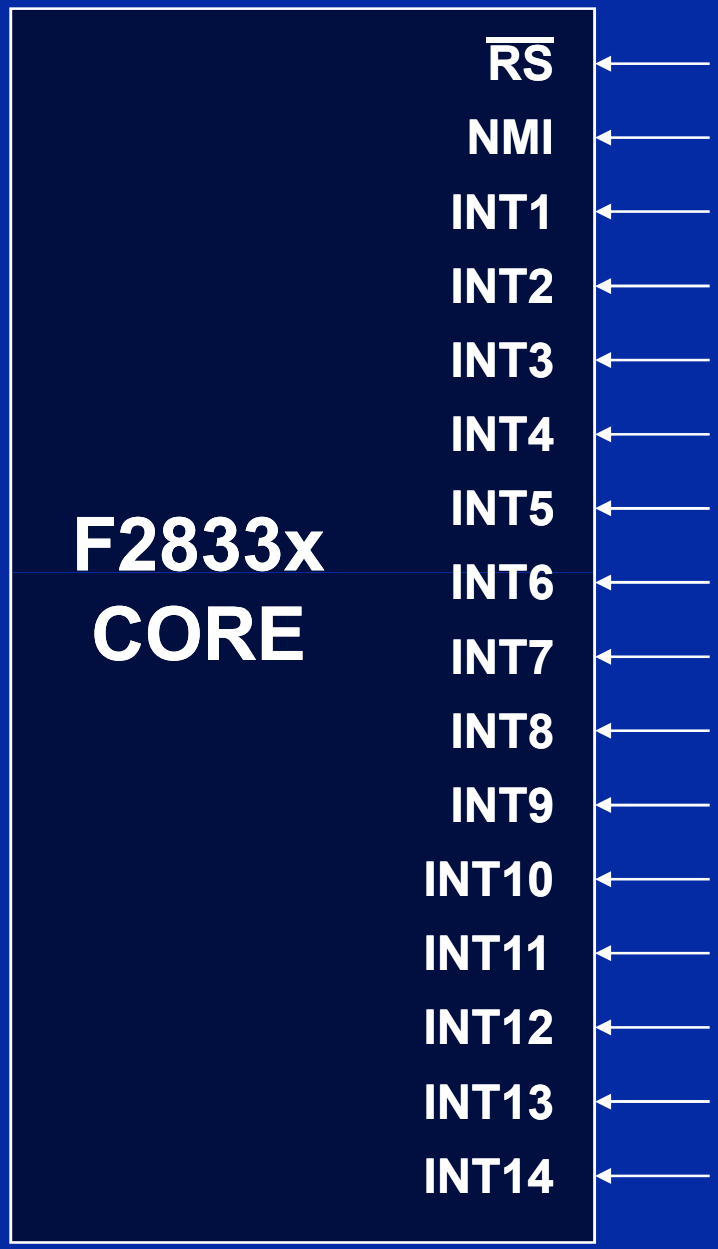
Modul 6: Prekidni sistem F2833x

Digital Signal Controller
TMS320F2833x

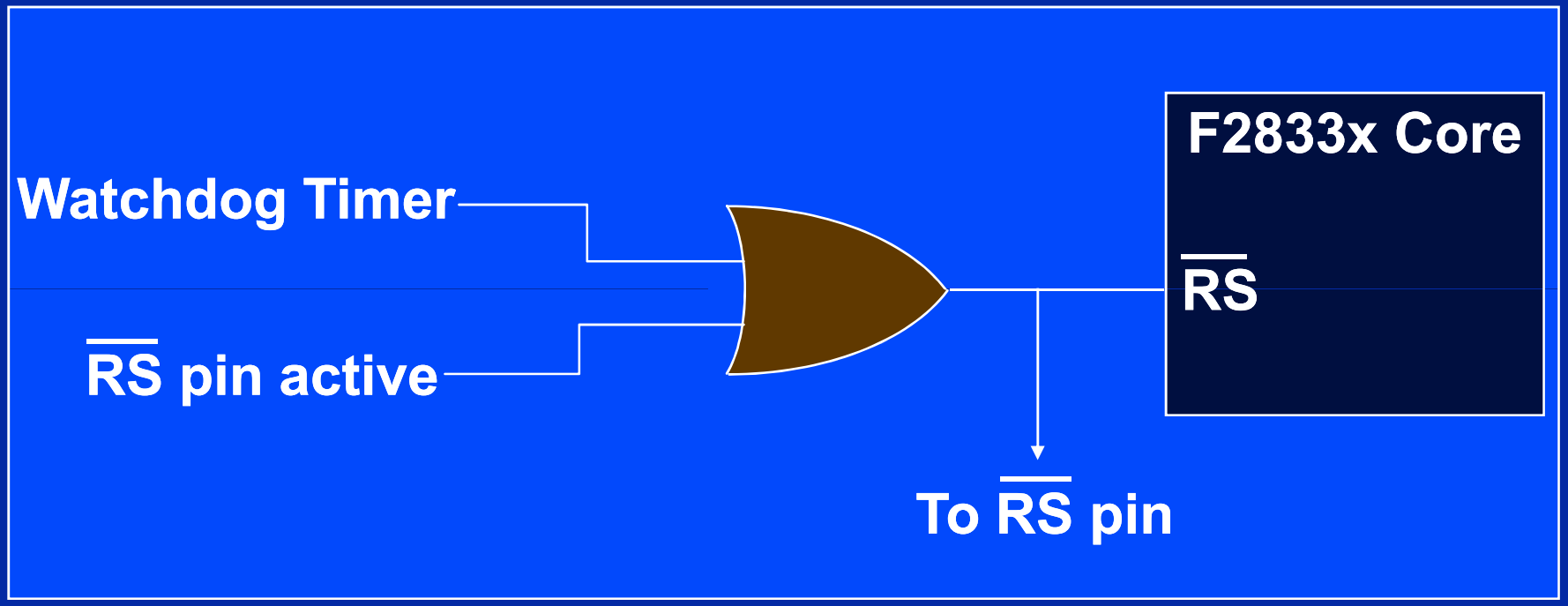
Texas Instruments Incorporated

F2833x Prekidne linije CPU jezgra

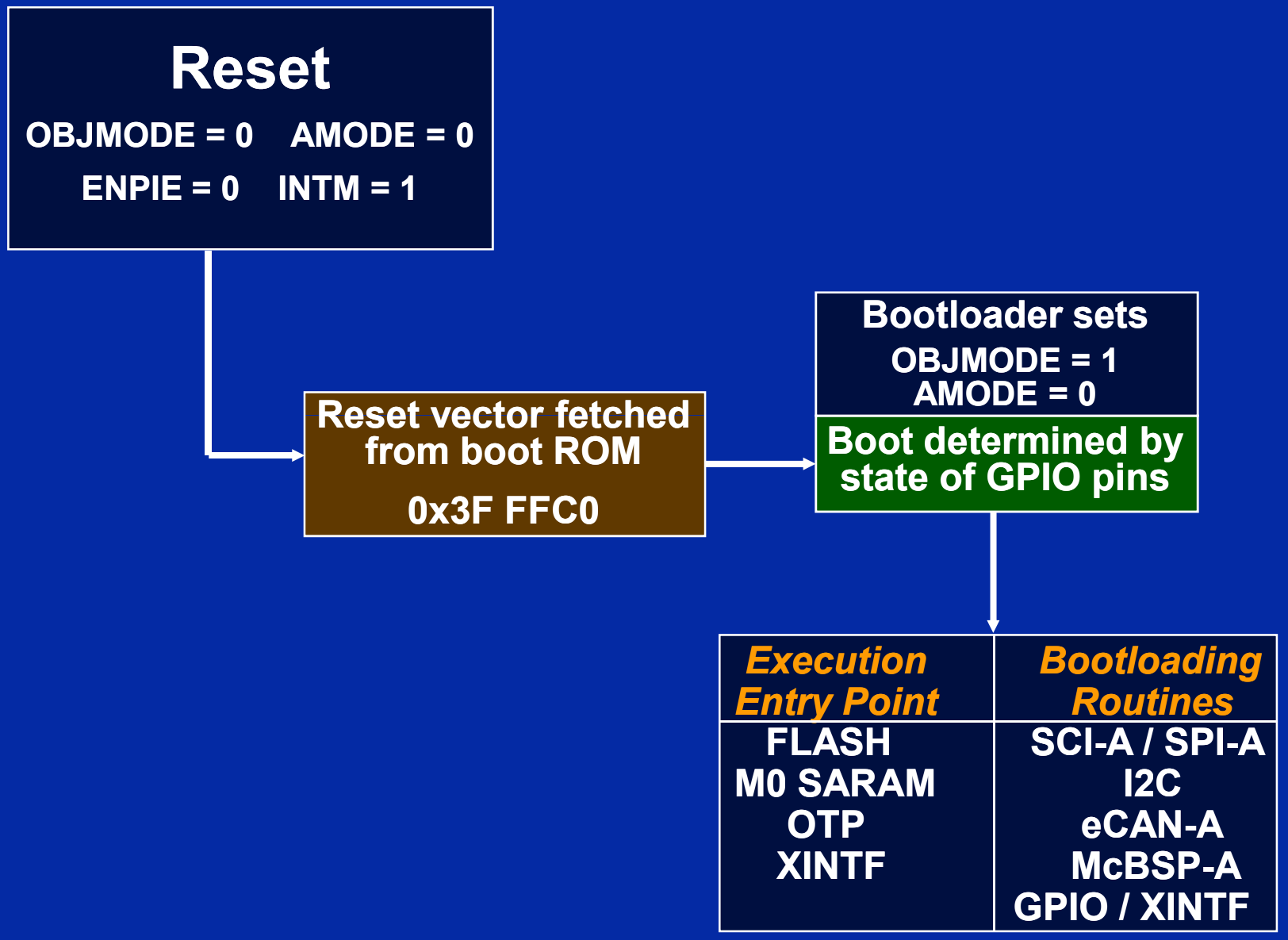


- 2 nemaskirajuća prekida ($\overline{RS}$ i "selectable" NMI)
- 14 maskirajućih prekida (INT1 – INT14)

F2833x Izvor reset signala



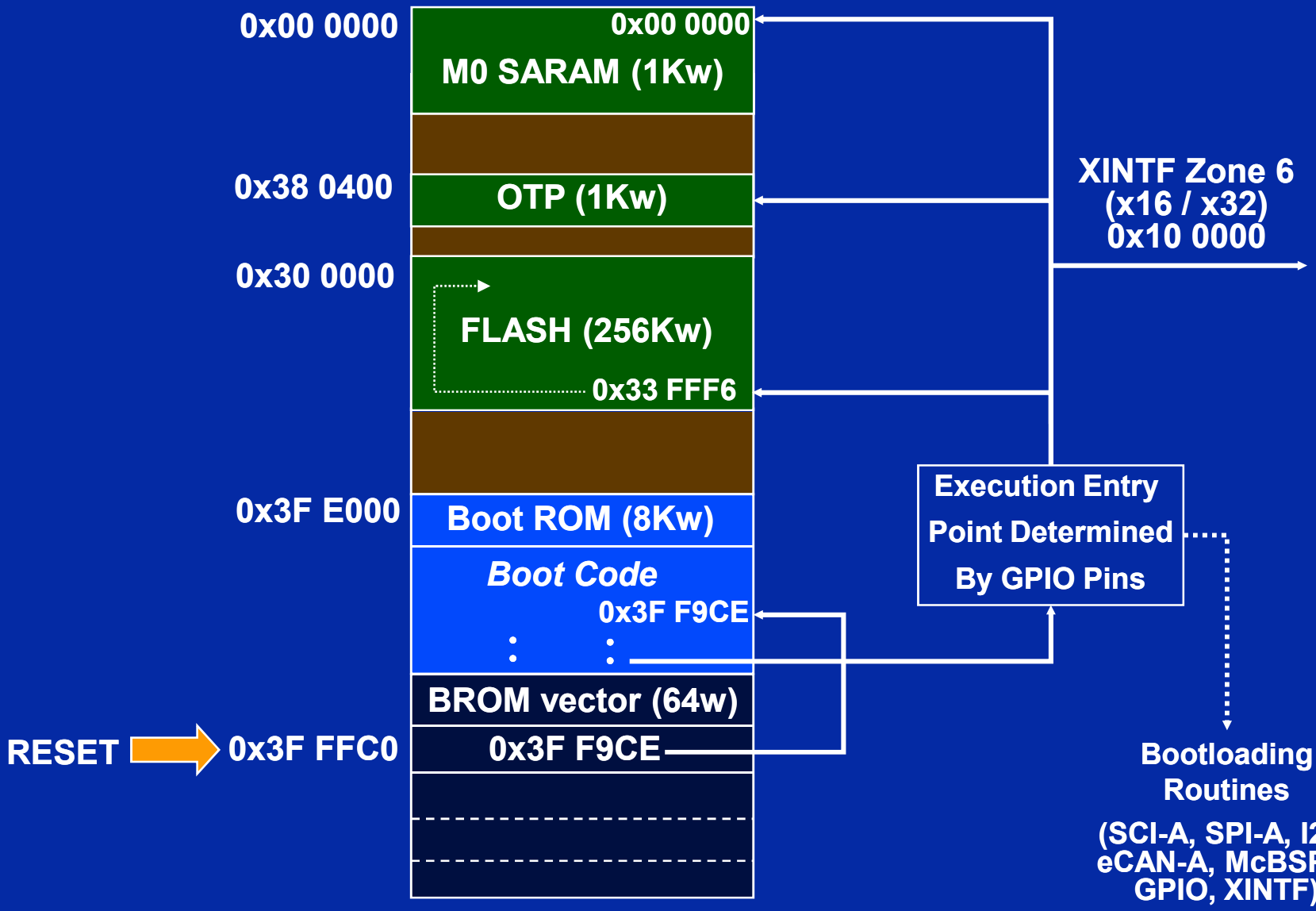
Reset – “Bootloader”



Opcije Bootloader -a

GPIO pins 87 / XA15	86 / XA14	85 / XA13	84 / XA12	
1	1	1	1	jump to *FLASH* address 0x33 FFF6
1	1	1	0	bootload code to on-chip memory via *SCI-A*
1	1	0	1	bootload external EEPROM to on-chip memory via *SPI-A*
1	1	0	0	bootload external EEPROM to on-chip memory via *I2C*
1	0	1	1	Call CAN Boot to load from *eCAN-A* mailbox 1
1	0	1	0	bootload code to on-chip memory via *McBSP-A*
1	0	0	1	jump to *XINTF* Zone 6 address 0x10 0000 for 16-bit data
1	0	0	0	jump to *XINTF* Zone 6 address 0x10 0000 for 32-bit data
0	1	1	1	jump to *OTP* address 0x38 0400
0	1	1	0	bootload code to on-chip memory via *GPIO port A* (parallel)
0	1	0	1	bootload code to on-chip memory via *XINTF* (parallel)
0	1	0	0	jump to *M0 SARAM* address 0x00 0000
0	0	1	1	branch to check boot mode
0	0	1	0	branch to Flash without ADC calibration (TI debug only)
0	0	0	1	branch to M0 SARAM without ADC calibration (TI debug only)
0	0	0	0	branch to SCI-A without ADC calibration (TI debug only)

Tok resetovanja



Inicijalizacija registara u resetu

Register bits defined by reset

PC	0x3F FFC0	PC loaded with reset vector
ACC	0x0000 0000	Accumulator cleared
XAR0 - XAR7	0x0000 0000	Auxiliary Registers
DP	0x0000	Data Page pointer points to page 0
P	0x0000 0000	P register cleared
XT	0x0000 0000	XT register cleared
SP	0x0400	Stack Pointer to address 0400
RPC	0x00 0000	Return Program Counter cleared
IFR	0x0000	no pending interrupts
IER	0x0000	maskable interrupts disabled
DBGIER	0x0000	debug interrupts disabled

Inicijalizacija kontrolnih bitova u resetu

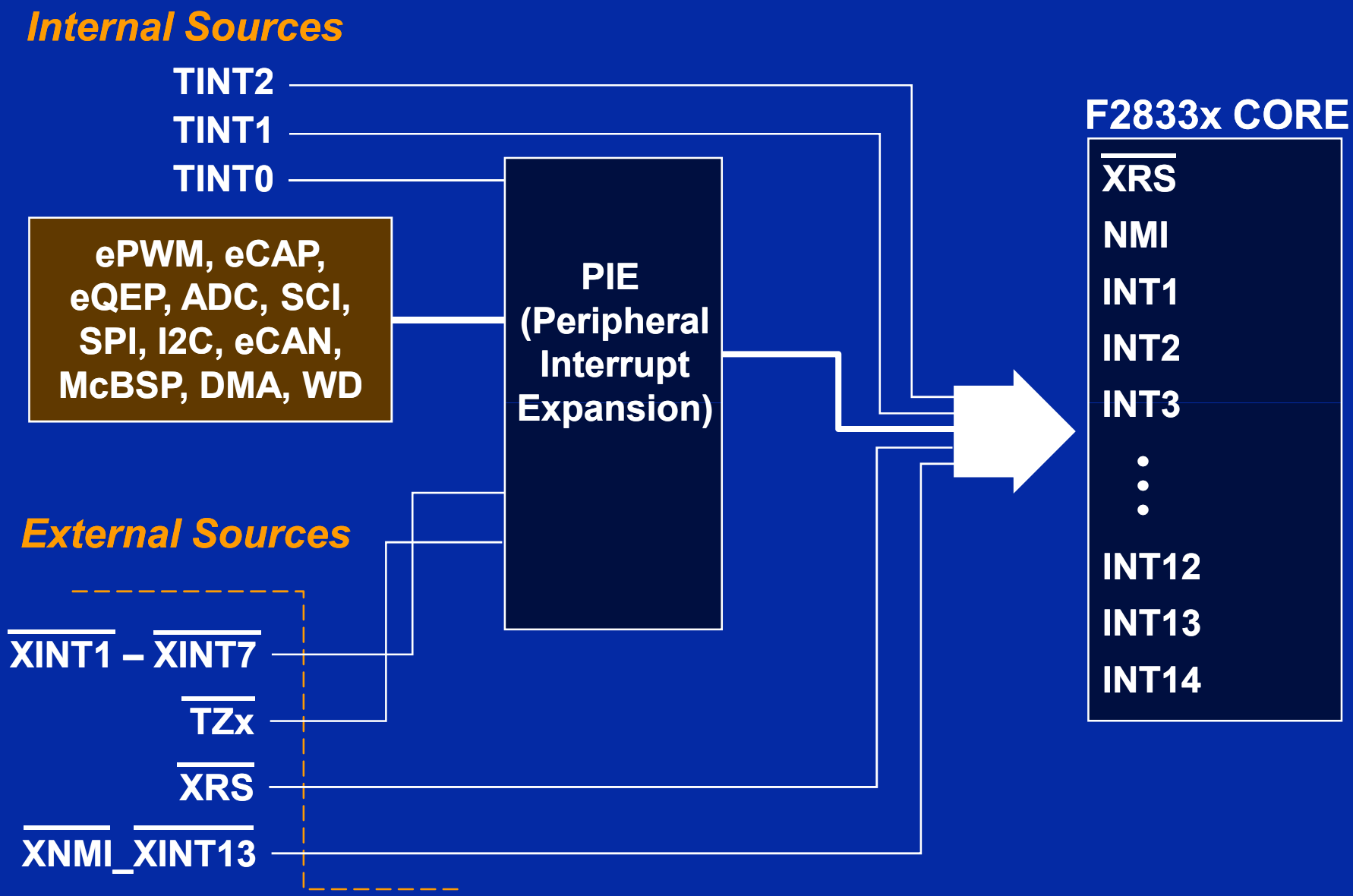
Status Register 0 (ST0)

SXM = 0	Sign extension off		
OVM = 0	Overflow mode off	N = 0	negative flag
TC = 0	test/control flag	V = 0	overflow bit
C = 0	carry bit	PM = 000	set to left-shift-by-1
Z = 0	zero flag	OVC = 00 0000	overflow counter

Status Register 1 (ST1)

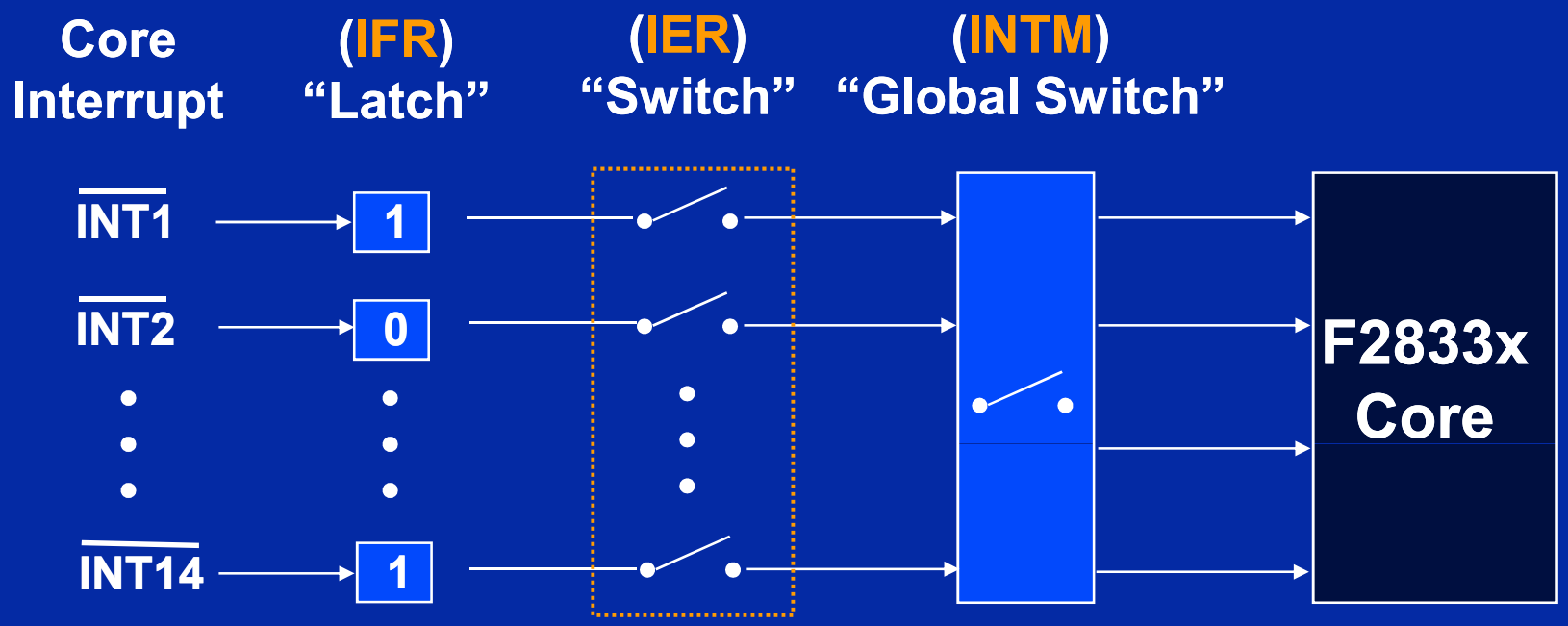
INTM = 1	Disable all maskable interrupts - global
DBGM = 1	Emulation access/events disabled
PAGE0 = 0	Stack addressing mode enabled/Direct addressing disabled
VMAP = 1	Interrupt vectors mapped to PM 0x3F FFC0 – 0x3F FFFF
SPA = 0	stack pointer even address alignment status bit
LOOP = 0	Loop instruction status bit
EALLOW = 0	emulation access enable bit
IDLESTAT = 0	Idle instruction status bit
AMODE = 0	C27x/C28x addressing mode
OBJMODE = 0	C27x object mode
M0M1MAP = 1	mapping mode bit
XF = 0	XF status bit
ARP = 0	ARP points to AR0

Izvori prekida



Procesiranje maskirajućih prekida

Konceptualni pregled



- Validni signal na određenoj prekidnoj liniji uzrokuje postavljanje leča na "1" na odgovarajućoj bit poziciji IFR
- Ako su individualni i globalni prekidači u "on" stanju prekid dolazi do jezgra

Interrupt Flag Register (IFR)

15	14	13	12	11	10	9	8
RTOSINT	DLOGINT	INT14	INT13	INT12	INT11	INT10	INT9

7	6	5	4	3	2	1	0
INT8	INT7	INT6	INT5	INT4	INT3	INT2	INT1

Pending : $IFR_{Bit} = 1$

Absent : $IFR_{Bit} = 0$

```
/*** Manual setting/clearing IFR ***/
extern cregister volatile unsigned int IFR;
    IFR  |= 0x0008;          //set INT4 in IFR
    IFR &= 0xFFF7;           //clear INT4 in IFR
```

- Kompajler generiše atomičnu instruktiju za postavljanje ili brisanje IFR bitova
- Ako dođe do prekida u toku upisa u IFR, prekid ima prioritet
- IFR(bit) se briše kada CPU odgovori na prekid
- IFR se briše pri resetu

Interrupt Enable Register (IER)

15	14	13	12	11	10	9	8
RTOSINT	DLOGINT	INT14	INT13	INT12	INT11	INT10	INT9

7	6	5	4	3	2	1	0
INT8	INT7	INT6	INT5	INT4	INT3	INT2	INT1

Enable: Set $IER_{Bit} = 1$
Disable: Clear $IER_{Bit} = 0$

```
/*** Interrupt Enable Register ***/
extern cregister volatile unsigned int IER;
      IER  |= 0x0008;                  //enable INT4 in IER
      IER &= 0xFFF7;                  //disable INT4 in IER
```

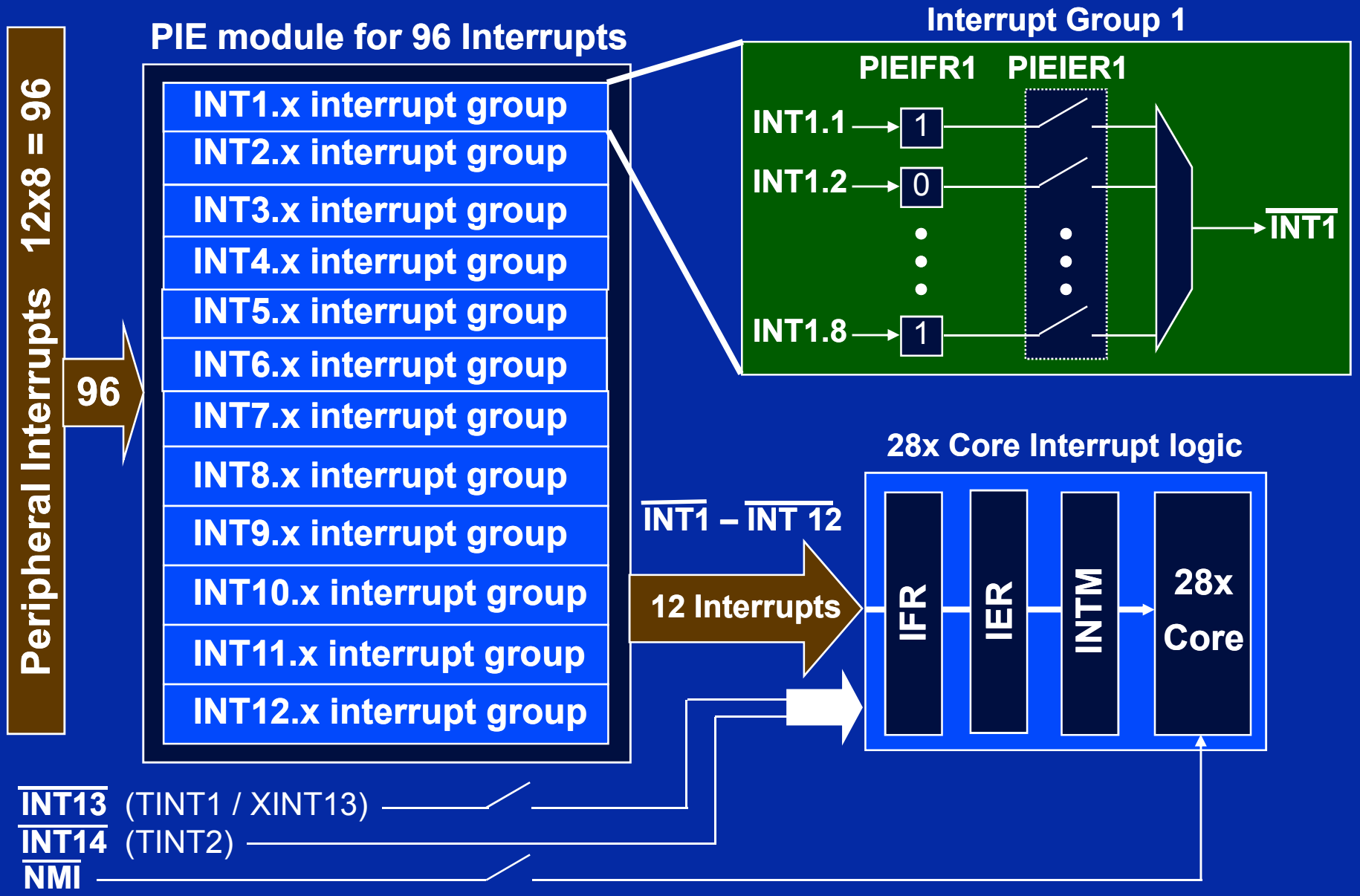
- Kompajler generiše atomičnu instruktiju za postavljanje ili brisanje IFR bitova
- Briše se pri resetu

Interrupt Global Mask Bit

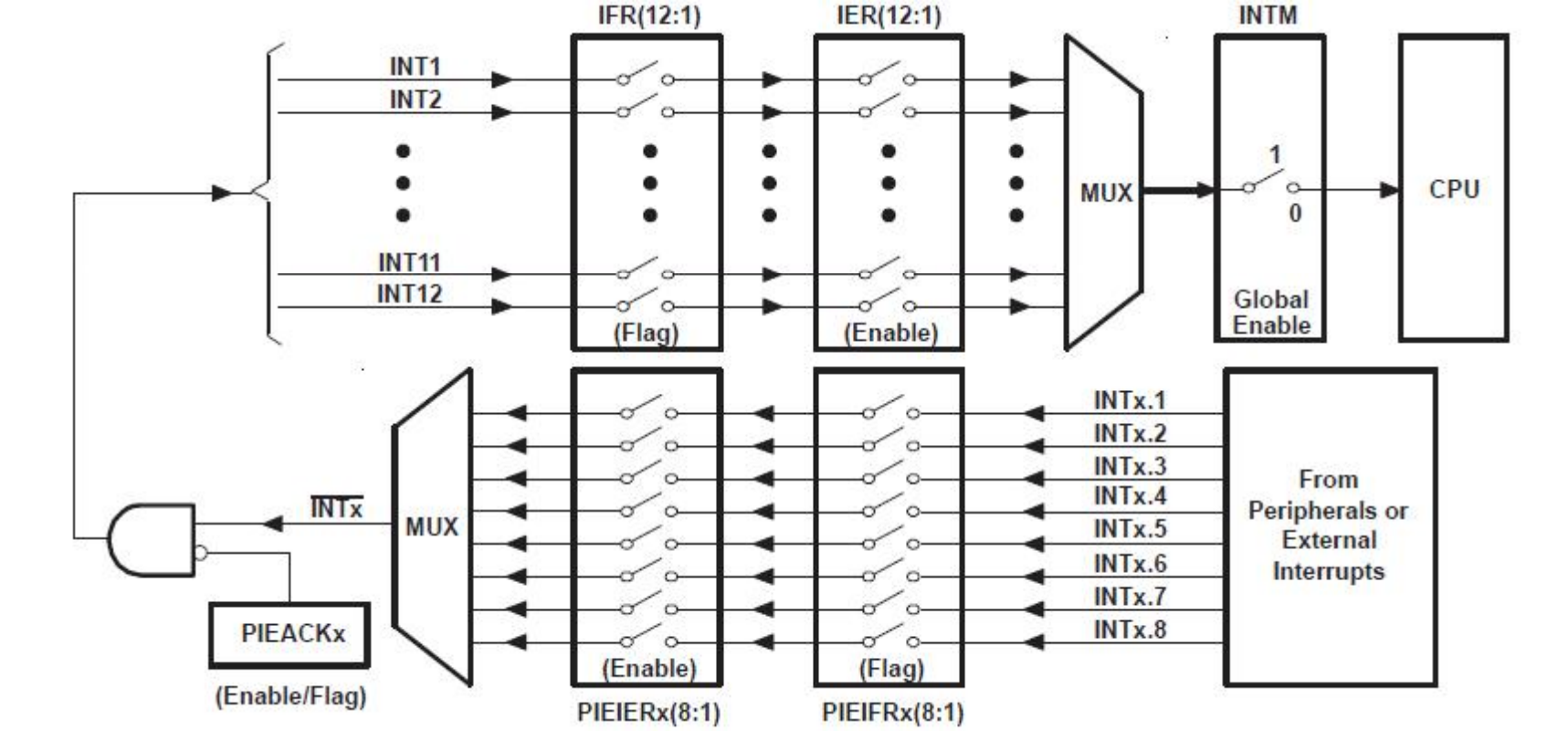
- INTM se koristi za globalnu dozvolu/zabranu prekida:
  - Enable: INTM = 0
  - Disable: INTM = 1 (reset value)
- INTM je moguće modifikovati samo iz asemblerskog koda:

```
/*** Global Interrupts ***/
    asm(“  CLRC  INTM”);     //enable global interrupts
    asm(“  SETC  INTM”);     //disable global interrupts
```

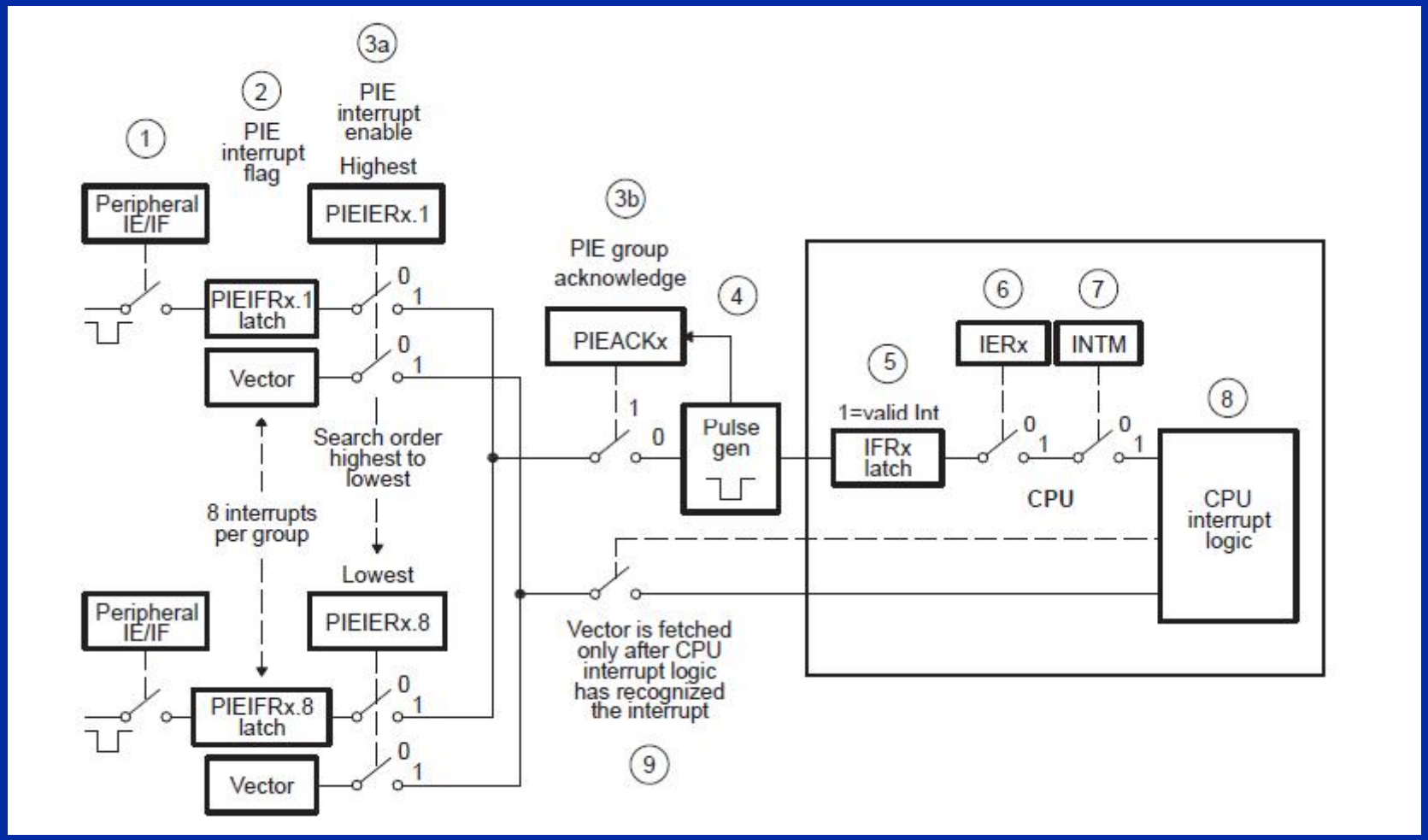
Peripheral Interrupt Expansion - PIE



Multipleksiranje prekida pomoću PIE



Dijagram toka zahteva za prekid (PIE)



PIE Registri

PIEIFRx registar (x = 1 to 12)

15 - 8	7	6	5	4	3	2	1	0
reserved	INTx.8	INTx.7	INTx.6	INTx.5	INTx.4	INTx.3	INTx.2	INTx.1

PIEIERx registar (x = 1 to 12)

15 - 8	7	6	5	4	3	2	1	0
reserved	INTx.8	INTx.7	INTx.6	INTx.5	INTx.4	INTx.3	INTx.2	INTx.1

PIE Interrupt Acknowledge Registar (PIEACK)

15 - 12	11	10	9	8	7	6	5	4	3	2	1	0
reserved	PIEACKx											

PIECTRL registar

15 - 1	0
PIEVECT	ENPIE

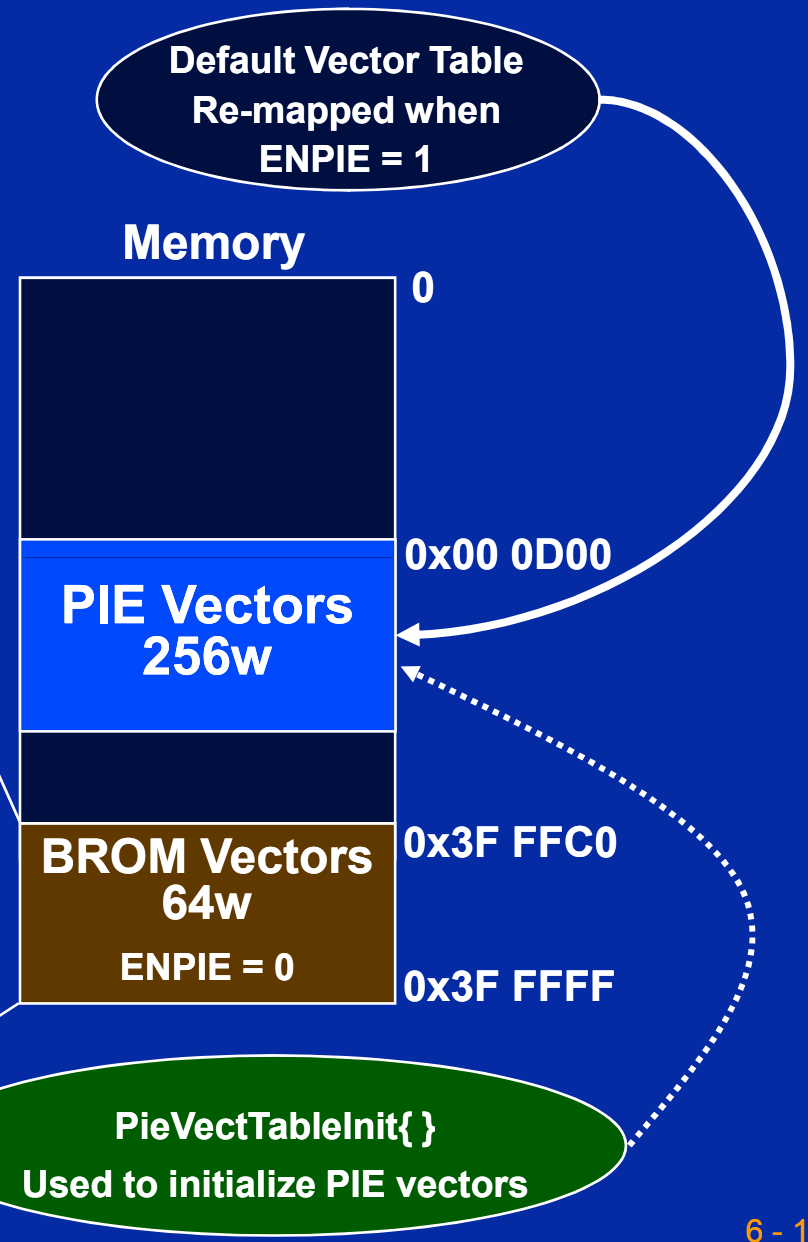
```
#include “DSP2833x_Device.h”
   PieCtrlRegs.PIEIFR1.bit.INTx4 = 1;    //manually set IFR for XINT1 in PIE group 1
   PieCtrlRegs.PIEIER3.bit.INTx5 = 1;    //enable CAPINT1 in PIE group 3
   PieCtrlRegs.PIEACK.all = 0x0004;      //acknowledge the PIE group 3
   PieCtrlRegs.PIECTRL.bit.ENPIE = 1;  //enable the PIE
```

F2833x PIE Tabela dodele prekida

	INTx.8	INTx.7	INTx.6	INTx.5	INTx.4	INTx.3	INTx.2	INTx.1
INT1	WAKEINT	TINT0	ADCINT	XINT2	XINT1		SEQ2INT	SEQ1INT
INT2			EPWM6_TZINT	EPWM5_TZINT	EPWM4_TZINT	EPWM3_TZINT	EPWM2_TZINT	EPWM1_TZINT
INT3			EPWM6_INT	EPWM5_INT	EPWM4_INT	EPWM3_INT	EPWM2_INT	EPWM1_INT
INT4			ECAP6_INT	ECAP5_INT	ECAP4_INT	ECAP3_INT	ECAP2_INT	ECAP1_INT
INT5							EQEP2_INT	EQEP1_INT
INT6			MXINTA	MRINTA	MXINTB	MRINTB	SPITXINTA	SPIRXINTA
INT7			DINTCH6	DINTCH5	DINTCH4	DINTCH3	DINTCH2	DINTCH1
INT8			SCITXINTC	SCIRXINTC			I2CINT2A	I2CINT1A
INT9	ECAN1_INTB	ECAN0_INTB	ECAN1_INTA	ECAN0_INTA	SCITXINTB	SCIRXINTB	SCITXINTA	SCIRXINTA
INT10								
INT11								
INT12	LUF	LVF		XINT7	XINT6	XINT5	XINT4	XINT3

Predefinisana vektorska tabela

Vector	Offset
RESET	00
INT1	02
INT2	04
INT3	06
INT4	08
INT5	0A
INT6	0C
INT7	0E
INT8	10
INT9	12
INT10	14
INT11	16
INT12	18
INT13	1A
INT14	1C
DATALOG	1E
RTOSINT	20
EMUINT	22
NMI	24
ILLEGAL	26
USER 1-12	28-3E

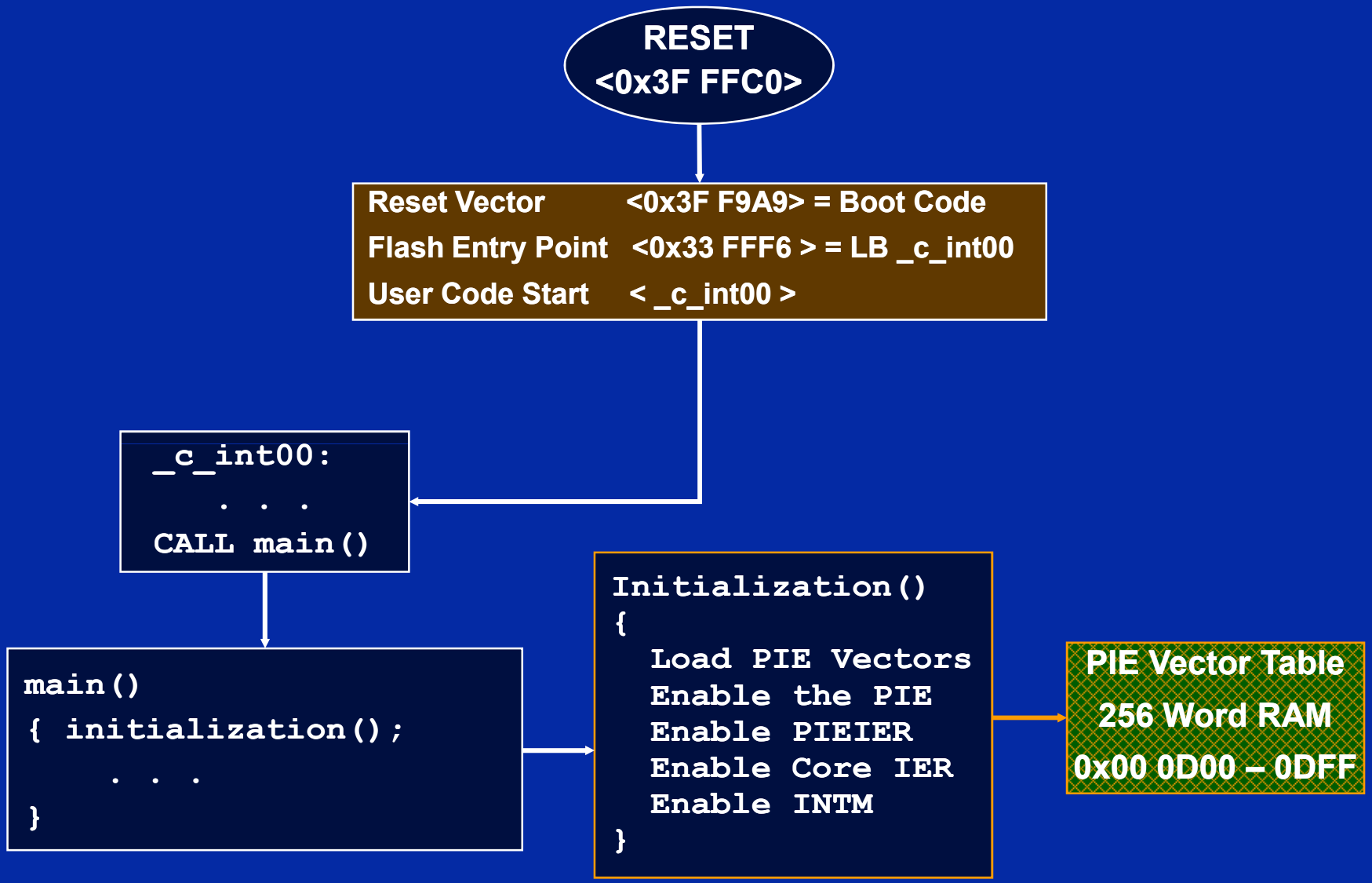


Mapiranje PIE vektora (ENPIE = 1)

Vector name	PIE vector address	PIE vector Description
not used	0x00 0D00	Reset vector (never fetched here)
INT1	0x00 0D02	INT1 re-mapped to PIE group below
……	……	…… re-mapped to PIE group below
INT12	0x00 0D18	INT12 re-mapped to PIE group below
INT13	0x00 0D1A	XINT13 Interrupt or CPU Timer 1 (RTOS)
INT14	0x00 0D1C	CPU Timer 2 (RTOS)
DATALOG	0x00 0D1D	CPU Data logging Interrupt
……	……	……
USER12	0x00 0D3E	User-defined Trap
INT1.1	0x00 0D40	PIEINT1.1 Interrupt Vector
……	……	……
INT1.8	0x00 0D4E	PIEINT1.8 Interrupt Vector
……	……	……
INT12.1	0x00 0DF0	PIEINT12.1 Interrupt Vector
……	……	……
INT12.8	0x00 0DFE	PIEINT12.8 Interrupt Vector

- PIE vector lokacije – 0x00 0D00 – 256 reči u *data* memoriji
- RESET i INT1-INT12 vektorse lokacije se re-mapiraju
- CPU vectori se re-mapiraju u 0x00 0D00 *data* memorije

Mapiranje prekidnih vektora



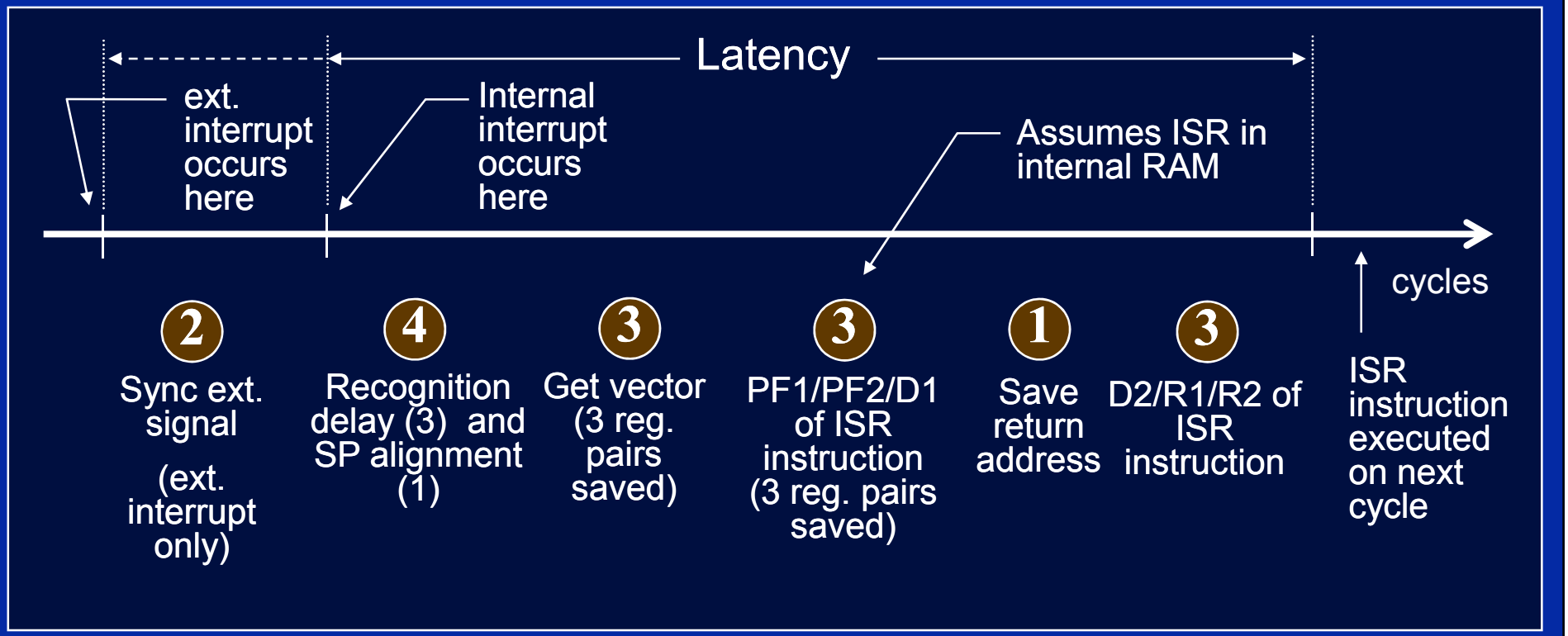
Odziv na prekid – Harderska sekvenca

CPU Action	Description
Registers → stack	14 Register words auto saved
0 → IFR (bit)	Clear corresponding IFR bit
0 → IER (bit)	Clear corresponding IER bit
1 → INTM/DBGM	Disable global ints/debug events
Vector → PC	Loads PC with int vector address
Clear other status bits	Clear LOOP, EALLOW, IDLESTAT

Note: some actions occur simultaneously, none are interruptible

T	ST0
AH	AL
PH	PL
AR1	AR0
DP	ST1
DBSTAT	IER
PC(msw)	PC(lsw)

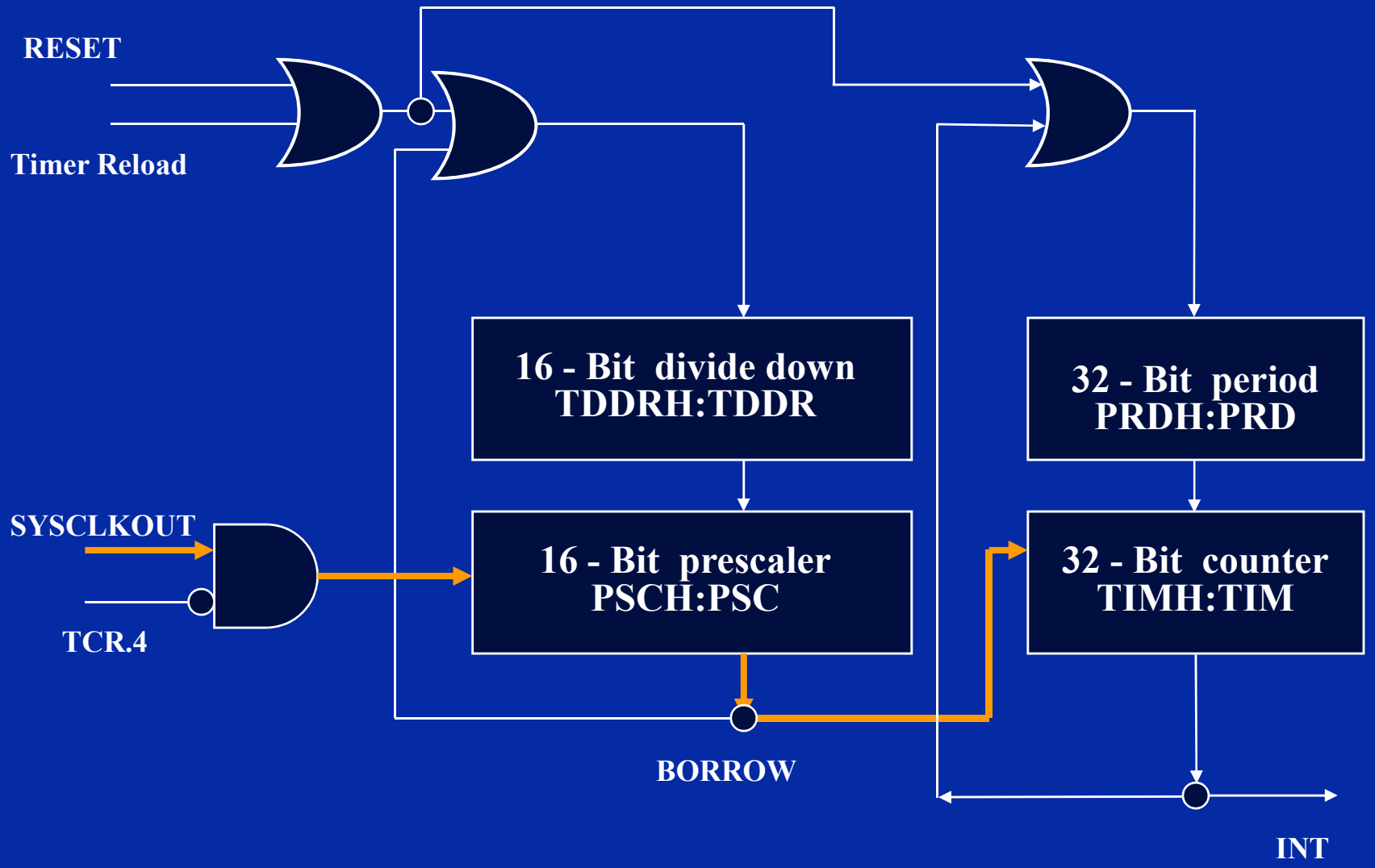
Latencija prekida



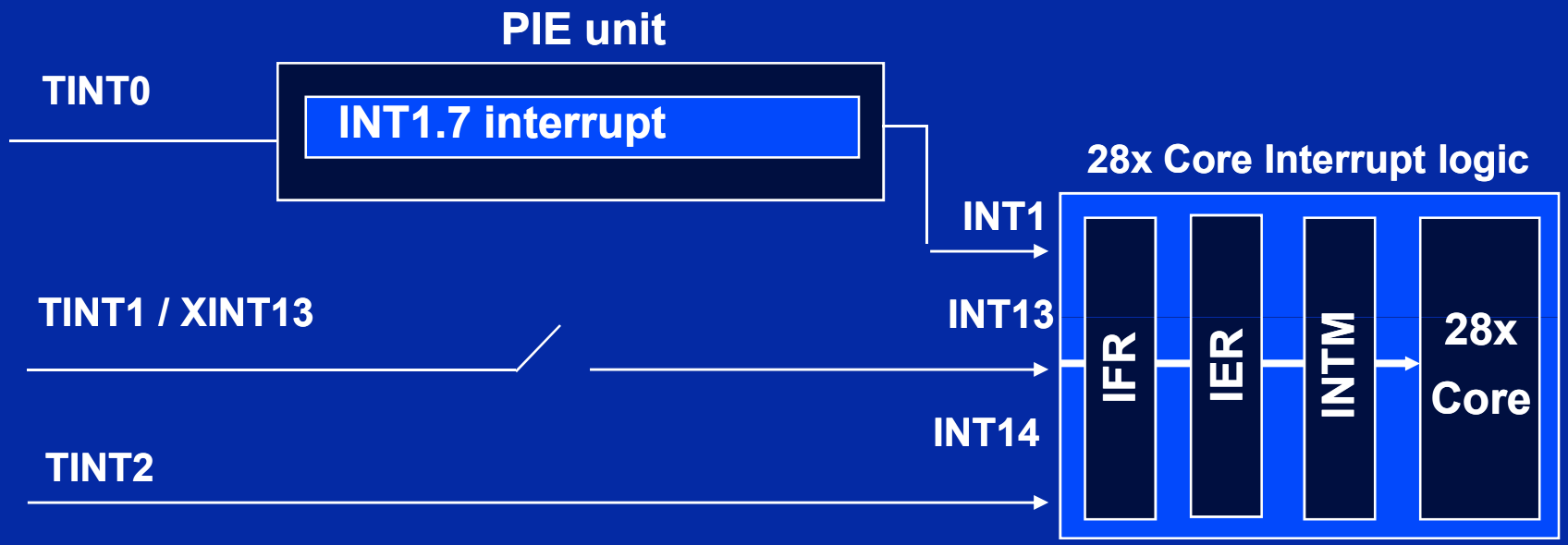
Above is for PIE enabled or disabled

- **Minimalna latencija (kada se obrada vrši unutar ISR):**
  - **Internal interrupts: 14 cycles**
  - **External interrupts: 16 cycles**
- **Maksimalna latencija: Zavisi od *wait* statanja, *ready*, INTM, ...**

F2833x CPU Tajmeri



F2833x *Timer Interrupt System*



F2833x Tajmer registri

Address	Register	Name
0x0000 0C00	TIMER0TIM	Timer 0, Counter Register Low
0x0000 0C01	TIMER0TIMH	Timer 0, Counter Register High
0x0000 0C02	TIMER0PRD	Timer 0, Period Register Low
0x0000 0C03	TIMER0PRDH	Timer 0, Period Register High
0x0000 0C04	TIMER0TCR	Timer 0, Control Register
0x0000 0C06	TIMER0TPR	Timer 0, Prescaler Register
0x0000 0C07	TIMER0TPRH	Timer 0, Prescaler Register High
0x0000 0C08	TIMER1TIM	Timer 1, Counter Register Low
0x0000 0C09	TIMER1TIMH	Timer 1, Counter Register High
0x0000 0C0A	TIMER1PRD	Timer 1, Period Register Low
0x0000 0C0B	TIMER1PRDH	Timer 1, Period Register High
0x0000 0C0C	TIMER1TCR	Timer 1, Control Register
0x0000 0C0D	TIMER1TPR	Timer 1, Prescaler Register
0x0000 0C0F	TIMER1TPRH	Timer 1, Prescaler Register High
0x0000 0C10 to 0C17	Timer 2 Registers ; same layout as above	

F2833x Timer Control Registers
TIMERxTCR

Timer Interrupt Flag
Write 1 clear bit

Timer Interrupt Enable
Write 1 to enable INT

Emulator Interaction
1x = run free

Timer Reload Bit
1 = reload

Timer Stop Status
0 = start / 1 = stop