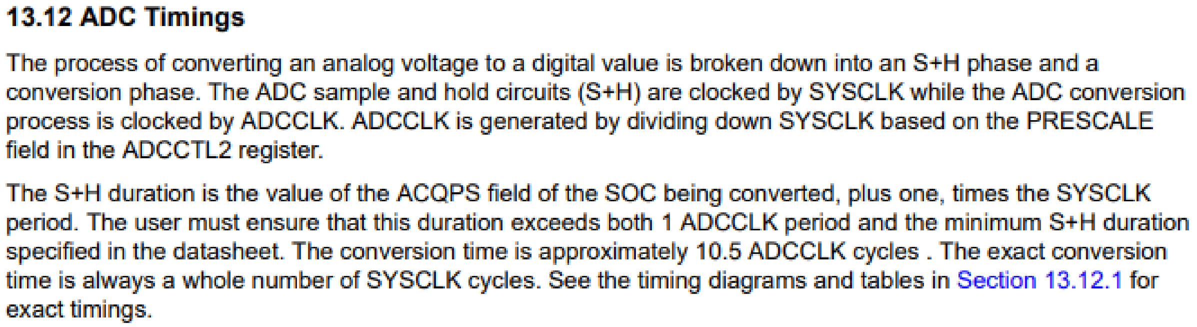
AdcaRegs.ADCCTL2.bit.PRESCALE pg(1463,1465,1501)

Def: Sets the ADC clock. EE533: ADC clock = system clock

```
AdcaRegs.ADCCTL2.bit.PRESCALE = 0;
```



13.12 ADC Timings

The process of converting an analog voltage to a digital value is broken down into an S+H phase and a conversion phase. The ADC sample and hold circuits (S+H) are clocked by SYSCLK while the ADC conversion process is clocked by ADCCLK. ADCCLK is generated by dividing down SYSCLK based on the PRESCALE field in the ADCCTL2 register.

The S+H duration is the value of the ACQPS field of the SOC being converted, plus one, times the SYSCLK period. The user must ensure that this duration exceeds both 1 ADCCLK period and the minimum S+H duration specified in the datasheet. The conversion time is approximately 10.5 ADCCLK cycles . The exact conversion time is always a whole number of SYSCLK cycles. See the timing diagrams and tables in Section 13.12.1 for exact timings.

www.ti.com Analog-to-Digital Converter (ADC)

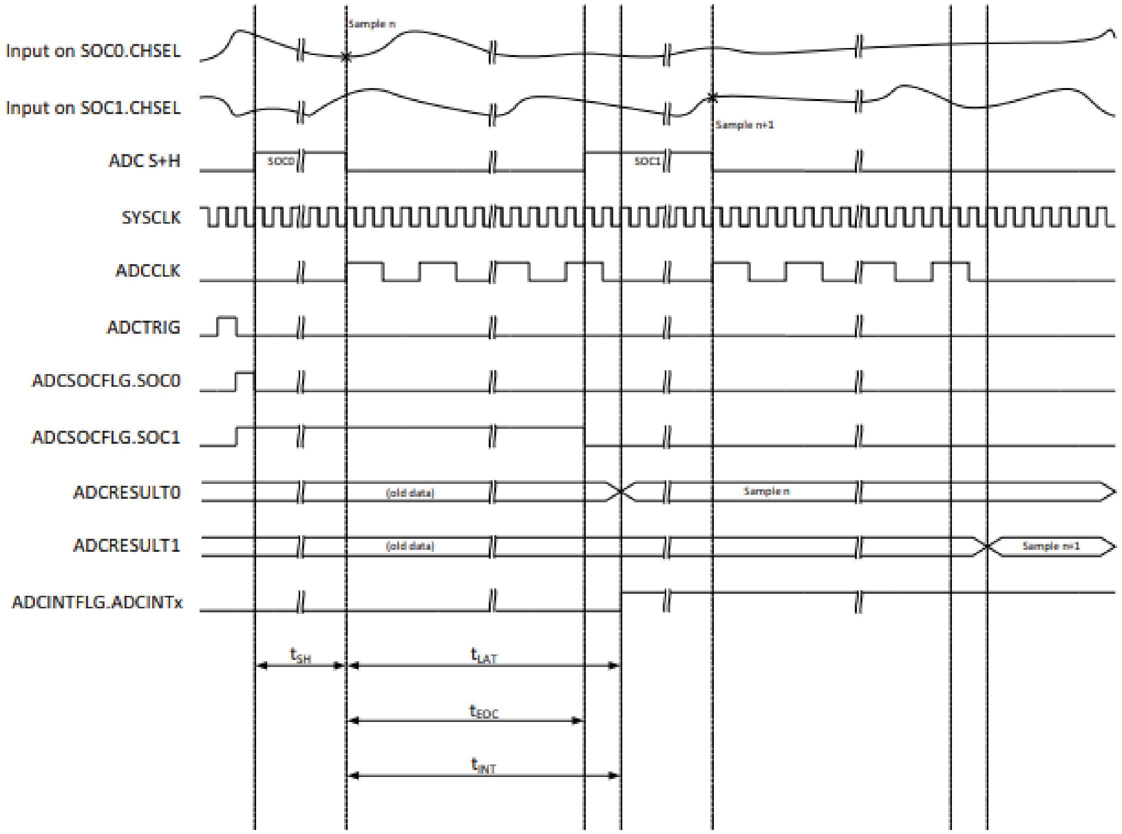


Figure 13-13. ADC Timings for 12-bit Mode in Late Interrupt Mode

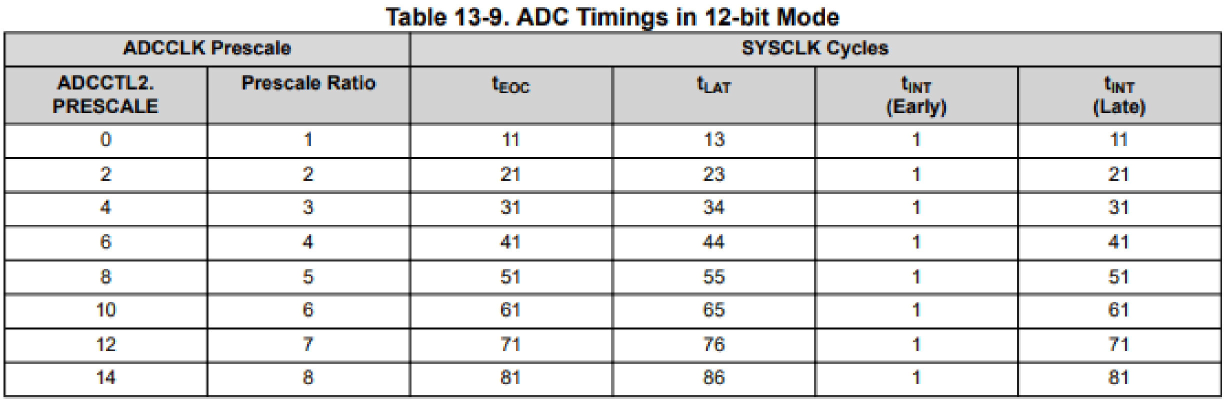


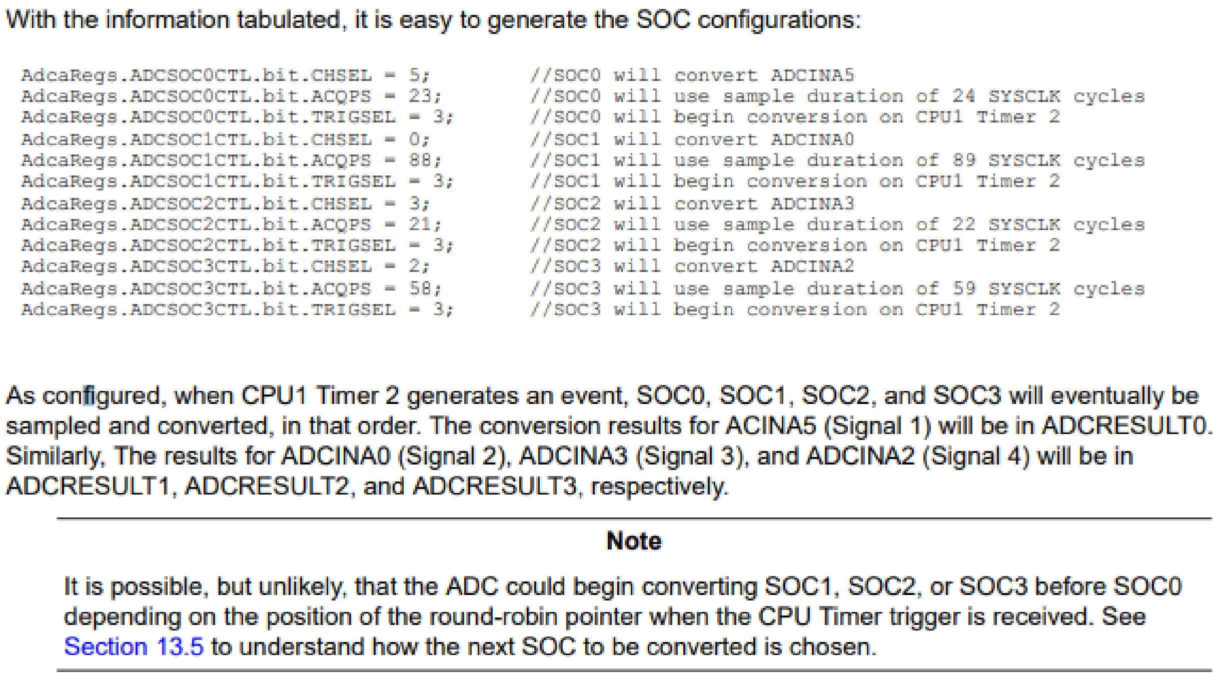
Table 13-9. ADC Timings in 12-bit Mode

ADCCLK Prescale		SYSCLK Cycles			
ADCCTL2. PRESCALE	Prescale Ratio	t_{EOC}	t_{LAT}	t_{INT} (Early)	t_{INT} (Late)
0	1	11	13	1	11
2	2	21	23	1	21
4	3	31	34	1	31
6	4	41	44	1	41
8	5	51	55	1	51
10	6	61	65	1	61
12	7	71	76	1	71
14	8	81	86	1	81

AdcaRegs.ADCSOCxCTL.bit.CHSEL pg(1445,1449,1467)

Def: selects the ADC channel
EE533: Configure ADCINA3, ADCINA5,ADCINA6 to convert on SOC0, SOC1 and SOC2 respectively

```
AdcaRegs.ADCSOC0CTL.bit.CHSEL = 3; //SOC0 will convert ADCINA3
AdcaRegs.ADCSOC1CTL.bit.CHSEL = 5; //SOC0 will convert ADCINA5
AdcaRegs.ADCSOC2CTL.bit.CHSEL = 6; //SOC0 will convert ADCINA6
```



With the information tabulated, it is easy to generate the SOC configurations:

```
AdcaRegs.ADCSOC0CTL.bit.CHSEL = 5;        //SOC0 will convert ADCINA5
AdcaRegs.ADCSOC0CTL.bit.ACQPS = 23;       //SOC0 will use sample duration of 24 SYSCLK cycles
AdcaRegs.ADCSOC0CTL.bit.TRIGSEL = 3;      //SOC0 will begin conversion on CPU1 Timer 2
AdcaRegs.ADCSOC1CTL.bit.CHSEL = 0;        //SOC1 will convert ADCINA0
AdcaRegs.ADCSOC1CTL.bit.ACQPS = 88;       //SOC1 will use sample duration of 89 SYSCLK cycles
AdcaRegs.ADCSOC1CTL.bit.TRIGSEL = 3;      //SOC1 will begin conversion on CPU1 Timer 2
AdcaRegs.ADCSOC2CTL.bit.CHSEL = 3;        //SOC2 will convert ADCINA3
AdcaRegs.ADCSOC2CTL.bit.ACQPS = 21;       //SOC2 will use sample duration of 22 SYSCLK cycles
AdcaRegs.ADCSOC2CTL.bit.TRIGSEL = 3;      //SOC2 will begin conversion on CPU1 Timer 2
AdcaRegs.ADCSOC3CTL.bit.CHSEL = 2;        //SOC3 will convert ADCINA2
AdcaRegs.ADCSOC3CTL.bit.ACQPS = 58;       //SOC3 will use sample duration of 59 SYSCLK cycles
AdcaRegs.ADCSOC3CTL.bit.TRIGSEL = 3;      //SOC3 will begin conversion on CPU1 Timer 2
```

As configured, when CPU1 Timer 2 generates an event, SOC0, SOC1, SOC2, and SOC3 will eventually be sampled and converted, in that order. The conversion results for ACINA5 (Signal 1) will be in ADCRESULT0. Similarly, The results for ADCINA0 (Signal 2), ADCINA3 (Signal 3), and ADCINA2 (Signal 4) will be in ADCRESULT1, ADCRESULT2, and ADCRESULT3, respectively.

Note

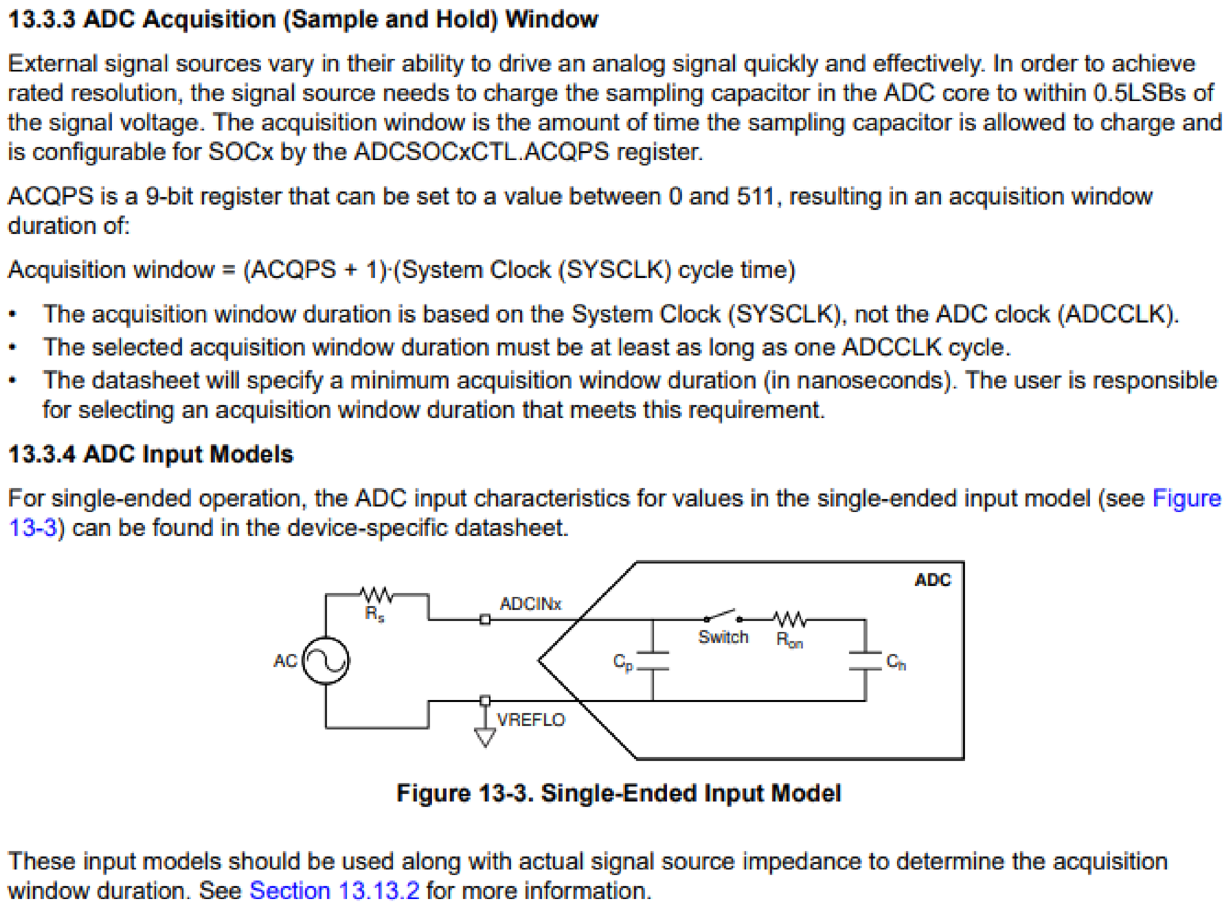
It is possible, but unlikely, that the ADC could begin converting SOC1, SOC2, or SOC3 before SOC0 depending on the position of the round-robin pointer when the CPU Timer trigger is received. See Section 13.5 to understand how the next SOC to be converted is chosen.

AdcaRegs.ADCSOCxCTL.bit.ACQPS pg(1446,1448)

Def: Selects ADC sample window based on number of cycles of ADC clock
EE533: 100ns sample window

```
Acquisition window = (ACQPS + 1)·(System Clock (SYSCLK) cycle time)
100E-9 = (ACQPS + 1)·(100MHz)
ACQPS = 9

AdcaRegs.ADCSOC0CTL.bit.ACQPS = 9;
AdcaRegs.ADCSOC1CTL.bit.ACQPS = 9;
AdcaRegs.ADCSOC2CTL.bit.ACQPS = 9;
```



13.3.3 ADC Acquisition (Sample and Hold) Window

External signal sources vary in their ability to drive an analog signal quickly and effectively. In order to achieve rated resolution, the signal source needs to charge the sampling capacitor in the ADC core to within 0.5LSBs of the signal voltage. The acquisition window is the amount of time the sampling capacitor is allowed to charge and is configurable for SOCx by the ADCSOCxCTL.ACQPS register.

ACQPS is a 9-bit register that can be set to a value between 0 and 511, resulting in an acquisition window duration of:

Acquisition window = (ACQPS + 1)·(System Clock (SYSCLK) cycle time)

- The acquisition window duration is based on the System Clock (SYSCLK), not the ADC clock (ADCCLK).
- The selected acquisition window duration must be at least as long as one ADCCLK cycle.
- The datasheet will specify a minimum acquisition window duration (in nanoseconds). The user is responsible for selecting an acquisition window duration that meets this requirement.

13.3.4 ADC Input Models

For single-ended operation, the ADC input characteristics for values in the single-ended input model (see Figure 13-3) can be found in the device-specific datasheet.

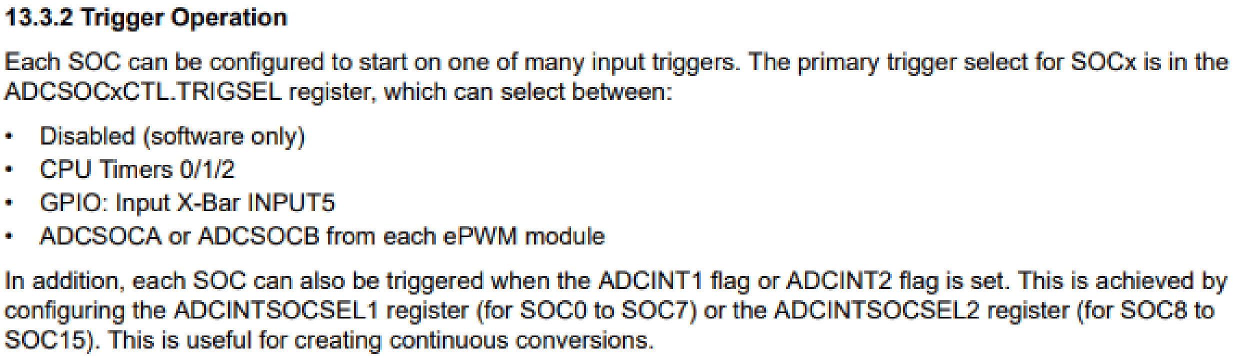
Figure 13-3. Single-Ended Input Model

These input models should be used along with actual signal source impedance to determine the acquisition window duration. See Section 13.13.2 for more information.

AdcaRegs.ADCSOCxCTL.bit.TRIGSEL pg(1446,1534)

Def: Selects trigger for ADC SOC
EE533: Trigger on EPWM1 SOCA

```
AdcaRegs.ADCSOC0CTL.bit.TRIGSEL = 0b05;
AdcaRegs.ADCSOC1CTL.bit.TRIGSEL = 0b05;
AdcaRegs.ADCSOC2CTL.bit.TRIGSEL = 0b05;
```



13.3.2 Trigger Operation

Each SOC can be configured to start on one of many input triggers. The primary trigger select for SOCx is in the ADCSOCxCTL.TRIGSEL register, which can select between:

- Disabled (software only)
- CPU Timers 0/1/2
- GPIO: Input X-Bar INPUT5
- ADCSOCA or ADCSOCB from each ePWM module

In addition, each SOC can also be triggered when the ADCINT1 flag or ADCINT2 flag is set. This is achieved by configuring the ADCINTSOCSEL1 register (for SOC0 to SOC7) or the ADCINTSOCSEL2 register (for SOC8 to SOC15). This is useful for creating continuous conversions.

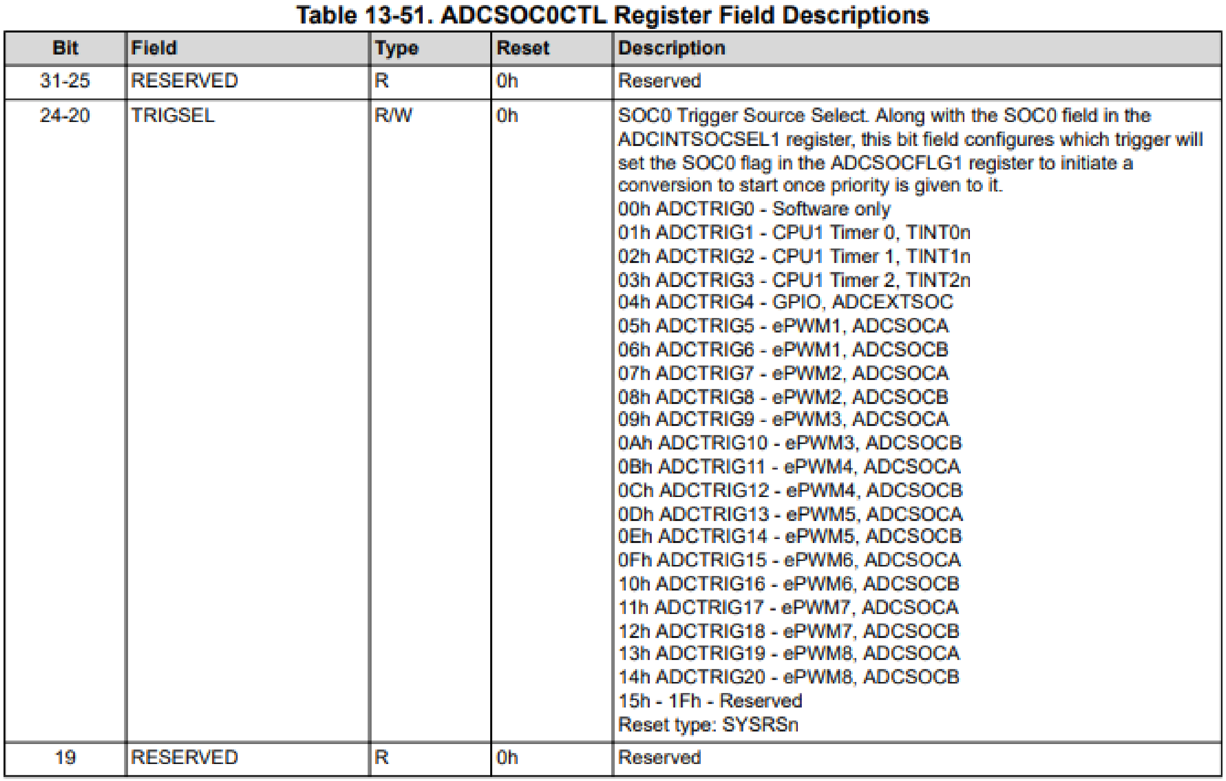


Table 13-51. ADCSOC0CTL Register Field Descriptions

Bit	Field	Type	Reset	Description
31-25	RESERVED	R	0h	Reserved
24-20	TRIGSEL	R/W	0h	SOC0 Trigger Source Select. Along with the SOC0 field in the ADCINTSOCSEL1 register, this bit field configures which trigger will set the SOC0 flag in the ADCSOCFLG1 register to initiate a conversion to start once priority is given to it. 00h ADCTRIG0 - Software only 01h ADCTRIG1 - CPU1 Timer 0, TINT0n 02h ADCTRIG2 - CPU1 Timer 1, TINT1n 03h ADCTRIG3 - CPU1 Timer 2, TINT2n 04h ADCTRIG4 - GPIO, ADCEXTSOC 05h ADCTRIG5 - ePWM1, ADCSOCA 06h ADCTRIG6 - ePWM1, ADCSOCB 07h ADCTRIG7 - ePWM2, ADCSOCA 08h ADCTRIG8 - ePWM2, ADCSOCB 09h ADCTRIG9 - ePWM3, ADCSOCA 0Ah ADCTRIG10 - ePWM3, ADCSOCB 0Bh ADCTRIG11 - ePWM4, ADCSOCA 0Ch ADCTRIG12 - ePWM4, ADCSOCB 0Dh ADCTRIG13 - ePWM5, ADCSOCA 0Eh ADCTRIG14 - ePWM5, ADCSOCB 0Fh ADCTRIG15 - ePWM6, ADCSOCA 10h ADCTRIG16 - ePWM6, ADCSOCB 11h ADCTRIG17 - ePWM7, ADCSOCA 12h ADCTRIG18 - ePWM7, ADCSOCB 13h ADCTRIG19 - ePWM8, ADCSOCA 14h ADCTRIG20 - ePWM8, ADCSOCB 15h - 1Fh - Reserved Reset type: SYSRSn
19	RESERVED	R	0h	Reserved