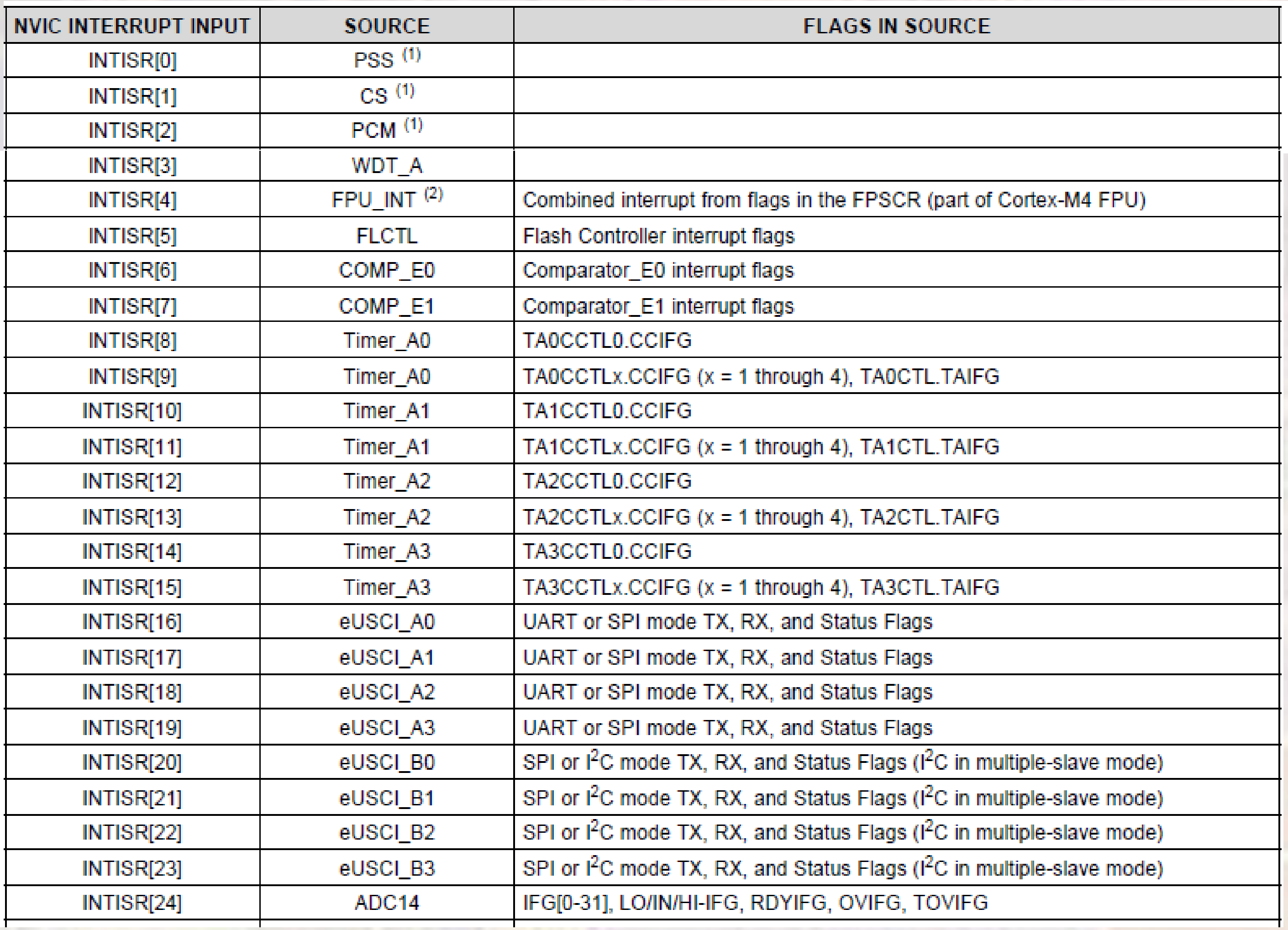
Interrupts

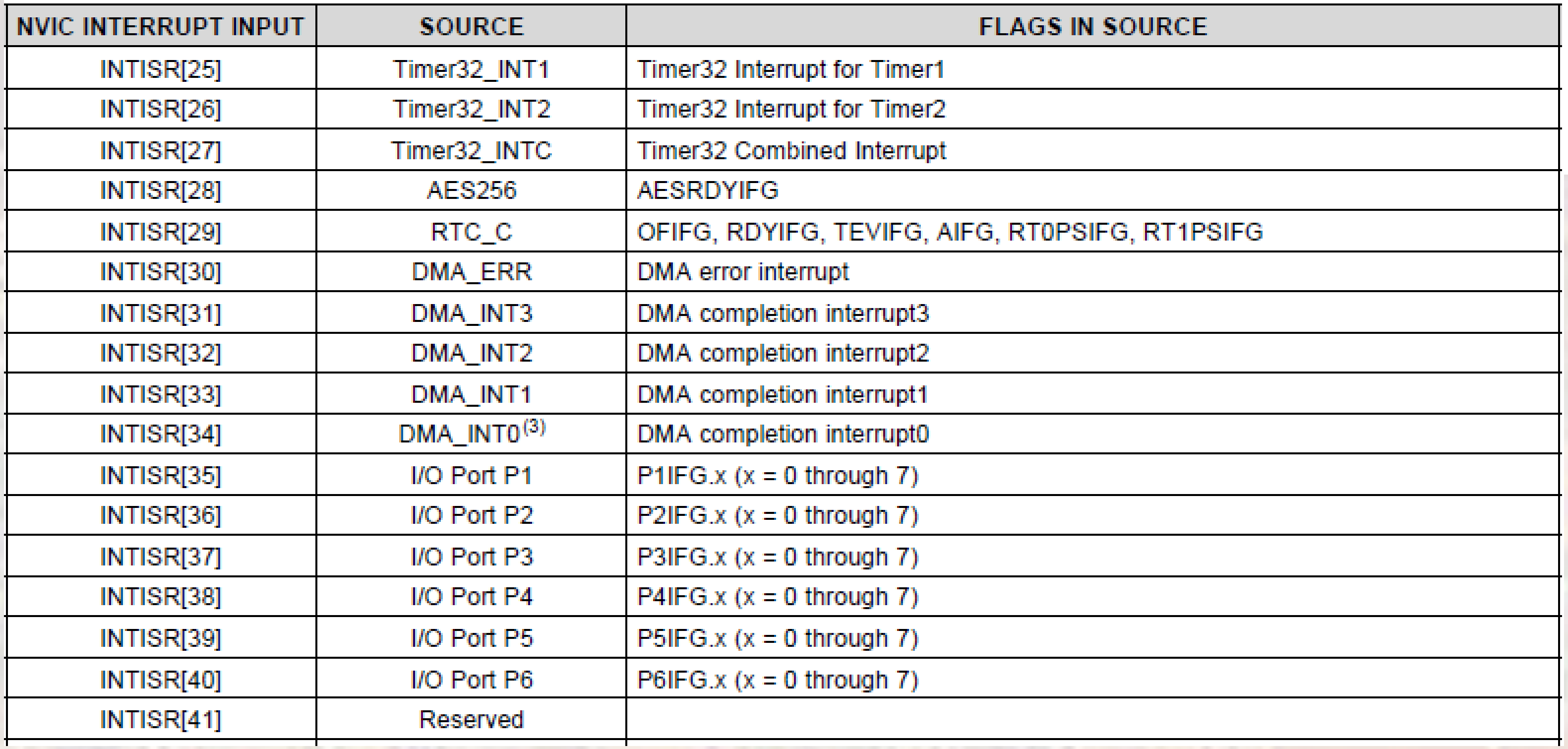
- Interrupt Vectors



NVIC INTERRUPT INPUT	SOURCE	FLAGS IN SOURCE
INTISR[0]	PSS [1]	
INTISR[1]	CS [1]	
INTISR[2]	PCM [1]	
INTISR[3]	WDT_A	
INTISR[4]	FPU_INT [2]	Combined interrupt from flags in the FPSCR (part of Cortex-M4 FPU)
INTISR[5]	FLCTL	Flash Controller interrupt flags
INTISR[6]	COMP_E0	Comparator_E0 interrupt flags
INTISR[7]	COMP_E1	Comparator_E1 interrupt flags
INTISR[8]	Timer_A0	TA0CCTL0.CCIFG
INTISR[9]	Timer_A0	TA0CCTLx.CCIFG (x = 1 through 4), TA0CTL.TAIFG
INTISR[10]	Timer_A1	TA1CCTL0.CCIFG
INTISR[11]	Timer_A1	TA1CCTLx.CCIFG (x = 1 through 4), TA1CTL.TAIFG
INTISR[12]	Timer_A2	TA2CCTL0.CCIFG
INTISR[13]	Timer_A2	TA2CCTLx.CCIFG (x = 1 through 4), TA2CTL.TAIFG
INTISR[14]	Timer_A3	TA3CCTL0.CCIFG
INTISR[15]	Timer_A3	TA3CCTLx.CCIFG (x = 1 through 4), TA3CTL.TAIFG
INTISR[16]	eUSCI_A0	UART or SPI mode TX, RX, and Status Flags
INTISR[17]	eUSCI_A1	UART or SPI mode TX, RX, and Status Flags
INTISR[18]	eUSCI_A2	UART or SPI mode TX, RX, and Status Flags
INTISR[19]	eUSCI_A3	UART or SPI mode TX, RX, and Status Flags
INTISR[20]	eUSCI_B0	SPI or I^{2}C mode TX, RX, and Status Flags (I^{2}C in multiple-slave mode)
INTISR[21]	eUSCI_B1	SPI or I^{2}C mode TX, RX, and Status Flags (I^{2}C in multiple-slave mode)
INTISR[22]	eUSCI_B2	SPI or I^{2}C mode TX, RX, and Status Flags (I^{2}C in multiple-slave mode)
INTISR[23]	eUSCI_B3	SPI or I^{2}C mode TX, RX, and Status Flags (I^{2}C in multiple-slave mode)
INTISR[24]	ADC14	IFG[0-31], LO/IN/HI-IFG, RDYIFG, OVIFG, TOVIFG

Interrupts

- Interrupt Vectors



NVIC INTERRUPT INPUT	SOURCE	FLAGS IN SOURCE
INTISR[25]	Timer32_INT1	Timer32 Interrupt for Timer1
INTISR[26]	Timer32_INT2	Timer32 Interrupt for Timer2
INTISR[27]	Timer32_INTC	Timer32 Combined Interrupt
INTISR[28]	AES256	AESRDYIFG
INTISR[29]	RTC_C	OFIFG, RDYIFG, TEVIFG, AIFG, RT0PSIFG, RT1PSIFG
INTISR[30]	DMA_ERR	DMA error interrupt
INTISR[31]	DMA_INT3	DMA completion interrupt3
INTISR[32]	DMA_INT2	DMA completion interrupt2
INTISR[33]	DMA_INT1	DMA completion interrupt1
INTISR[34]	DMA_INT0(3)	DMA completion interrupt0
INTISR[35]	I/O Port P1	P1IFG.x (x = 0 through 7)
INTISR[36]	I/O Port P2	P2IFG.x (x = 0 through 7)
INTISR[37]	I/O Port P3	P3IFG.x (x = 0 through 7)
INTISR[38]	I/O Port P4	P4IFG.x (x = 0 through 7)
INTISR[39]	I/O Port P5	P5IFG.x (x = 0 through 7)
INTISR[40]	I/O Port P6	P6IFG.x (x = 0 through 7)
INTISR[41]	Reserved	

Interrupts

- Interrupt Vector Handlers

```
void (* const interruptVectors[])(void) =
{
   (void (*)(void))((uint32_t)&__STACK_END),
                                /* The initial stack pointer */
   Reset_Handler,               /* The reset handler         */
   NMI_Handler,                 /* The NMI handler           */
   HardFault_Handler,           /* The hard fault handler    */
   MemManage_Handler,           /* The MPU fault handler     */
   BusFault_Handler,            /* The bus fault handler     */
   UsageFault_Handler,          /* The usage fault handler   */
   0,                           /* Reserved                  */
   0,                           /* Reserved                  */
   0,                           /* Reserved                  */
   0,                           /* Reserved                  */
   SVC_Handler,                 /* SVCall handler            */
   DebugMon_Handler,            /* Debug monitor handler     */
   0,                           /* Reserved                  */
   PendSV_Handler,              /* The PendSV handler        */
   SysTick_Handler,             /* The SysTick handler       */
   PSS_IRQHandler,              /* PSS ISR                   */
   CS_IRQHandler,               /* CS ISR                    */
   PCM_IRQHandler,              /* PCM ISR                   */
   WDT_A_IRQHandler,            /* WDT ISR                   */
   FPU_IRQHandler,              /* FPU ISR                   */
   FLCTL_IRQHandler,            /* FLCTL ISR                 */
   COMP_E0_IRQHandler,          /* COMP0 ISR                 */
   COMP_E1_IRQHandler,          /* COMP1 ISR                 */
   TA0_0_IRQHandler,            /* TA0_0 ISR                 */
   TA0_N_IRQHandler,            /* TA0_N ISR                 */
   TA1_0_IRQHandler,            /* TA1_0 ISR                 */
   TA1_N_IRQHandler,            /* TA1_N ISR                 */
   TA2_0_IRQHandler,            /* TA2_0 ISR                 */
   TA2_N_IRQHandler,            /* TA2_N ISR                 */
   TA3_0_IRQHandler,            /* TA3_0 ISR                 */
   TA3_N_IRQHandler,            /* TA3_N ISR                 */
   EUSCIA0_IRQHandler,          /* EUSCIA0 ISR               */
   EUSCIA1_IRQHandler,          /* EUSCIA1 ISR               */
   EUSCIA2_IRQHandler,          /* EUSCIA2 ISR               */
   EUSCIA3_IRQHandler,          /* EUSCIA3 ISR               */
   EUSCIB0_IRQHandler,          /* EUSCIB0 ISR               */
   EUSCIB1_IRQHandler,          /* EUSCIB1 ISR               */
   EUSCIB2_IRQHandler,          /* EUSCIB2 ISR               */
   EUSCIB3_IRQHandler,          /* EUSCIB3 ISR               */
   ADC14_IRQHandler,            /* ADC14 ISR                 */
   T32_INT1_IRQHandler,         /* T32_INT1 ISR              */
   T32_INT2_IRQHandler,         /* T32_INT2 ISR              */
   T32_INTC_IRQHandler,         /* T32_INTC ISR              */
   AES256_IRQHandler,           /* AES ISR                   */
   RTC_C_IRQHandler,            /* RTC ISR                   */
   DMA_ERR_IRQHandler,          /* DMA_ERR ISR               */
   DMA_INT3_IRQHandler,         /* DMA_INT3 ISR              */
   DMA_INT2_IRQHandler,         /* DMA_INT2 ISR              */
   DMA_INT1_IRQHandler,         /* DMA_INT1 ISR              */
   DMA_INT0_IRQHandler,         /* DMA_INT0 ISR              */
   PORT1_IRQHandler,            /* PORT1 ISR                 */
   PORT2_IRQHandler,            /* PORT2 ISR                 */
   PORT3_IRQHandler,            /* PORT3 ISR                 */
   PORT4_IRQHandler,            /* PORT4 ISR                 */
   PORT5_IRQHandler,            /* PORT5 ISR                 */
   PORT6_IRQHandler             /* PORT6 ISR                 */
};
```