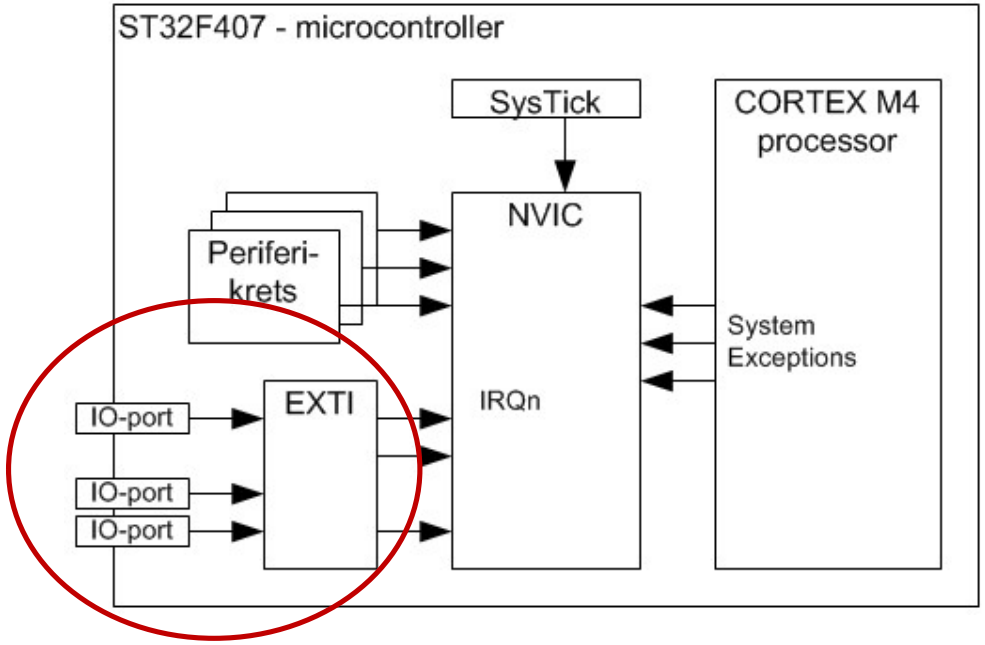
Externa avbrott – via portpinnar från kretsar utanför enchipsdatorn.



Externa avbrott

En portpinne kan konfigureras som avbrottsingång via EXTI-modulen (External interrupt/event controller)

Detta möjliggör avbrott från enheter utanför enchipsdatorn, *externa avbrott*

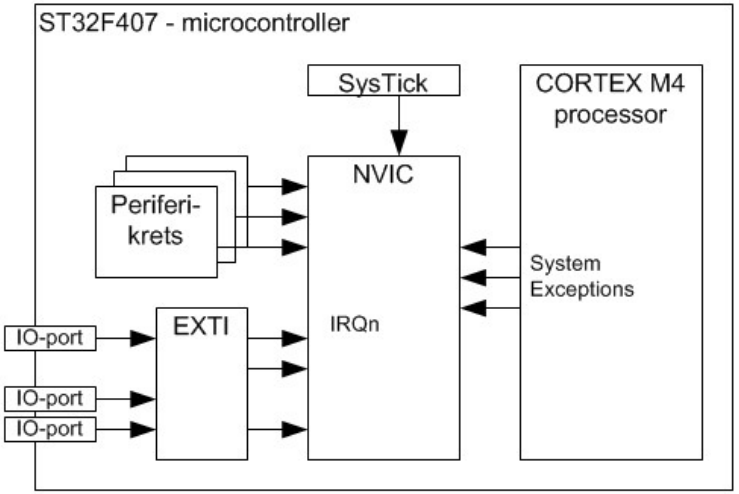


	6	settable	SysTick	System tick timer
0	7	settable	WWDG	Window Watchdog interrupt
1	8	settable	PVD	PVD through EXTI line detection interrupt
2	9	settable	TAMP_STAMP	Tamper and TimeStamp interrupts through the EXTI line
3	10	settable	RTC_WKUP	RTC Wakeup interrupt through the EXTI line
4	11	settable	FLASH	Flash global interrupt
5	12	settable	RCC	RCC global interrupt
6	13	settable	EXTI0	EXTI Line0 interrupt
7	14	settable	EXTI1	EXTI Line1 interrupt
8	15	settable	EXTI2	EXTI Line2 interrupt
9	16	settable	EXTI3	EXTI Line3 interrupt
10	17	settable	EXTI4	EXTI Line4 interrupt
11	18	settable	DMA1_Stream0	DMA1 Stream0 global interrupt
12	19	settable	DMA1_Stream1	DMA1 Stream1 global interrupt

Vektorer för externa avbrottskällor

Avbrottsingångar konfigureras i tre steg

- SYSCFG-modulen används för att koppla en GPIO-pinne från någon port till en specifik avbrottslina (1 av 16) .
- EXTI-modulen programmeras.
- NVIC-modulen (Nested vectored interrupt controller) programmeras för att hantera avbrott från denna avbrottsvektor (NVIC_ISERx).



Irq	Hanteringsrutin	Beskrivning
EXTI0_IRQn	EXTI0_IRQHandler	för IO-pinnar anslutna till avbrottslina 0
EXTI1_IRQn	EXTI1_IRQHandler	för IO-pinnar anslutna till avbrottslina 1
EXTI2_IRQn	EXTI2_IRQHandler	för IO-pinnar anslutna till avbrottslina 2
EXTI3_IRQn	EXTI3_IRQHandler	för IO-pinnar anslutna till avbrottslina 3
EXTI4_IRQn	EXTI4_IRQHandler	för IO-pinnar anslutna till avbrottslina 4
EXTI9_5_IRQn	EXTI9_5_IRQHandler	för IO-pinnar anslutna till avbrottslinor 5 till 9
EXTI15_10_IRQn	EXTI15_10_IRQHandler	för IO-pinnar anslutna till avbrottslinor 10 till 15

SYSCFG EXTICRx används för att koppla IO-pinnar till processorns avbrottssystem

offset	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Register
0																																	SYSCFG_MEMRMP
4																																	SYSCFG_PMC
8																																	SYSCFG_EXTICR1
0xC																																	SYSCFG_ EXTICR2
0x10																																	SYSCFG_ EXTICR3
0x14																																	SYSCFG_ EXTICR4
0x20																																	SYSCFG_CMPCR

EXTICRx är fyra kontrollregister med samma struktur:

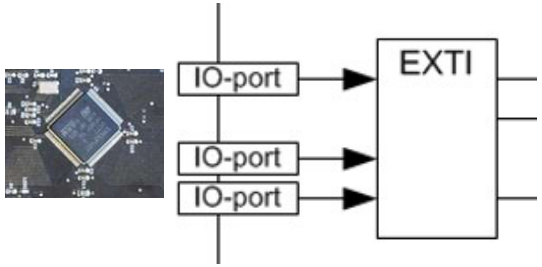
offset	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Register
8	EXTI3[3:0]				EXTI2[3:0]				EXTI1[3:0]				EXTI0[3:0]				SYSCFG_EXTICR1
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	

Bitarna i EXTICRx bestämmer hur en IO-pinne dirigeras till någon av de 16 avbrottslinorna EXTI0..EXTI15 genom att motsvarande fält skrivs med fyra bitar.
Undantag: PK[15:8] används ej.

0000: PA[x]	0001: PB[x]	0010: PC[x]	0011: PD[x]	0100: PE[x]	0101: PF[x]
0110: PG[x]	0111: PH[x]	1000: PI[x]	1001: PJ[x]	1010: PK[x]	

Exempel:

Koppla pinne PC0 till “Extern avbrott 0”



5	12	settable	RCC	RCC global interrupt	0x0000 0054
6	13	settable	EXTI0	EXTI Line0 interrupt	0x0000 0058
7	14	settable	EXTI1	EXTI Line1 interrupt	0x0000 005C

offset	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Register
8	EXTI3[3:0]				EXTI2[3:0]				EXTI1[3:0]				EXTI0[3:0]				SYSCFG_EXTICR1
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	

0000: PA[x]	0001: PB[x]	0010: PC[x]	0011: PD[x]	0100: PE[x]	0101: PF[x]
0110: PG[x]	0111: PH[x]	1000: PI[x]	1001: PJ[x]	1010: PK[x]	

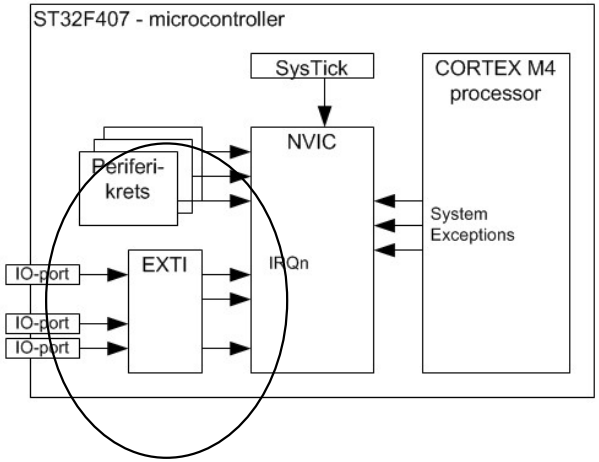
```
*SYSCFG_EXTICR1 &= 0xFFF0;      /* Nollställ bitfältet */
*SYSCFG_EXTICR1 |= 0x0002;      /* 0010 -> EXTI0[3:0] */
```

EXTI-modul

Totalt finns det 23 olika möjliga avbrottslinor i EXTI-modulen, i STM32F407 används de 16 första för externa avbrott. Några av dessa har gemensam avbrottsvektor.

Irq	Hanteringsrutin	Beskrivning
EXTI0_IRQn	EXTI0_IRQHandler	för IO-pinnar anslutna till avbrottslina 0
EXTI1_IRQn	EXTI1_IRQHandler	för IO-pinnar anslutna till avbrottslina 1
EXTI2_IRQn	EXTI2_IRQHandler	för IO-pinnar anslutna till avbrottslina 2
EXTI3_IRQn	EXTI3_IRQHandler	för IO-pinnar anslutna till avbrottslina 3
EXTI4_IRQn	EXTI4_IRQHandler	för IO-pinnar anslutna till avbrottslina 4
EXTI9_5_IRQn	EXTI9_5_IRQHandler	för IO-pinnar anslutna till avbrottslinor 5 till 9
EXTI15_10_IRQn	EXTI15_10_IRQHandler	för IO-pinnar anslutna till avbrottslinor 10 till 15

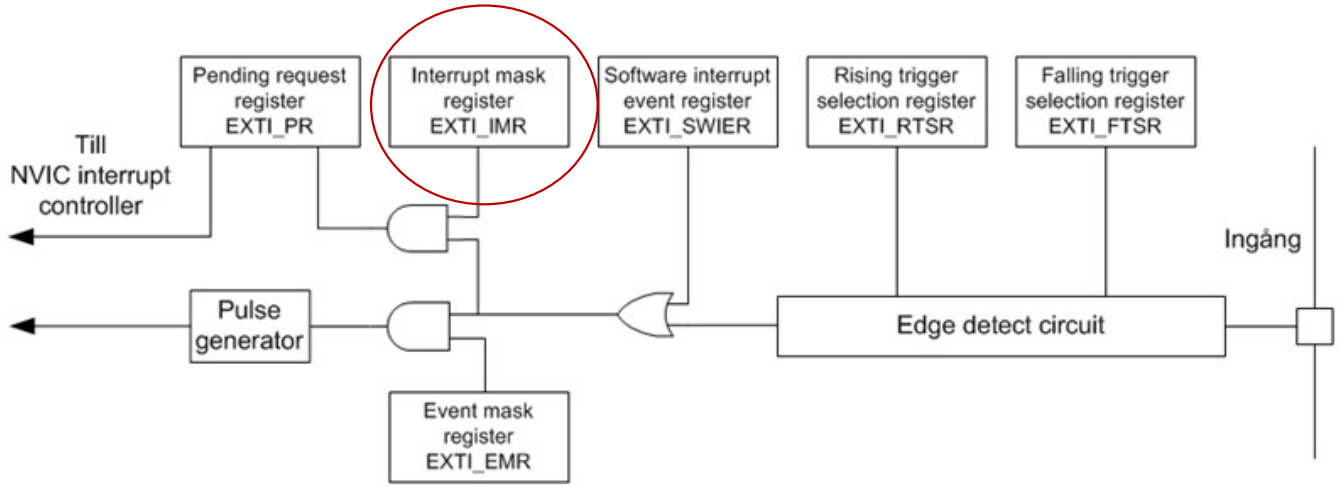
Irq	Beskrivning
EXTI16	PVD utgång fast ansluten till EXTIO16
EXTI17	RTC "Alarm event" fast ansluten till EXTIO17
EXTI18	USB OTG FS "Wakeup event" fast ansluten till EXTIO18
EXTI19	Ethernet "Wakeup event" fast ansluten till EXTIO19
EXTI20	USB OTG HS "Wakeup event" fast ansluten till EXTIO20
EXTI21	RTC "Tamper and TimeStamp" fast ansluten till EXTIO21
EXTI22	RTC "Wakeup event" fast ansluten till EXTIO22



offset	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Register
0																																	EXTI_IMR
4																																	EXTI_EMR
8																																	EXTI_RTSR
0xC																																	EXTI_FTSR
0x10																																	EXTI_SWIER
0x14																																	EXTI_PR

Port pinnar Px 15 .. Px 0 (x=A,B,C,D,E,F)

EXTI– registren har samma struktur



EXTI_IMR (0x40013C00) Interrupt Mask Register

offset	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	mnemonic
0										rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	EXTI_IMR

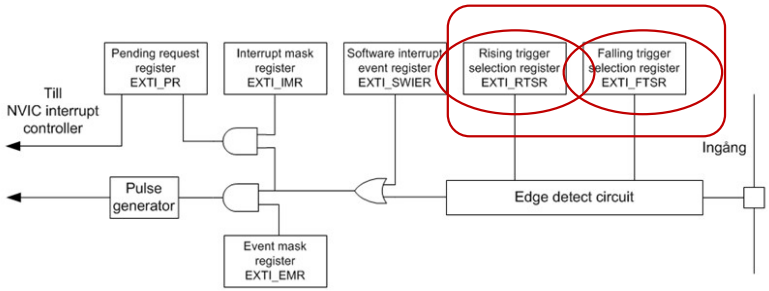
Kontrollerar EXTI 15

Kontrollerar EXTI 0

Exempel: Tillåt avbrott från EXTI0

```
*EXTI_IMR |= 1;
```

EXTI– triggfunktioner



EXTI_RTSR (0x40013C08) Rising Trigger Selection Register

offset	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	mnemonic
8										rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	EXTI_RTSR

Exempel: Generera avbrott då insignalen från den pinne som "routats" till EXTI6 går från 0 till1, dvs "positiv flank":

```
*EXTI_RTSR |= 1<<6;
```

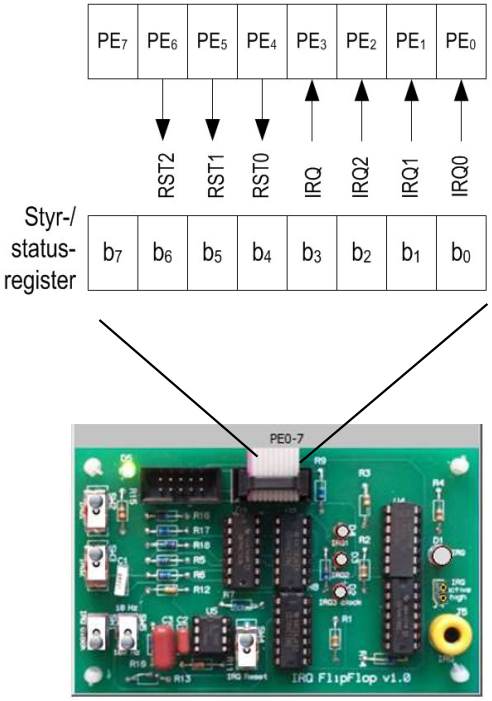
EXTI_FTSR (0x40013C0C) Falling Trigger Selection Register

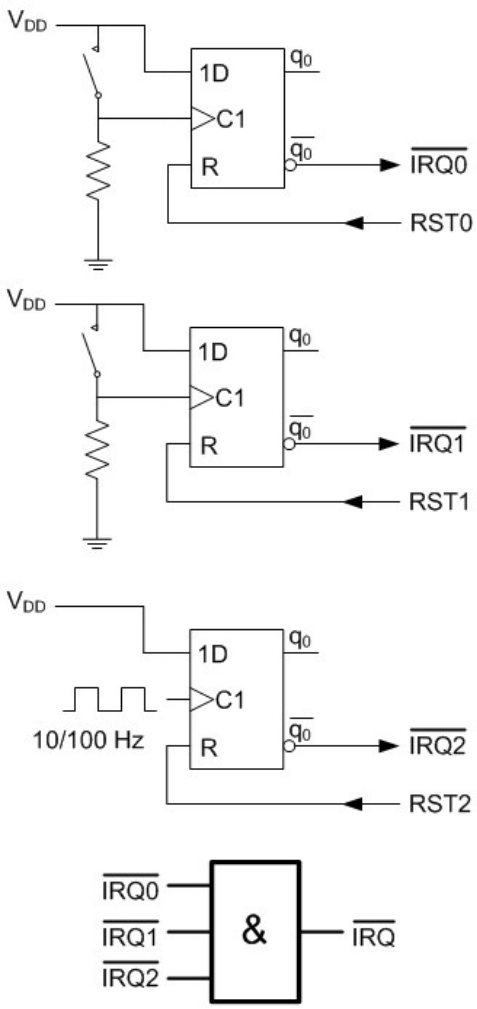
offset	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	mnemonic
0xC										rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	EXTI_FTSR

Exempel: Generera avbrott då insignalen från den pinne som "routats" till EXTI2 går från 1 till 0, dvs "negativ flank":

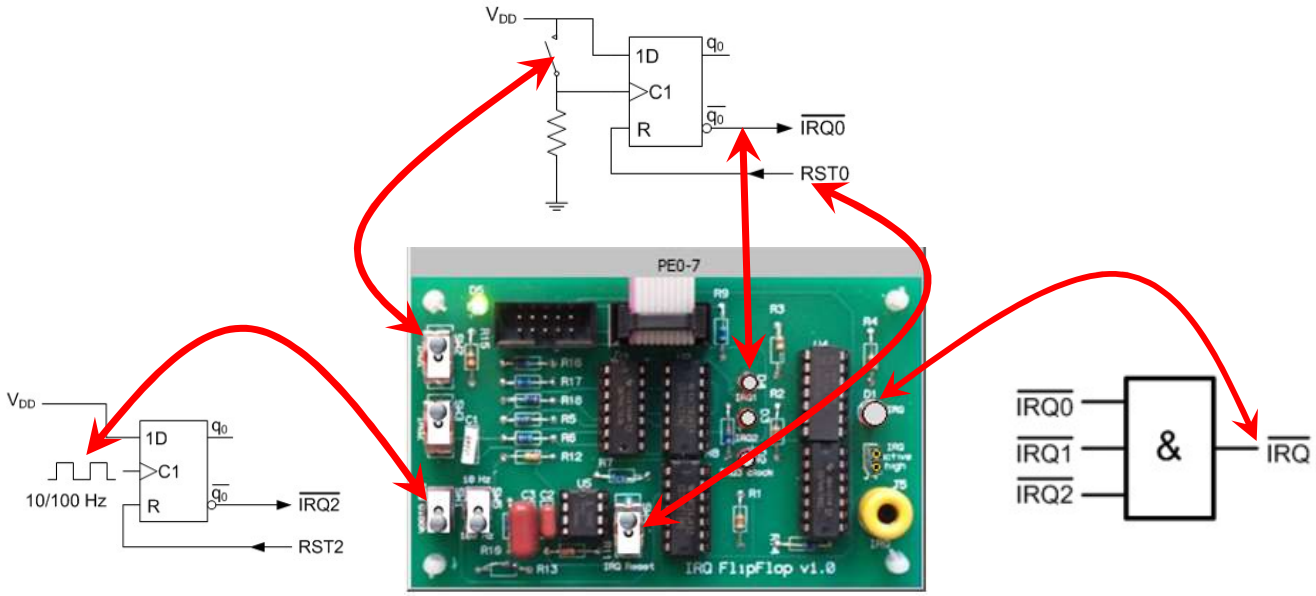
```
*EXTI_FTSR |= 1<<2;
```

IRQ-FlipFlop - avbrottsvippor

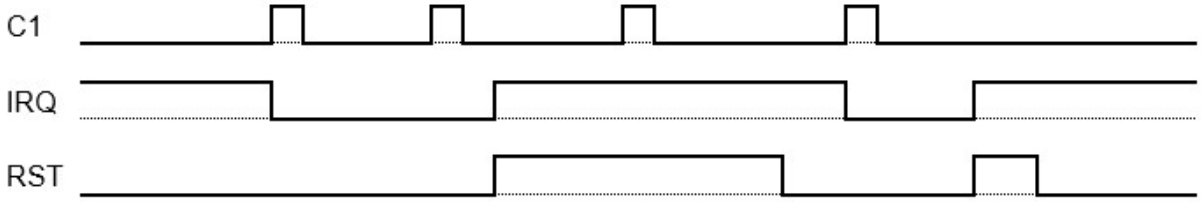




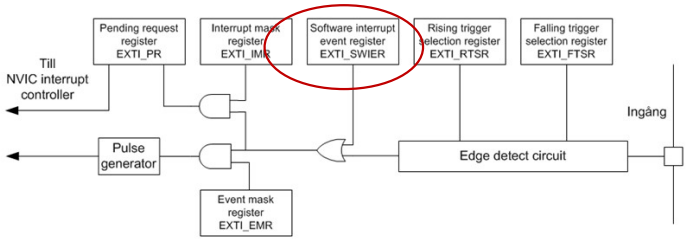
IRQ-FlipFlop - funktion



Exempel: Aktivering och återställning av IRQ-signal (aktiv låg)



EXTI– ”software interrupt”



Avbrott kan också genereras av programvara.

Om avbrott är tillåtet för lina *x*, kan programvara ge detta avvaktande tillstånd genom att skriva '1' till motsvarande bit i EXTI_SWIER.

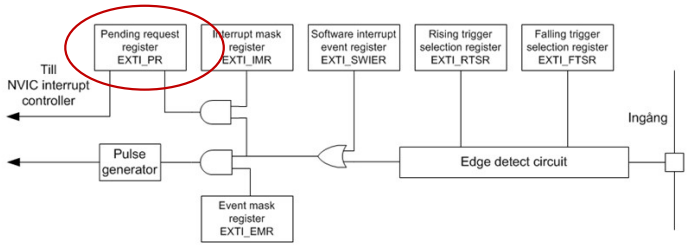
EXTI_SWIER (0x40013C10) Software Interrupt Event Register

offset	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	mnemonic
0x10										rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	EXTI_SWIER

Exempel: Generera avbrott hos EXTI3

```
*EXTI_SWIER |= 1<<3;
```

EXTI– ”interrupt pending”



Om ett avbrott avvaktar är motsvarande bit 1 i EXTI_PR.

Men registret är också skrivbart och biten återställs till 0 genom att 1 skrivs till bitpositionen.

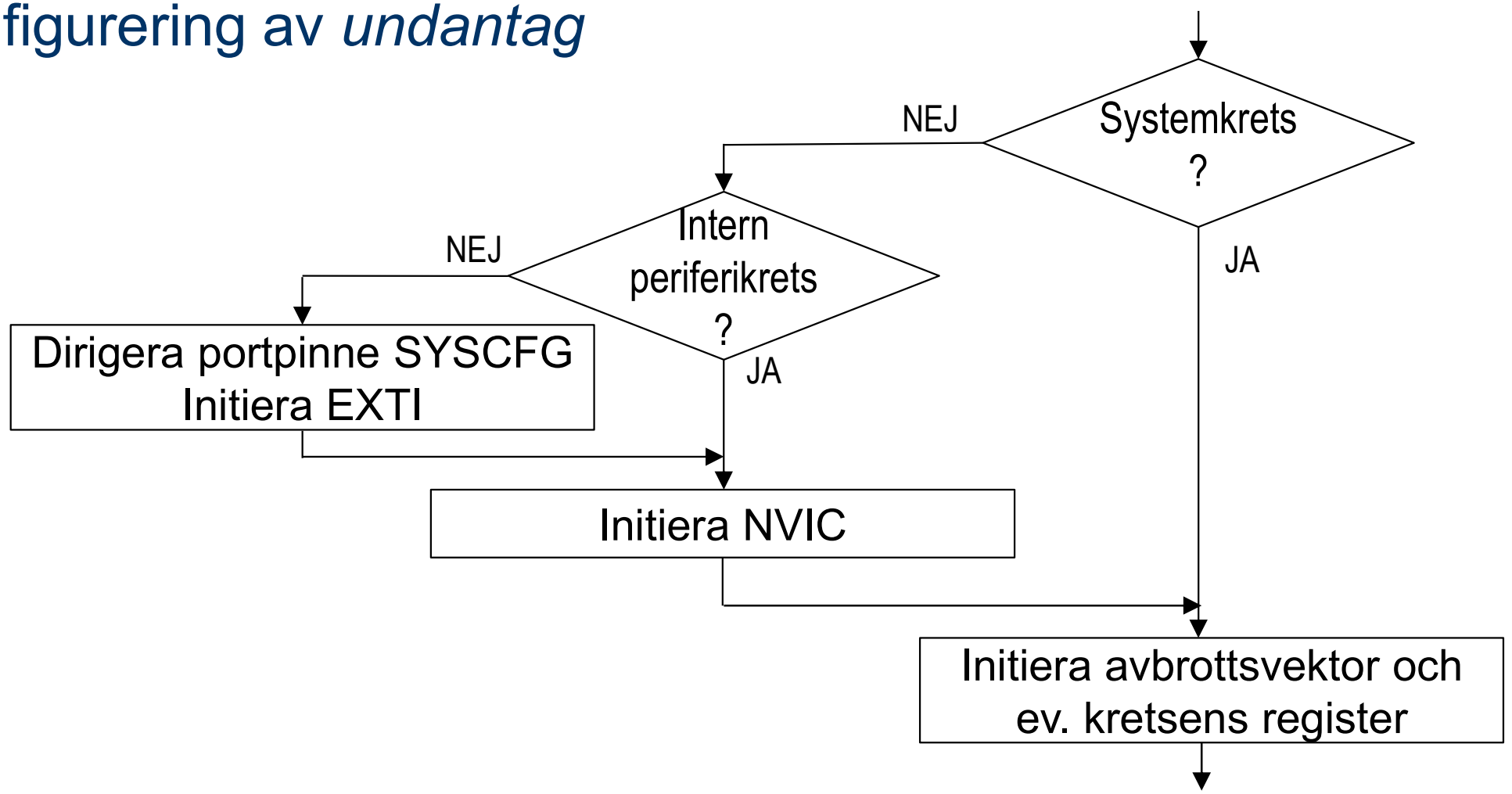
EXTI_PR (0x40013C14) Pending Register

offset	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	mnemonic
0x14										rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	EXTI_PR

Exempel: Om ett avbrott avvaktar för EXTI15, "kvittera" (återställ) då detta.

```
if( *EXTI_PR & (1<<15) )
   *EXTI_PR |= 1<<15;
```

Konfigurering av *undantag*



Pseudoparallell exekvering - en enkel ”TASK-switch”

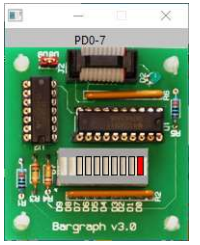
En enkel applikation med fyra olika program, (*processerna*) p0, p1, p2 och p3

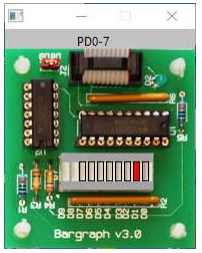
Växling mellan programmen utförs av ett överordnat program SV(*Supervisor*) vid ett avbrott från EXTI0 (bit 0).

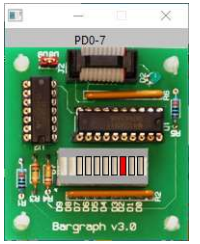
”Processerna”...

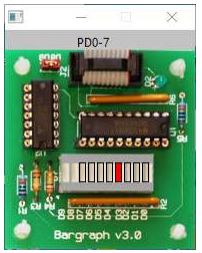
```
void p0( void )
{
  while(1)
    *GPIO_ODR_LOW = 1;
}
void p1( void )
{
  while(1)
    *GPIO_ODR_LOW = 2;
}
void p2( void )
{
  while(1)
    *GPIO_ODR_LOW = 4;
}
void p3( void )
{
  while(1)
    *GPIO_ODR_LOW = 8;
}
```

Processerna tänder var sin diod i diodrampen, vi kan då tydligt se vilken process som körs...

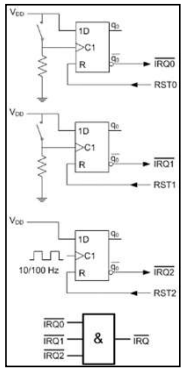






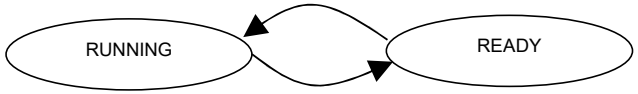


Processtillstånd



Avbrotten kan också triggas manuellt eller periodiskt från FLIP-FLOP-kortet...

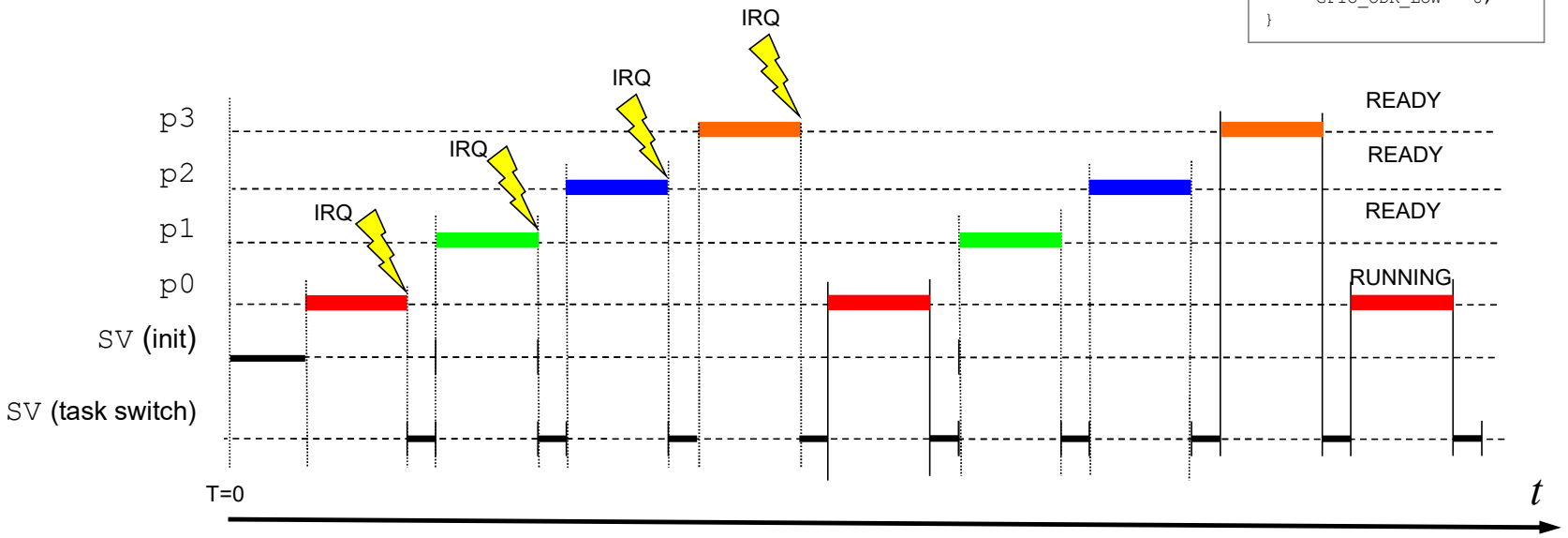
Vi har bara en CPU, som vi växlar mellan processerna. Processen kan därför befinna sig i olika *tillstånd.*



"Processerna"...

```
void    p0( void )
{
  while(1)
    *GPIO_ODR_LOW = 1;
}
void    p1( void )
{
  while(1)
    *GPIO_ODR_LOW = 2;
}
void    p2( void )
{
  while(1)
    *GPIO_ODR_LOW = 4;
}
void    p3( void )
{
  while(1)
    *GPIO_ODR_LOW = 8;
}
```

SV sköter initiering och växling



Task-switch

"Processerna"...

```
void    p0( void )
{
  while(1)
    *GPIO_ODR_LOW = 1;
}
void    p1( void )
{
  while(1)
    *GPIO_ODR_LOW = 2;
}
void    p2( void )
{
  while(1)
    *GPIO_ODR_LOW = 4;
}
void    p3( void )
{
  while(1)
    *GPIO_ODR_LOW = 8;
}
```