第七讲 模数转换模块ADC

主讲：叶剑
电话：13728639620
Email：yejian@hit.edu.cn

第七讲 模数转换模块ADC

1、A/D转换基本原理

2、F28335的ADC转换模块

3、ADC单元寄存器

A/D转换基本原理

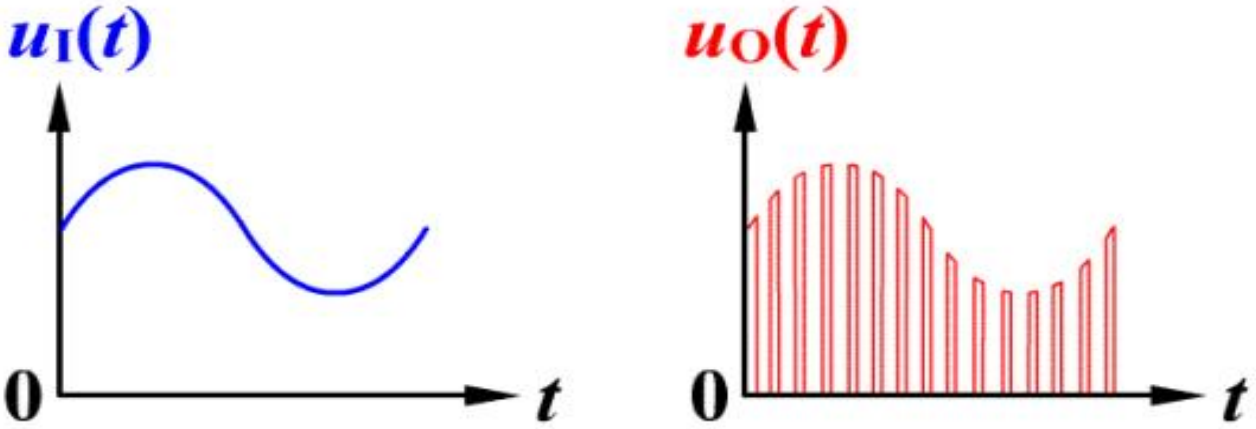
“模拟信号无处不在”。

在数字世界里，模拟信号只能转换为数字信号来处理。

ADC转换步骤：采样、保持、量化、编码 。

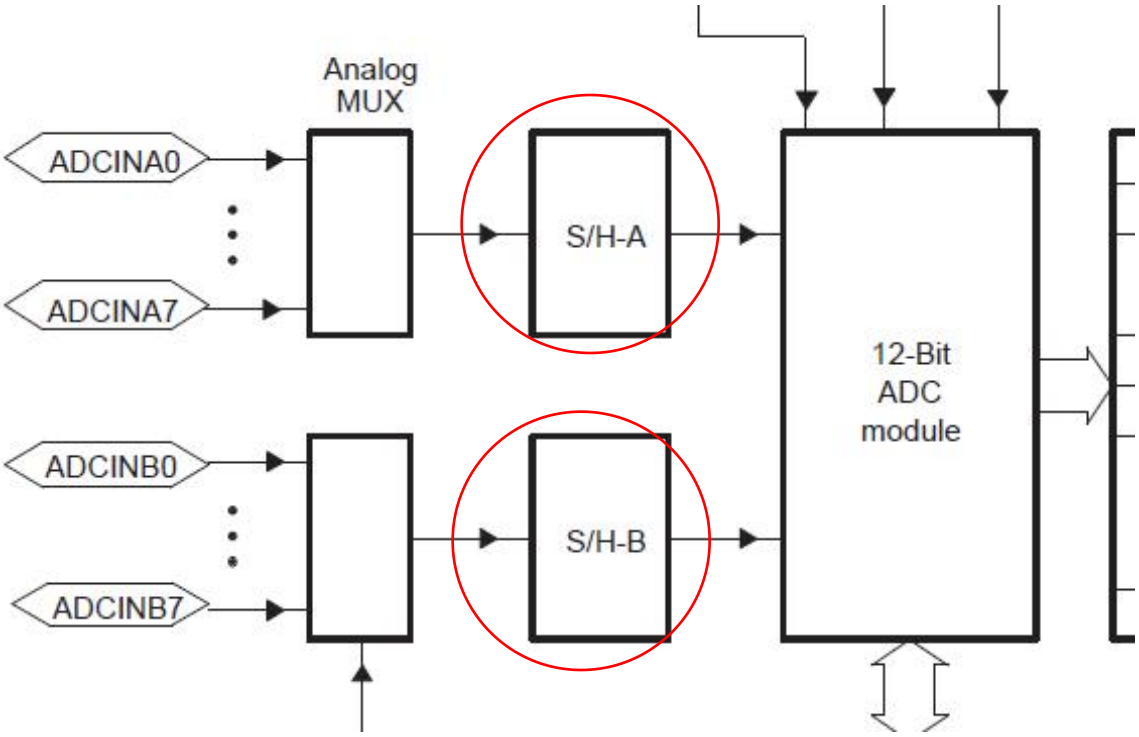
A/D转换基本原理

所谓采样，就是将一个时间上连续变化的模拟量转化为时间上离散变化的模拟量。



A/D转换基本原理

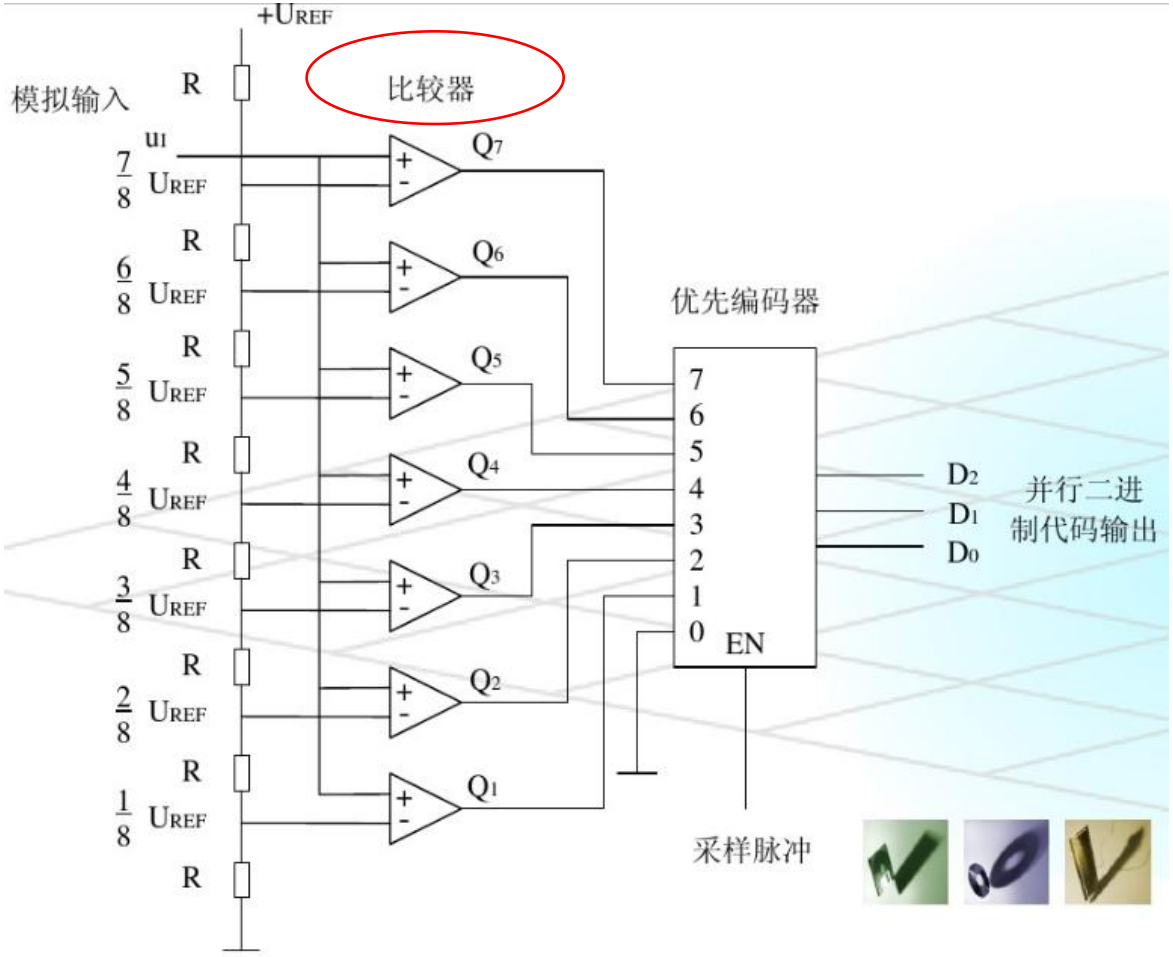
将采样结果存储起来，直到下次采样，这个过程称作保持。一般，采样器和保持电路一起总称为采样保持电路。



A/D转换基本原理

将采样电平转化为最小单位电平整数倍称作量化。

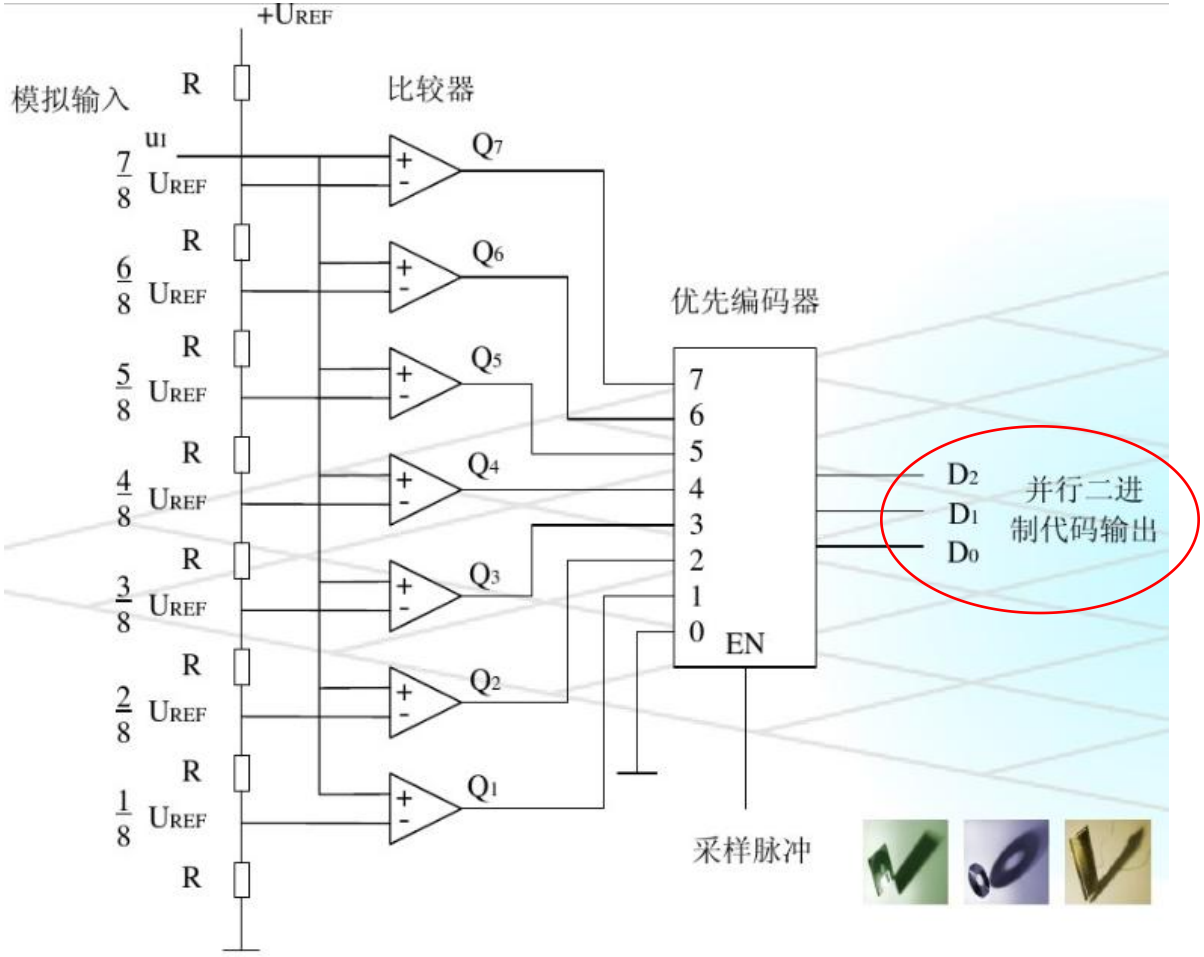
输入模拟电压u_I
$0 \le u_I < \frac{1}{8}U_{REF}$
$\frac{1}{8}U_{REF} \le u_I < \frac{2}{8}U_{REF}$
$\frac{2}{8}U_{REF} \le u_I < \frac{3}{8}U_{REF}$
$\frac{3}{8}U_{REF} \le u_I < \frac{4}{8}U_{REF}$
$\frac{4}{8}U_{REF} \le u_I < \frac{5}{8}U_{REF}$
$\frac{5}{8}U_{REF} \le u_I < \frac{6}{8}U_{REF}$
$\frac{6}{8}U_{REF} \le u_I < \frac{7}{8}U_{REF}$
$\frac{7}{8}U_{REF} \le u_I < U_{REF}$



A/D转换基本原理

将量化后的结果按照一定数制形式表示就是编码。

输入模拟电压u_I
$0 \le u_I < \frac{1}{8}U_{REF}$
$\frac{1}{8}U_{REF} \le u_I < \frac{2}{8}U_{REF}$
$\frac{2}{8}U_{REF} \le u_I < \frac{3}{8}U_{REF}$
$\frac{3}{8}U_{REF} \le u_I < \frac{4}{8}U_{REF}$
$\frac{4}{8}U_{REF} \le u_I < \frac{5}{8}U_{REF}$
$\frac{5}{8}U_{REF} \le u_I < \frac{6}{8}U_{REF}$
$\frac{6}{8}U_{REF} \le u_I < \frac{7}{8}U_{REF}$
$\frac{7}{8}U_{REF} \le u_I < U_{REF}$



A/D转换基本原理

F28335的模拟电压转换值

- Sixteen result registers (individually addressable) to store conversion values
  - The digital value of the input analog voltage is derived by:

$$\text{Digital Value} = 0, \quad \text{when input} \le 0\ \text{V}$$

$$\text{Digital Value} = 4096 \times \frac{\text{Input Analog Voltage} - \text{ADCLO}}{3}, \quad \text{when } 0\ \text{V} < \text{input} < 3\ \text{V}$$

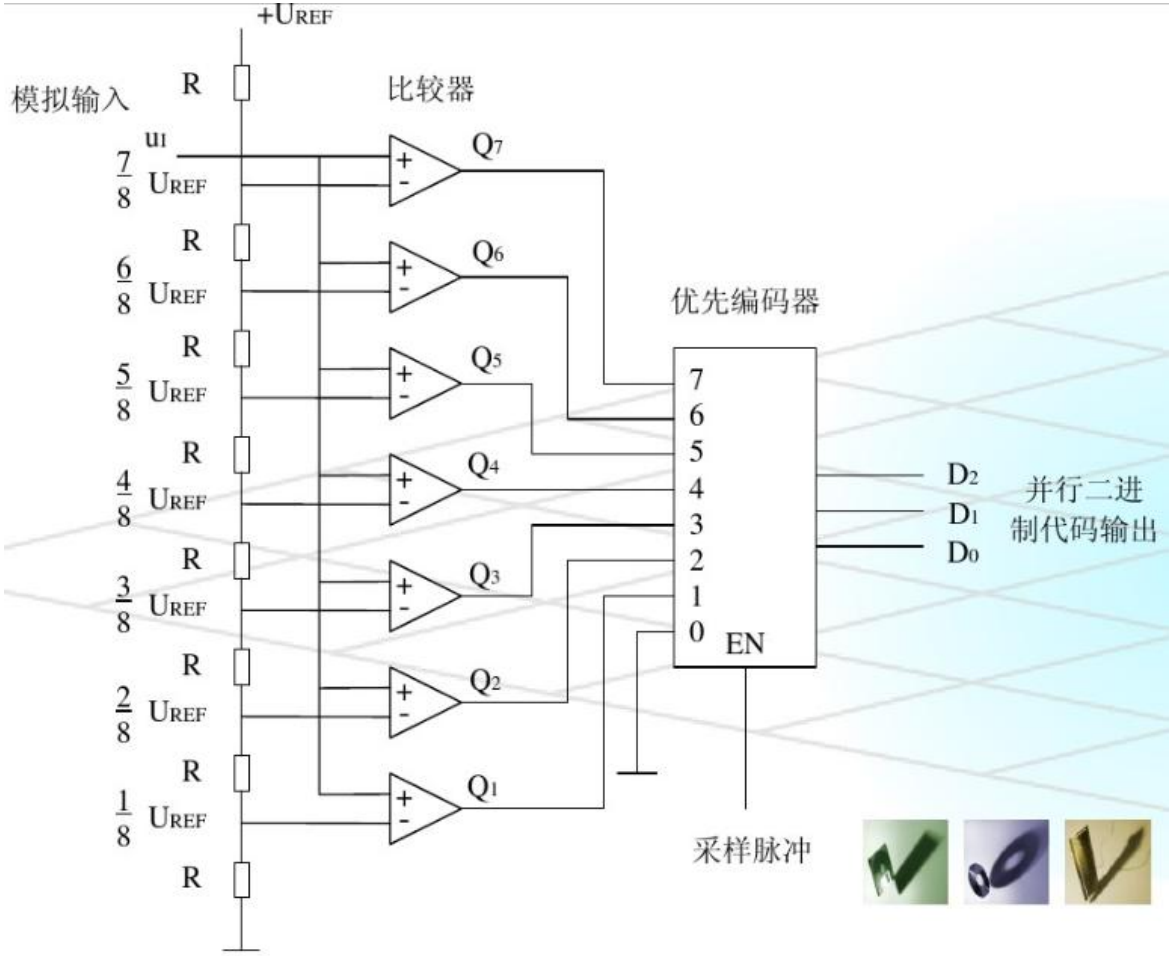
$$\text{Digital Value} = 4095, \quad \text{when input} \ge 3\ \text{V}$$

A All fractional values are truncated.

A/D转换基本原理

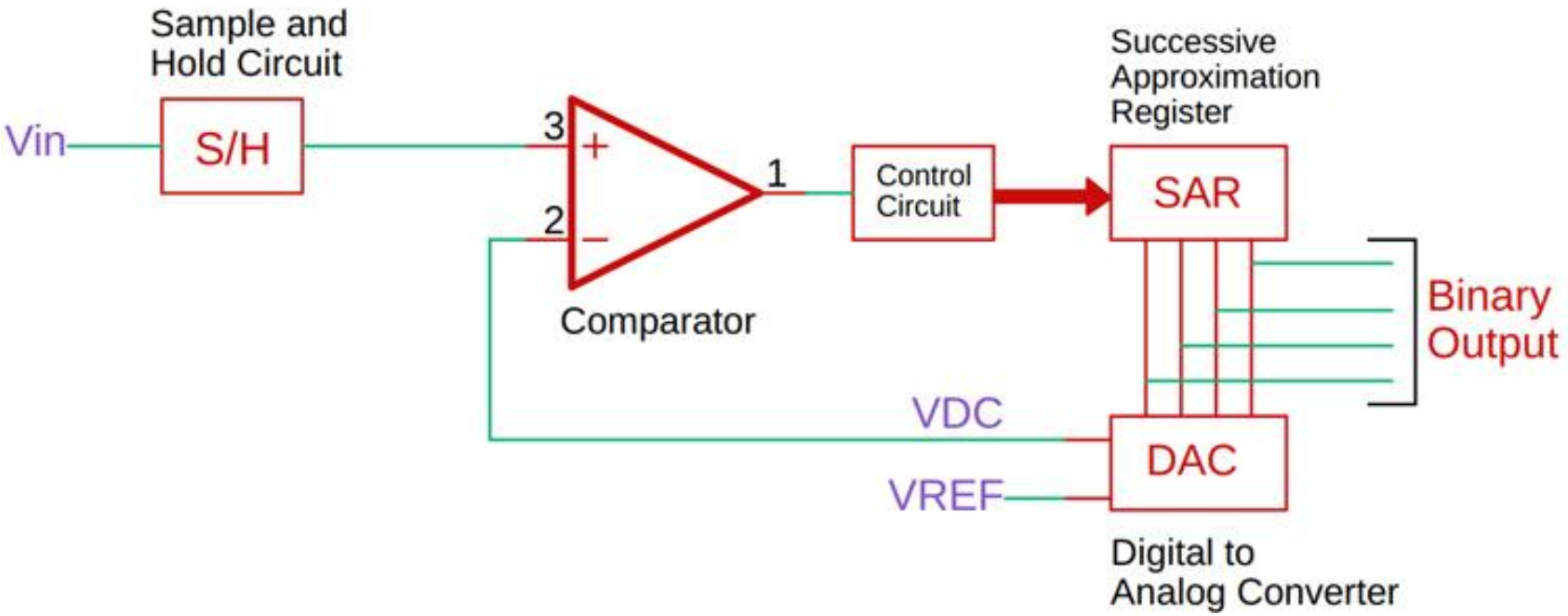
F28335逐步逼近型ADC

输入模拟电压u_I
$0 \le u_I < \frac{1}{8}U_{REF}$
$\frac{1}{8}U_{REF} \le u_I < \frac{2}{8}U_{REF}$
$\frac{2}{8}U_{REF} \le u_I < \frac{3}{8}U_{REF}$
$\frac{3}{8}U_{REF} \le u_I < \frac{4}{8}U_{REF}$
$\frac{4}{8}U_{REF} \le u_I < \frac{5}{8}U_{REF}$
$\frac{5}{8}U_{REF} \le u_I < \frac{6}{8}U_{REF}$
$\frac{6}{8}U_{REF} \le u_I < \frac{7}{8}U_{REF}$
$\frac{7}{8}U_{REF} \le u_I < U_{REF}$



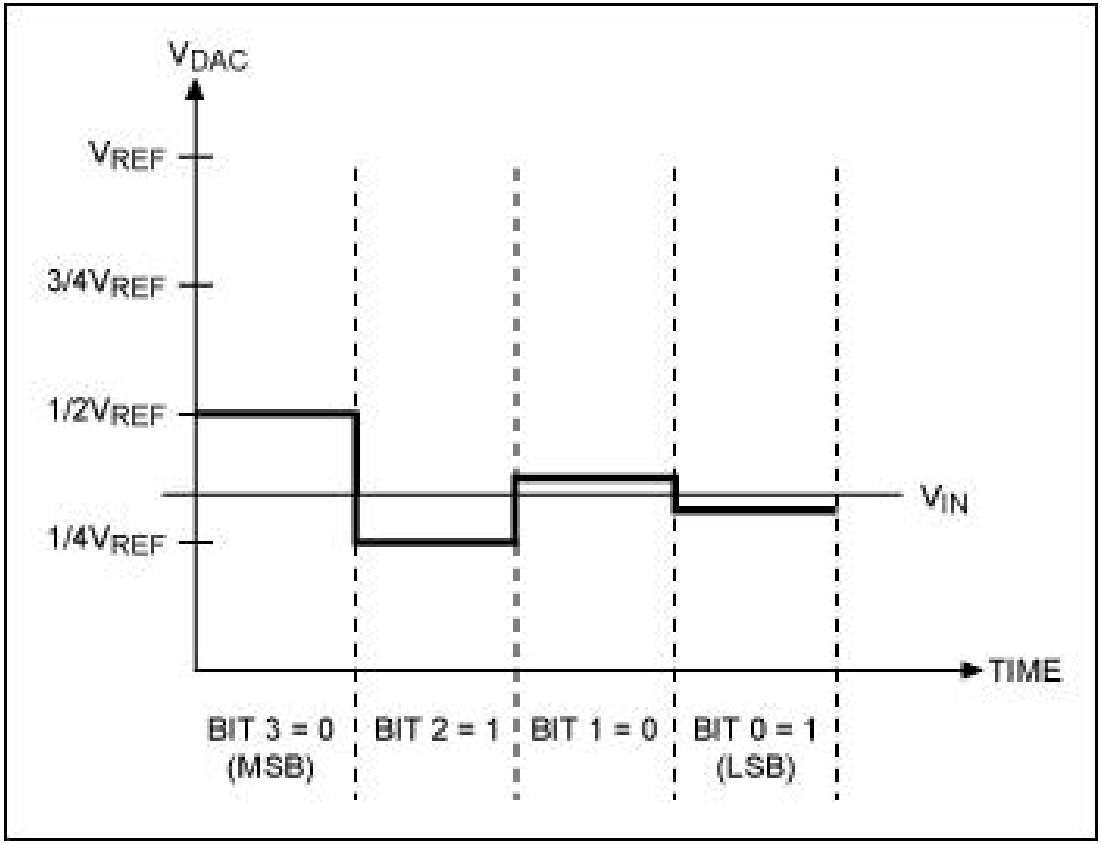
A/D转换基本原理

F28335逐步逼近型ADC



A/D转换基本原理

F28335逐步逼近型ADC



第七讲 ADC转换单元

1、A/D转换基本原理

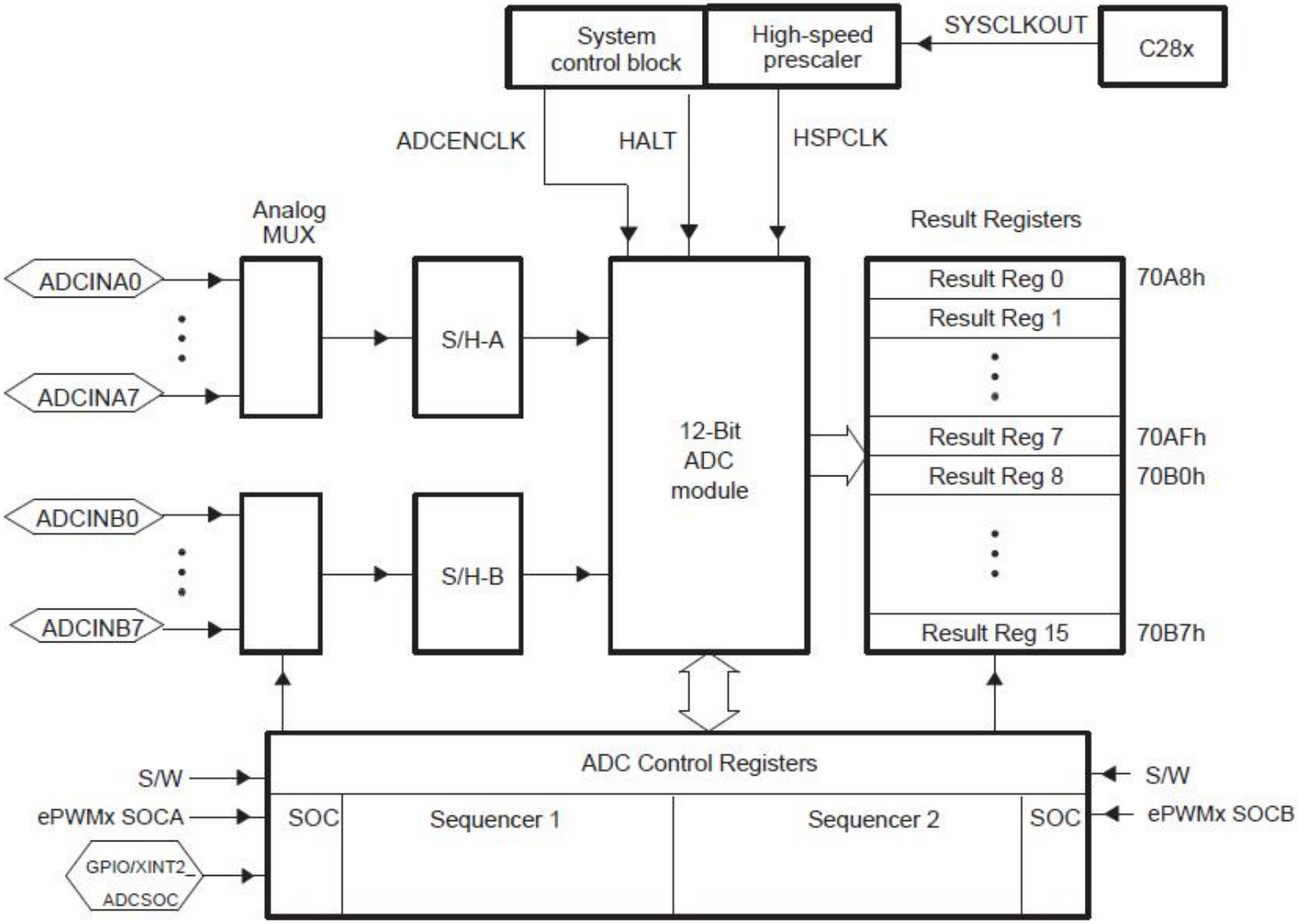
2、F28335的ADC转换模块

3、ADC单元寄存器

2.1 F28335的ADC模块特点

1. 具有双采样保持器(S/H)的12位模数转换内核；
2. 同时采样模式或顺序采样模式；
3. 模拟电压输入范围0～3V；
4. 快速采样功能，转换时钟12.5MHz，采样速度6.25MSPS；
5. 16通道模拟输入；
6. 自动定序功能，在一个采样序列内支持16次“自动转换”，每次转换可选16个通道任意一个；
7. 序列发生器可以配置成两个独立8通道或一个16通道；
8. 16个结果寄存器存放ADC转换的结果；
9. 多个触发源启动ADC转换(SOC)；
10. 灵活的中断控制，允许中断请求出现在每个转换序列结尾（EOS）；
11. 排序器可工作在启动/停止模式；
12. 采样保持(S/H)采集时间窗口有独立的预定标控制；

2.1 F28335的ADC模块特点



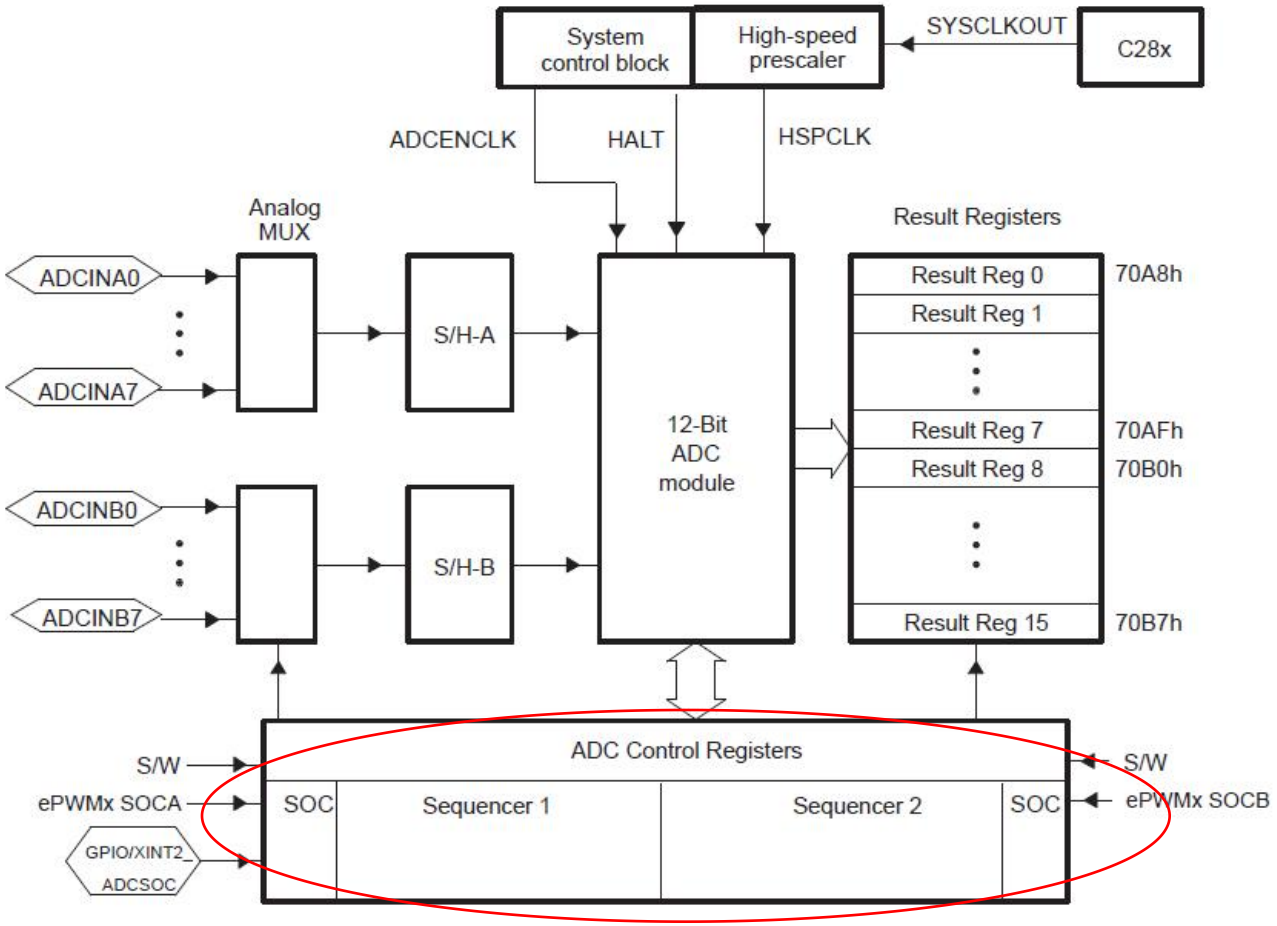
2.2 转换序列发生器工作原理

1. 序列发生器可以工作在级联模式和双序列模式。

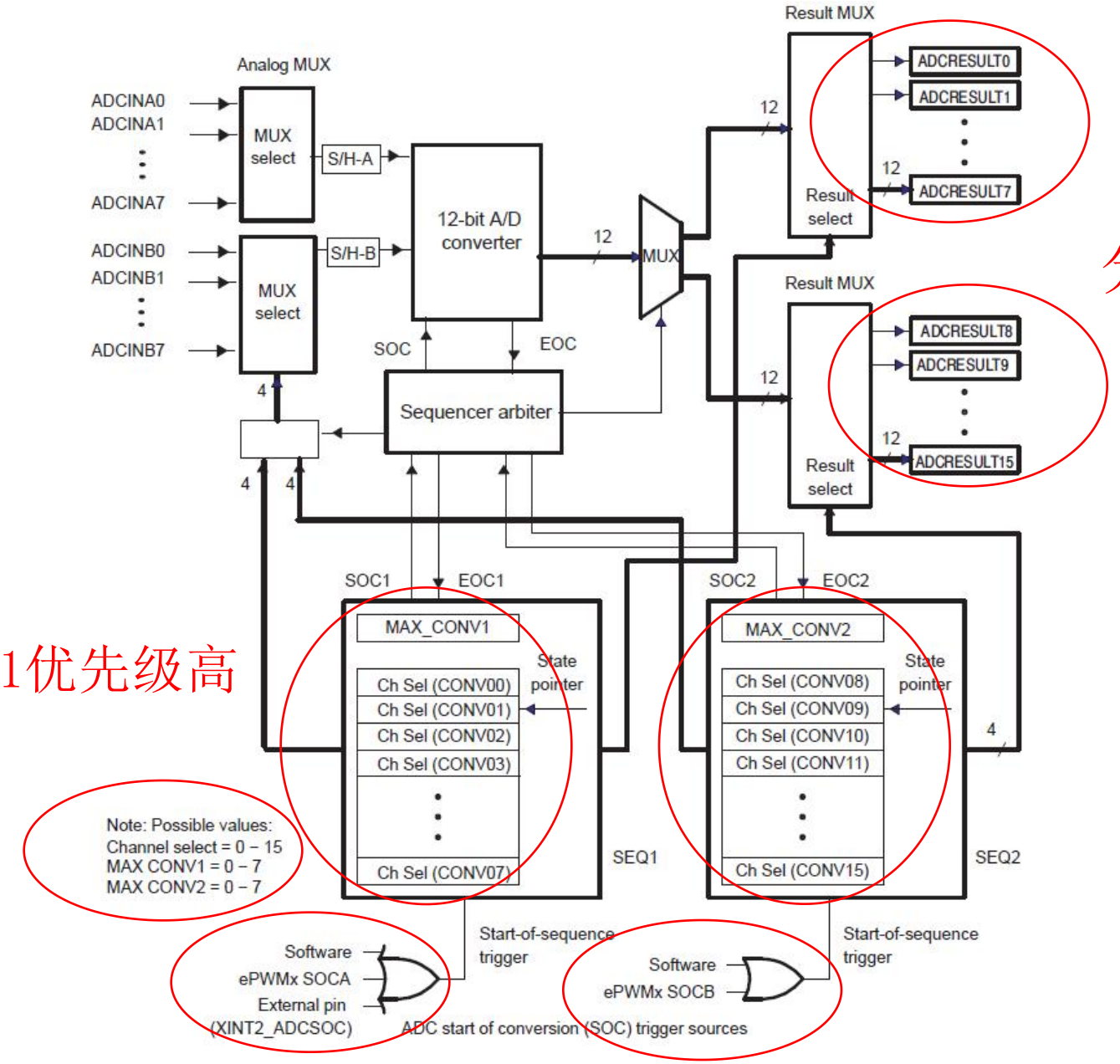
2. 序列发生器可以工作在顺序采样模式和同时采样模式。

3. 序列发生器可以工作在连续运行模式和启动/停止模式。

1. 序列发生器可以工作在级联模式和双序列模式。



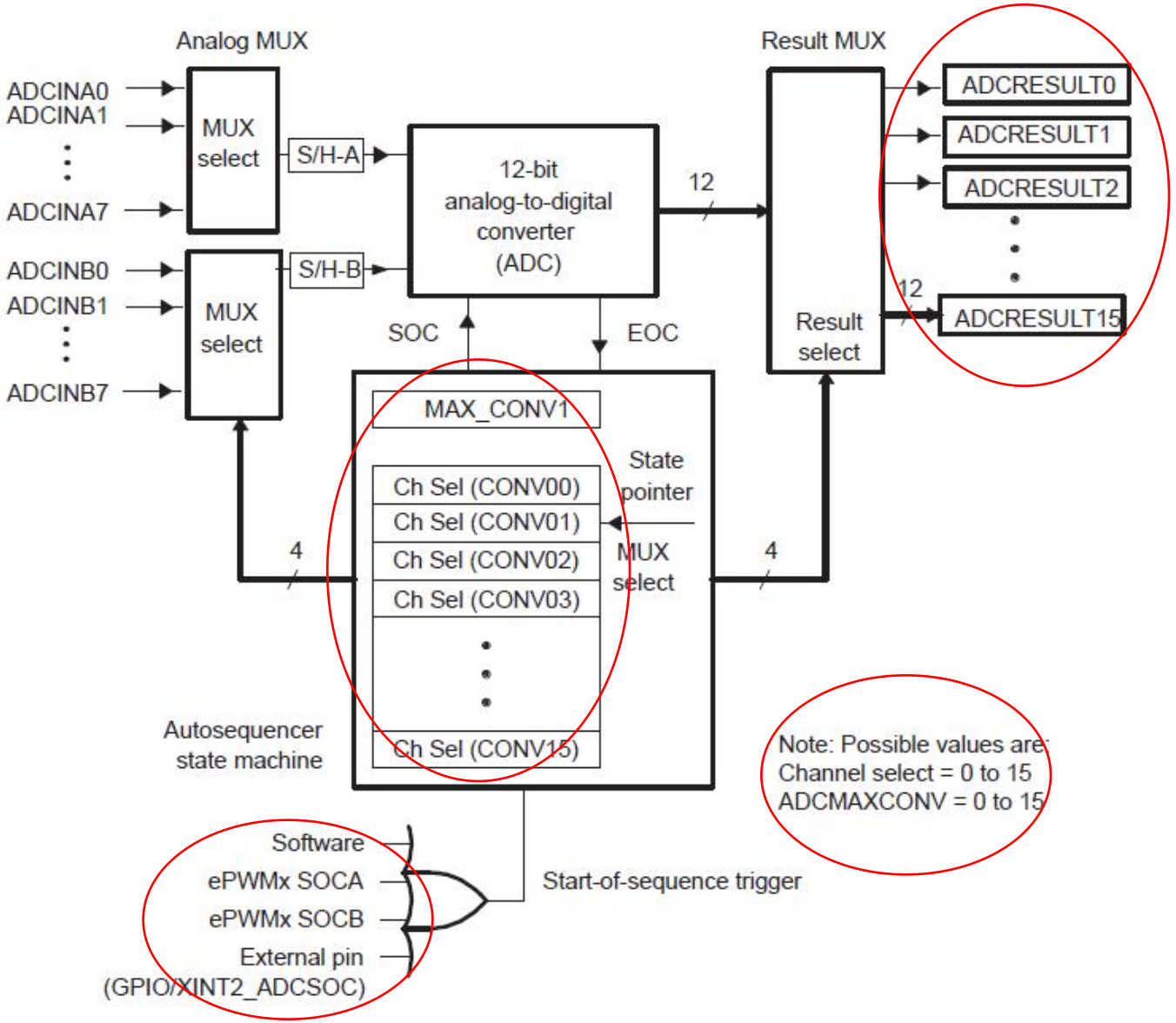
双序列模式：两个独立8通道SEQ1和SEQ2。



分开

SEQ1优先级高

级联模式：一个独立16通道SEQ。

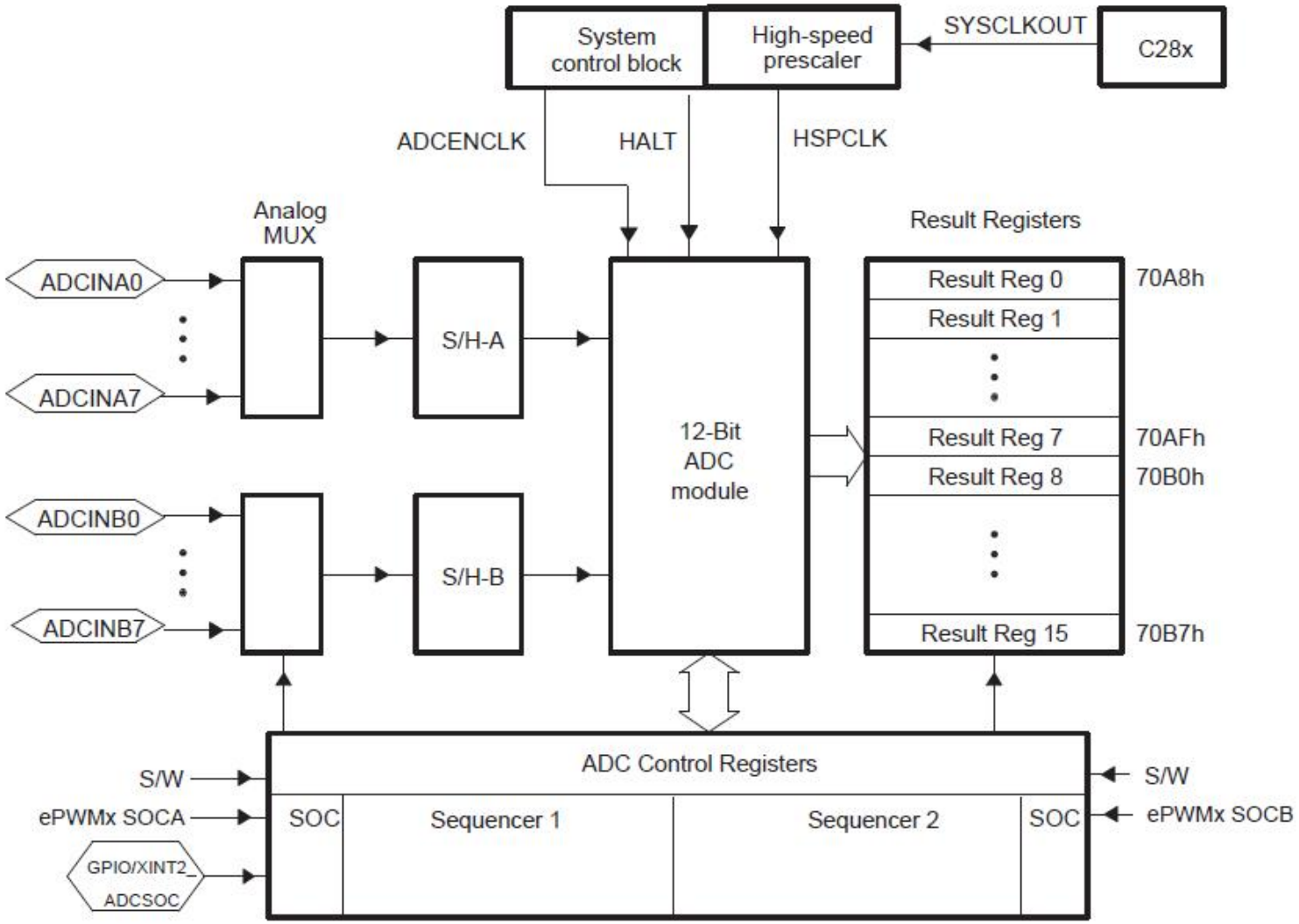


不分开

级联模式 VS 双序列模式

Feature	Single 8-state sequencer #1 (SEQ1)	Single 8-state sequencer #2 (SEQ2)	Cascaded 16-state sequencer (SEQ)
Start-of-conversion (SOC) triggers	ePWMx SOCA, software, external pin	ePWMx SOCB, software	ePWMx SOCA, ePWMx SOCB, software, external pin
Maximum number of autoconversions (i.e., sequence length)	8	8	16
Autostop at end-of-sequence (EOS)	Yes	Yes	Yes
Arbitration priority	High	Low	Not applicable
ADC conversion result register locations	0 to 7	8 to 15	0 to 15
ADCCHSELSEQn bit field assignment	CONV00 to CONV07	CONV08 to CONV15	CONV00 to CONV15

2. 序列发生器可以工作在顺序采样模式和同时采样模式。



顺序采样模式：ADCMAXCONV决定对多少个通道采样；

Maximum Conversion Channels Register (ADCMAXCONV) Field Descriptions

15							8
Reserved							
R-0							

7	6		4	3			0
Reserved	MAX_CONV2			MAX_CONV1			
R-0	R/W-0			R/W-0			

LEGEND: R/W = Read/Write; R = Read only; x = undefined, -*n* = value after reset

6-0	MAX_CONVn	MAX_CONVn bit field defines the maximum number of conversions executed in an autoconversion session. The bit fields and their operation vary according to the sequencer modes (dual/cascaded). For SEQ1 operation, bits MAX_CONV1[2:0] are used. For SEQ2 operation, bits MAX_CONV2[2:0] are used. For SEQ operation, bits MAX_CONV1[3:0] are used. An autoconversion session always starts with the initial state and continues sequentially until the end state if allowed. The result buffer is filled in a sequential order. Any number of conversions between 1 and (MAX_CONVn +1) can be programmed for a session.

顺序采样模式：CONVnn位段决定要采样和转换的通道；

CONVnn Value	ADC Input Channel Selected
0000	ADCINA0
0001	ADCINA1
0010	ADCINA2
0011	ADCINA3
0100	ADCINA4
0101	ADCINA5
0110	ADCINA6
0111	ADCINA7
1000	ADCINB0
1001	ADCINB1
1010	ADCINB2
1011	ADCINB3
1100	ADCINB4
1101	ADCINB5
1110	ADCINB6
1111	ADCINB7

顺序采样模式：ADCCHSELSEQn决定采样顺序。

ADCCHSELSEQ1，ADCCHSELSEQ2，ADCCHSELSEQ3，ADCCHSELSEQ4，每个寄存器都是16位的，顺序的4位决定一个输入通道，转换顺序是从ADCCHSELSEQ1最低4位到ADCCHSELSEQ4的最高4位，最多16个。

顺序采样模式：ADCCHSELSEQn决定采样顺序。

Figure 2-9. ADC Input Channel Select Sequencing Control Registers (ADCCHSELSEQ1) (Address Offset 03h)

15 12	11 8	7 4	3 0
CONV03	CONV02	CONV01	CONV00
R/W-0	R/W-0	R/W-0	R/W-0

LEGEND: R/W = Read/Write; -*n* = value after reset

Figure 2-10. ADC Input Channel Select Sequencing Control Registers (ADCCHSELSEQ2) (Address Offset 04h)

15 12	11 8	7 4	3 0
CONV07	CONV06	CONV05	CONV04
R/W-0	R/W-0	R/W-0	R/W-0

LEGEND: R/W = Read/Write; -*n* = value after reset

Figure 2-11. ADC Input Channel Select Sequencing Control Registers (ADCCHSELSEQ3) (Address Offset 05h)

15 12	11 8	7 4	3 0
CONV11	CONV10	CONV09	CONV08
R/W-0	R/W-0	R/W-0	R/W-0

LEGEND: R/W = Read/Write; -*n* = value after reset

Figure 2-12. ADC Input Channel Select Sequencing Control Registers (ADCCHSELSEQ4) (Address Offset 06h)

15 12	11 8	7 4	3 0
CONV15	CONV14	CONV13	CONV12
R/W-0	R/W-0	R/W-0	R/W-0

LEGEND: R/W = Read/Write; -*n* = value after reset

顺序采样模式：ADCCHSELSEQn决定采样顺序。

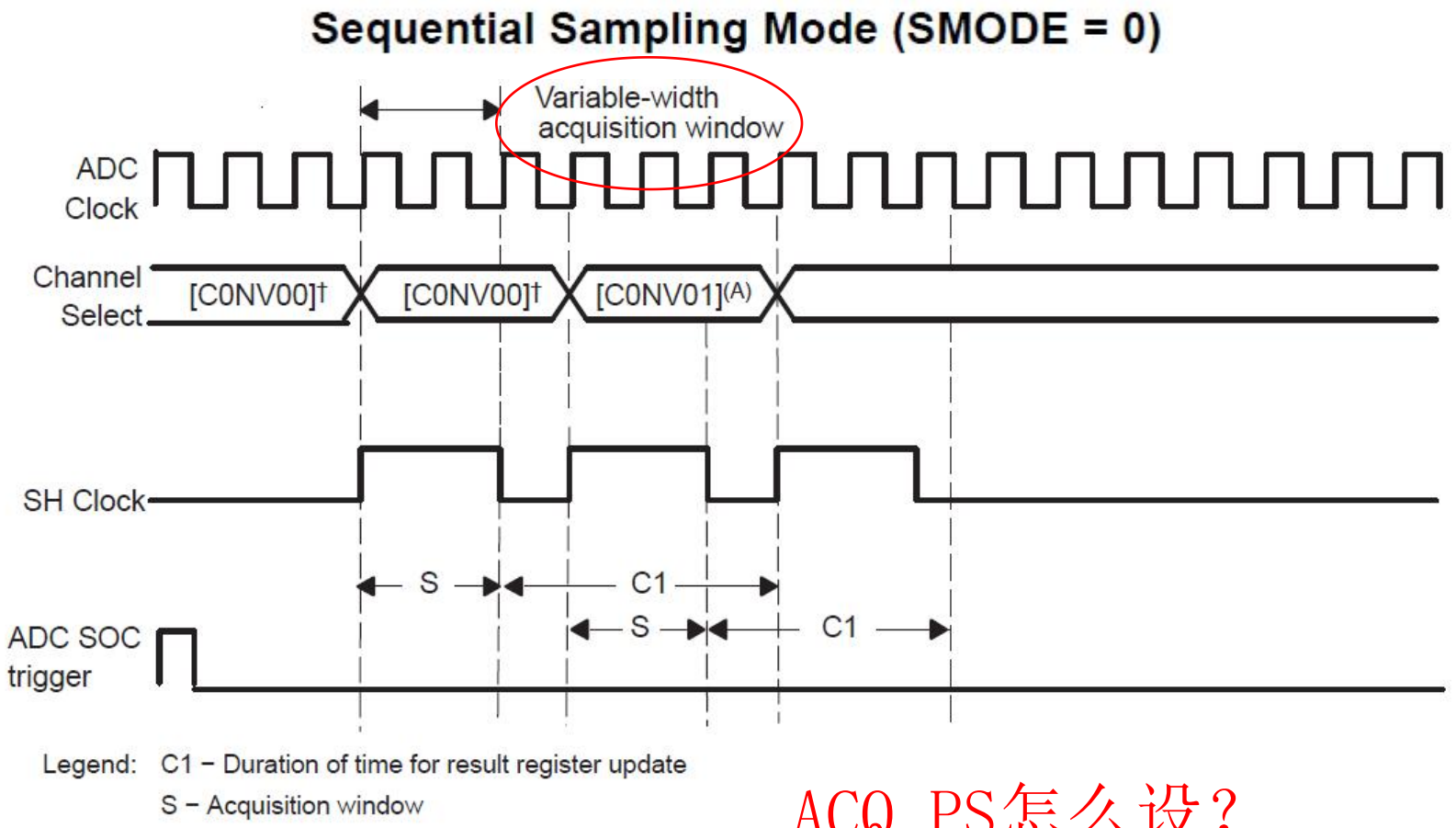
例：双序列发生器模式，用SEQ1进行转换，ADCINA2和ADCINA3各两次，ADCINA6、ADCINA7和ADCINB4各一次。

Values for ADCCHSELSEQn Registers (MAX_CONV1 Set to 6)

	Bits 15-12[1]	Bits 11-8[1]	Bits 7-4[1]	Bits 3-0[1]	
70A3h	3	2	3	2	ADCCHSELSEQ1
70A4h	x	12	7	6	ADCCHSELSEQ2
70A5h	x	x	x	x	ADCCHSELSEQ3
70A6h	x	x	x	x	ADCCHSELSEQ4

(1) Values are in decimal, and x = don't care

顺序采样模式：ACQ_PS决定采样窗口宽度



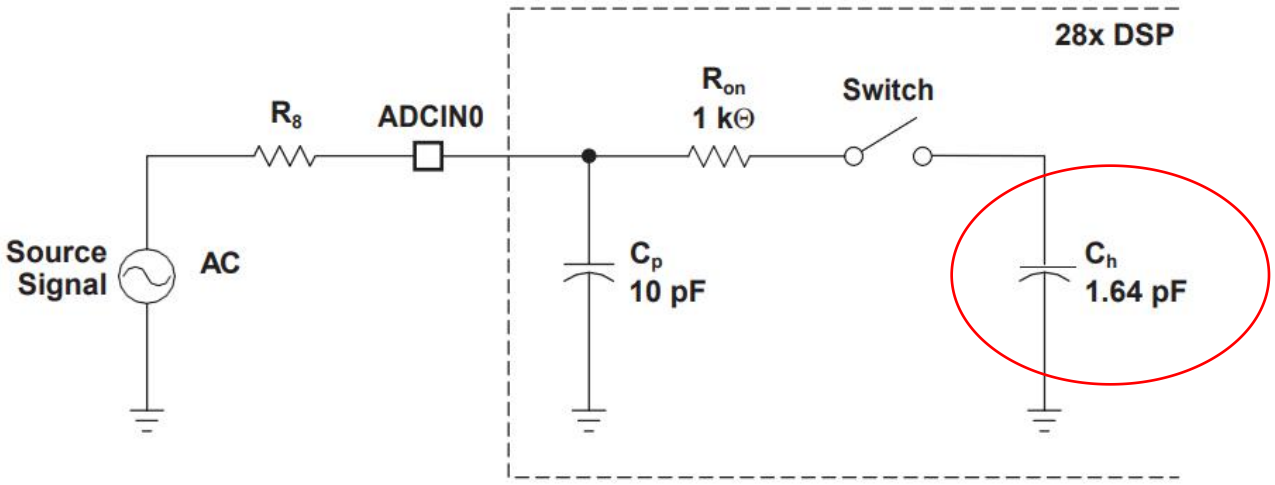
ACQ_PS怎么设？

11-8	ACQ_PS[3:0]		Acquisition window size. This bit field controls the width of SOC pulse, which, in turn, determines for what time duration the sampling switch is closed. The width of SOC pulse is ADCTRL1[11:8] + 1 times the ADCLK period.

顺序采样模式：ACQ_PS决定采样窗口宽度

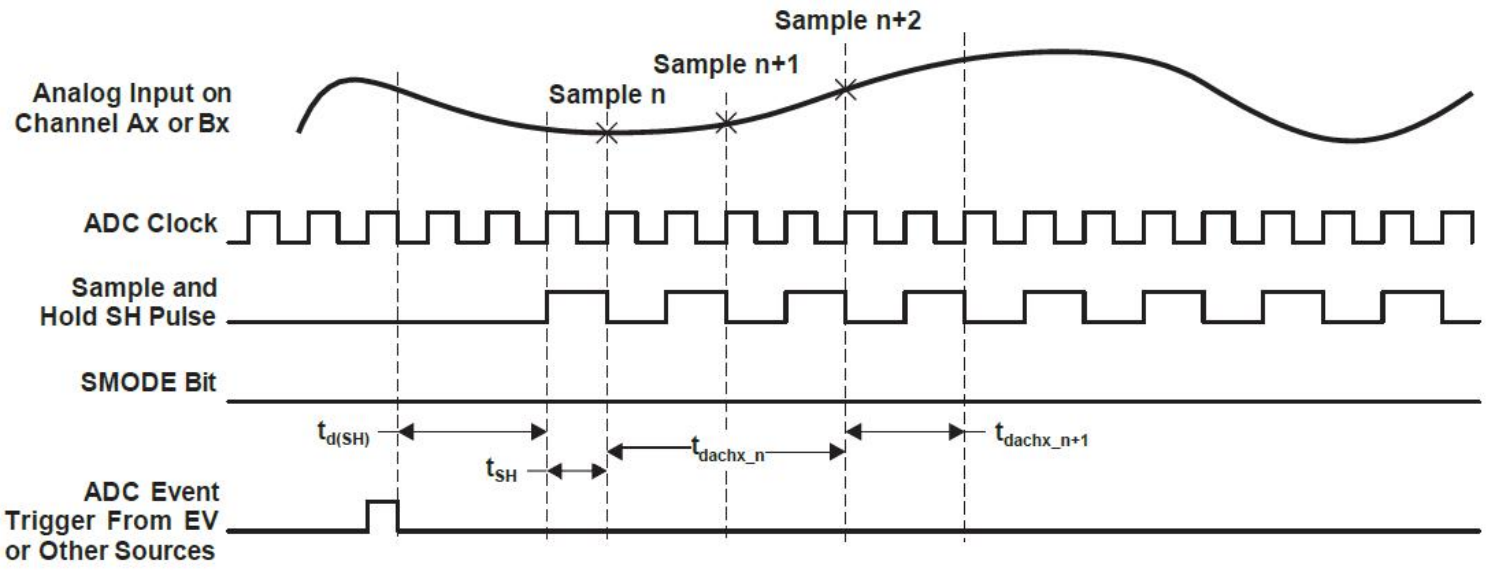
The higher the source impedance, the higher the ACQ_PS (sample time) value number should be set. The goal is to charge the S/H capacitor to the voltage equal to the VIN value.

- 模拟输入阻抗模型



顺序采样模式：采样窗口时序

- First S/H pulse is active after 2.5tc(ADCCLK), from the -ve edge of the SOC trigger.
- S/H period is (1 + Acqps) * tc(ADCCLK)
- For the sequential sampling mode:
  - The first result appears at (1 + Acqps) * tc(ADCCLK), from the -ve edge of the S/H pulse
  - Successive results appear at every (2 + Acqps) * tc(ADCCLK)



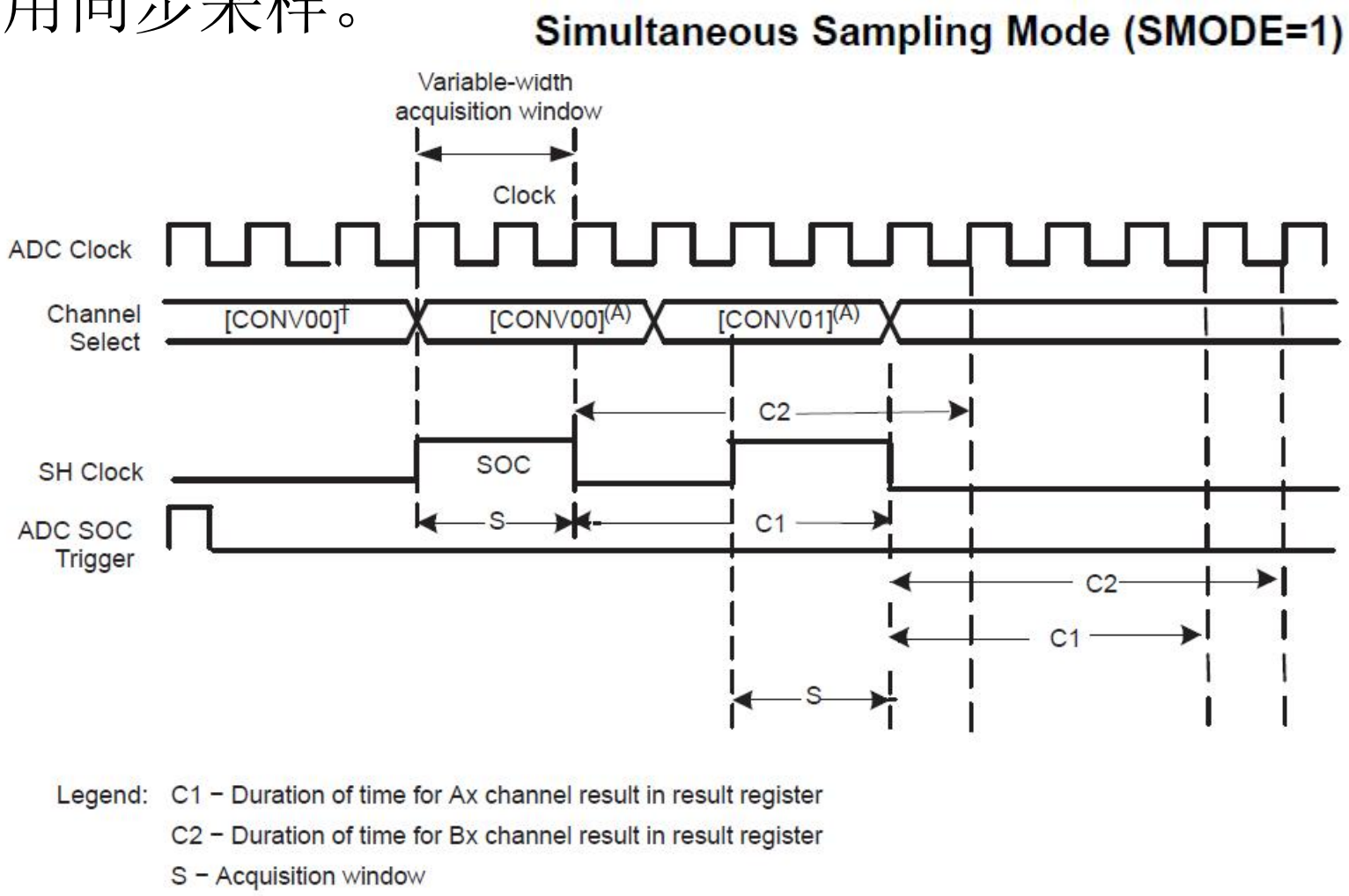
同时采样模式：可以两路信号同时采样保持，先后转换。当你有相位一致或者同步采样的要求时，建议用同步采样。

CONVnn最高位被忽略，由低三位决定S/H-A和S/H-B的输入通道。

同时采样模式： CONVnn最高位被忽略，由低三位决定S/H-A和S/H-B的输入通道。

CONVnn Value	ADC Input Channel Selected
0000	ADCINA0
0001	ADCINA1
0010	ADCINA2
0011	ADCINA3
0100	ADCINA4
0101	ADCINA5
0110	ADCINA6
0111	ADCINA7
1000	ADCINB0
1001	ADCINB1
1010	ADCINB2
1011	ADCINB3
1100	ADCINB4
1101	ADCINB5
1110	ADCINB6
1111	ADCINB7

同时采样模式：可以两路信号同时采样保持，先后转换。当你有相位一致或者同步采样的要求时，建议用同步采样。



同时采样模式：采样窗口时序。

- First S/H pulse is active after 2.5tc(ADCCLK), from the -ve edge of the SOC trigger.
- S/H period is (1 + Acqps) * tc(ADCCLK)
- For the simultaneous sampling mode:
  - The first result for the A0 channel appears at (4 + Acqps) * tc(ADCCLK), from the -ve edge of the S/H pulse
  - The first result for the B0 channel appears at (5 + Acqps) * tc(ADCCLK), from the -ve edge of the S/H pulse
  - Successive results for Ax and Bx channels appear at every (3 + Acqps) * tc(ADCCLK)

Simultaneous Sampling Dual Sequencer Mode Example

Example initialization:

```
AdcRegs.ADCTRL3.bit.SMODE_SEL =   0x1;    // Setup simultaneous sampling mode
AdcRegs.ADCMAXCONV.all =       0x0033;    // 4 double conv's each sequencer (8 total)
AdcRegs.ADCCHSELSEQ1.bit.CONV00 = 0x0;    // Setup conv from ADCINA0 &amp; ADCINB0
AdcRegs.ADCCHSELSEQ1.bit.CONV01 = 0x1;    // Setup conv from ADCINA1 &amp; ADCINB1
AdcRegs.ADCCHSELSEQ1.bit.CONV02 = 0x2;    // Setup conv from ADCINA2 &amp; ADCINB2
AdcRegs.ADCCHSELSEQ1.bit.CONV03 = 0x3;    // Setup conv from ADCINA3 &amp; ADCINB3
AdcRegs.ADCCHSELSEQ3.bit.CONV08 = 0x4;    // Setup conv from ADCINA4 &amp; ADCINB4
AdcRegs.ADCCHSELSEQ3.bit.CONV09 = 0x5;    // Setup conv from ADCINA5 &amp; ADCINB5
AdcRegs.ADCCHSELSEQ3.bit.CONV10 = 0x6;    // Setup conv from ADCINA6 &amp; ADCINB6
AdcRegs.ADCCHSELSEQ3.bit.CONV11 = 0x7;    // Setup conv from ADCINA7 &amp; ADCINB7
```

If SEQ1 and SEQ2 were both executed, the results would go to the following RESULT registers:

```
ADCINA0 -> ADCRESULT0
ADCINB0 -> ADCRESULT1
ADCINA1 -> ADCRESULT2
ADCINB1 -> ADCRESULT3
ADCINA2 -> ADCRESULT4
ADCINB2 -> ADCRESULT5
ADCINA3 -> ADCRESULT6
ADCINB3 -> ADCRESULT7
ADCINA4 -> ADCRESULT8
ADCINB4 -> ADCRESULT9
ADCINA5 -> ADCRESULT10
ADCINB5 -> ADCRESULT11
ADCINA6 -> ADCRESULT12
ADCINB6 -> ADCRESULT13
ADCINA7 -> ADCRESULT14
ADCINB7 -> ADCRESULT15
```

Simultaneous Sampling Cascaded Sequencer Mode Example

```
AdcRegs.ADCTRL3.bit.SMODE_SEL    =    0x1;     // Setup simultaneous sampling mode
AdcRegs.ADCTRL1.bit.SEQ_CASC     =    0x1;     // Setup cascaded sequencer mode
AdcRegs.ADCMAXCONV.all =        0x0007;     // 8 double conv's (16 total)
AdcRegs.ADCCHSELSEQ1.bit.CONV00     = 0x0;        // Setup conv from ADCINA0 &amp; ADCINB0
AdcRegs.ADCCHSELSEQ1.bit.CONV01     = 0x1;        // Setup conv from ADCINA1 &amp; ADCINB1
AdcRegs.ADCCHSELSEQ1.bit.CONV02     = 0x2;        // Setup conv from ADCINA2 &amp; ADCINB2
AdcRegs.ADCCHSELSEQ1.bit.CONV03     = 0x3;        // Setup conv from ADCINA3 &amp; ADCINB3
AdcRegs.ADCCHSELSEQ2.bit.CONV04     = 0x4;        // Setup conv from ADCINA4 &amp; ADCINB4
AdcRegs.ADCCHSELSEQ2.bit.CONV05     = 0x5;        // Setup conv from ADCINA5 &amp; ADCINB5
AdcRegs.ADCCHSELSEQ2.bit.CONV06     = 0x6;        // Setup conv from ADCINA6 &amp; ADCINB6
AdcRegs.ADCCHSELSEQ2.bit.CONV07     = 0x7;        // Setup conv from ADCINA7 &amp; ADCINB7
```

If the cascaded SEQ was executed, the results would go to the following ADCRESULT registers:

```
ADCINA0 -> ADCRESULT0
ADCINB0 -> ADCRESULT1
ADCINA1 -> ADCRESULT2
ADCINB1 -> ADCRESULT3
ADCINA2 -> ADCRESULT4
ADCINB2 -> ADCRESULT5
ADCINA3 -> ADCRESULT6
ADCINB3 -> ADCRESULT7
ADCINA4 -> ADCRESULT8
ADCINB4 -> ADCRESULT9
ADCINA5 -> ADCRESULT10
ADCINB5 -> ADCRESULT11
ADCINA6 -> ADCRESULT12
ADCINB6 -> ADCRESULT13
ADCINA7 -> ADCRESULT14
ADCINB7 -> ADCRESULT15
```

3. 序列发生器可以工作在连续运行模式和启动/停止模式。

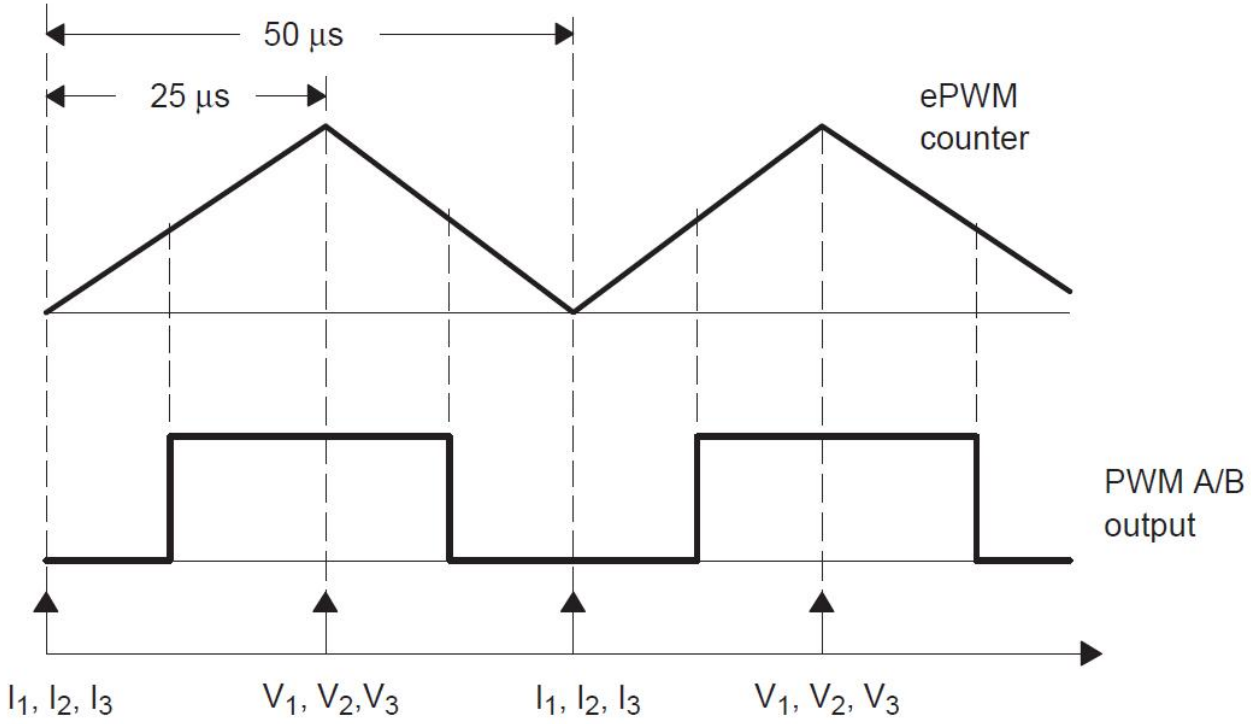
6	CONT_RUN	用来决定序列发生器工作在连续模式还是启动/停止模式。即使当前转换正在进行，也可以对此位进行写操作，在当前转换结束后，此位开始作用。 0：启动/停止模式，序列发生器在接收到 EOS 后停止，在下次 SOC 到来时，序列发生器从上次结束时的状态开始启动，除非期间对序列发生器进行复位； 1：连续运行模式，在接收到 EOS 信号后，序列发生器接下来的动作取决于 SEQ_OVRD 位。如果 SEQ_OVRD=0，那么序列发生器复位到其初始状态（SEQ1 和 SEQ 回到 CONV00，SEQ2 回到 CONV08），如果 SEQ_OVRD=1，序列发生器不复位，并从其当前位置继续运行

启动/停止模式：序列发生器不复位，接收多个SOC信号。

6	CONT_RUN	用来决定序列发生器工作在连续模式还是启动/停止模式。即使当前转换正在进行，也可以对此位进行写操作，在当前转换结束后，此位开始作用。 0：启动/停止模式，序列发生器在接收到 EOS 后停止，在下次 SOC 到来时，序列发生器从上次结束时的状态开始启动，除非期间对序列发生器进行复位； 1：连续运行模式，在接收到 EOS 信号后，序列发生器接下来的动作取决于 SEQ_OVRD 位。如果 SEQ_OVRD=0，那么序列发生器复位到其初始状态（SEQ1 和 SEQ 回到 CONV00，SEQ2 回到 CONV08），如果 SEQ_OVRD=1，序列发生器不复位，并从其当前位置继续运行

启动/停止模式：序列发生器不复位，接收多个SOC信号。

	Bits 15-12	Bits 11-8	Bits 7-4	Bits 3-0	
70A3h	V_1	I_3	I_2	I_1	ADCCHSELSEQ1
70A4h	x	x	V_3	V_2	ADCCHSELSEQ2
70A5h	x	x	x	x	ADCCHSELSEQ3
70A6h	x	x	x	x	ADCCHSELSEQ4

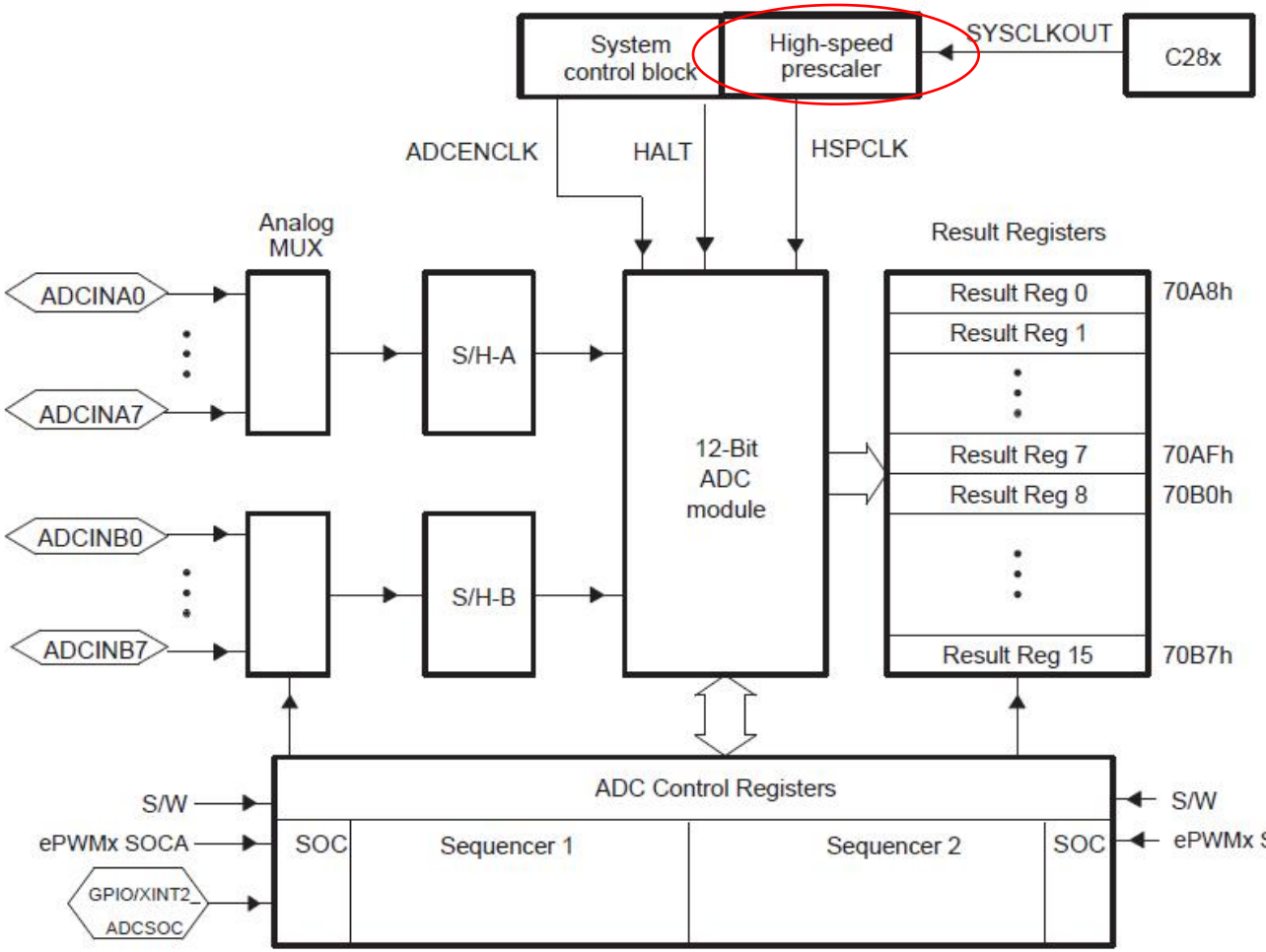


连续运行模式：接收1个SOC信号。

6	CONT_RUN	用来决定序列发生器工作在连续模式还是启动/停止模式。即使当前转换正在进行，也可以对此位进行写操作，在当前转换结束后，此位开始作用。 0：启动/停止模式，序列发生器在接收到 EOS 后停止，在下次 SOC 到来时，序列发生器从上次结束时的状态开始启动，除非期间对序列发生器进行复位； 1：连续运行模式，在接收到 EOS 信号后，序列发生器接下来的动作取决于 SEQ_OVRD 位。如果 SEQ_OVRD=0，那么序列发生器复位到其初始状态（SEQ1 和 SEQ 回到 CONV00，SEQ2 回到 CONV08），如果 SEQ_OVRD=1，序列发生器不复位，并从其当前位置继续运行
5	SEQ_OVRD	序列发生器覆盖功能。 0：允许序列发生器在完成 MAX_CONVn 个转换后回绕； 1：在序列发生器完成 MAX_CONVn 个转换后发生覆盖，只有在序列发生器中的末端发生回绕

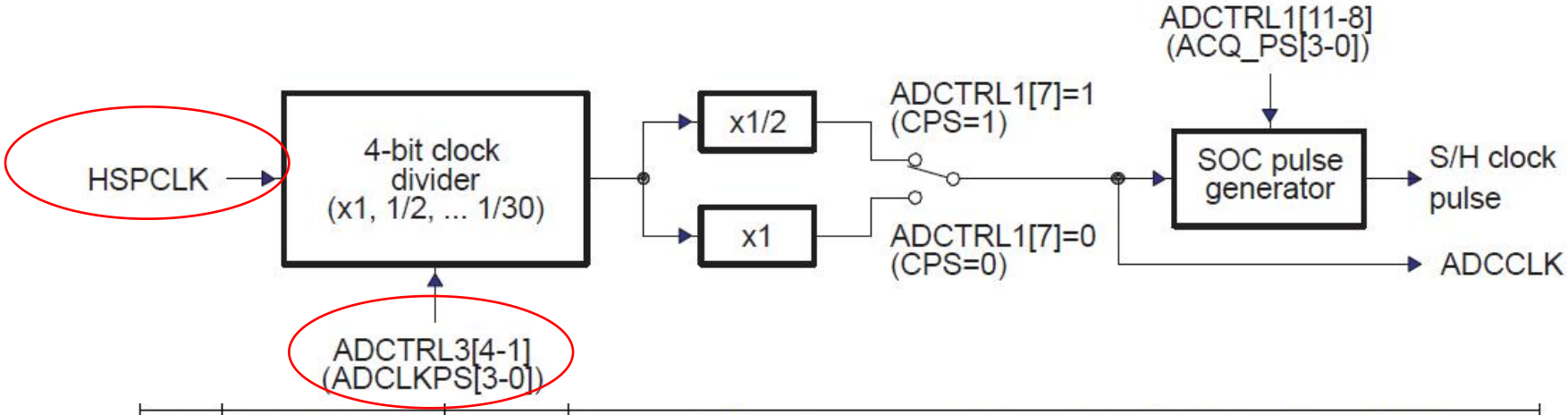
2.3 ADC的转换时钟

1.外设时钟HSPCLK由系统时钟SYSCLKOUT分频而来。



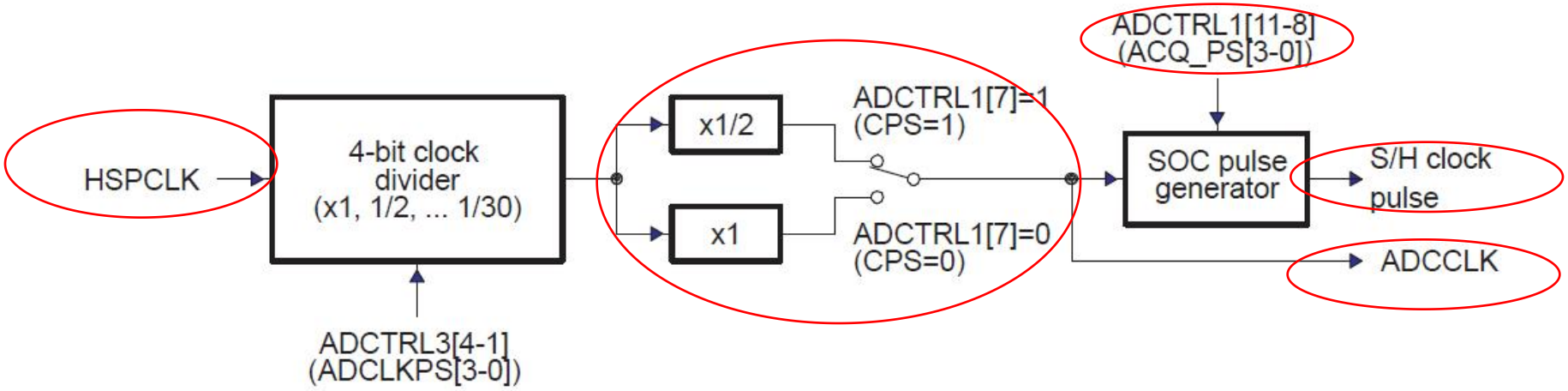
HSPCLK=SYSCLKOUT/2*HISPCP

2. ADC的转换时钟ADCCLK由外设时钟HSPCLK分频而来。



4-1	ADCCLKPS [3:0]		Core clock divider. 28x peripheral clock, HSPCLK, is divided by 2*ADCCLKPS[3-0], except when ADCCLKPS[3-0] is 0000, in which case HSPCLK is directly passed on. The divided clock is further divided by ADCTRL1[7]+1 to generate the core clock, ADCLK.
			ADCCLKPS [3:0] Core Clock Divider ADCLK
		0000	0 HSPCLK/(ADCTRL1[7] + 1)
		0001	1 HSPCLK/[2*(ADCTRL1[7] + 1)]
		0010	2 HSPCLK/[4*(ADCTRL1[7] + 1)]
		0011	3 HSPCLK/[6*(ADCTRL1[7] + 1)]
		0100	4 HSPCLK/[8*(ADCTRL1[7] + 1)]
		0101	5 HSPCLK/[10*(ADCTRL1[7] + 1)]
		0110	6 HSPCLK/[12*(ADCTRL1[7] + 1)]
		0111	7 HSPCLK/[14*(ADCTRL1[7] + 1)]
		1000	8 HSPCLK/[16*(ADCTRL1[7] + 1)]
		1001	9 HSPCLK/[18*(ADCTRL1[7] + 1)]
		1010	10 HSPCLK/[20*(ADCTRL1[7] + 1)]
		1011	11 HSPCLK/[22*(ADCTRL1[7] + 1)]
		1100	12 HSPCLK/[24*(ADCTRL1[7] + 1)]
		1101	13 HSPCLK/[26*(ADCTRL1[7] + 1)]
		1110	14 HSPCLK/[28*(ADCTRL1[7] + 1)]
		1111	15 HSPCLK/[30*(ADCTRL1[7] + 1)]

2. ADC的转换时钟ADCCLK由外设时钟HSPCLK分频而来。



7	CPS		Core clock prescaler. The prescaler is applied to divided device peripheral clock, HSPCLK.
		0	ADCCLK = F_{clk}/1
		1	ADCCLK = F_{clk}/2
			Note: F_{clk} = Prescaled HSPCLK (ADCCLKPS[3:0])

11-8	ACQ_PS[3:0]		Acquisition window size. This bit field controls the width of SOC pulse, which, in turn, determines for what time duration the sampling switch is closed. The width of SOC pulse is ADCTRL1[11:8] + 1 times the ADCLK period.

3. ADC的转换时钟例子。

XCLKIN	SYSCLKOUT	HSPCLK	ADCTRL3[4-1]	ADCTRL1[7]	ADCCLK	ADCTRL1[11-8]	SH Width
		HISPCP=3	ADCLKPS = 0	CPS=0		ACQ_PS = 0	
30 MHz	150 MHz	150 MHz/ 2 X 3 = 25 MHz	25 MHz	25 MHz	25 MHz	12.5 MHz 12.5 MSPS sustained conversion rate	1 ADC Clock 40 ns
		HISPCP=2	ADCLKPS = 2	CPS = 1		ACQ_PS = 15	
20 MHz	100 MHz	100 MHz/ 2 X 2 = 25 MHz	25/2 X 2 = 6.25 MHz	6.25 MHz/ 2 X 1 = 3.125 MHz	3.125 MHz	183.824 kHz 183.824 kSPS sustained conversion rate	16 ADC Clocks 5.12 µs

怎么得到的？

2.4 ADC的基本电气特性

1.低功耗模式

ADC支持3种不同的供电模式：ADC上电、ADC断电和ADC关闭。

Power Level	ADCBGRFDN1	ADCBGRFDN0	ADCPWDN
ADC power-up	1	1	1
ADC power-down	1	1	0
ADC off	0	0	0
Reserved	1	0	X
Reserved	0	1	X

2. 片内/片外参考电压选择

F28335处理器的ADC单元的参考电压有2种提供方式，即内部参考电压和外部参考电压，外部参考电压分别为2.048V、1.5V或1.024V。一般情况下，尽量选择内部参考源。

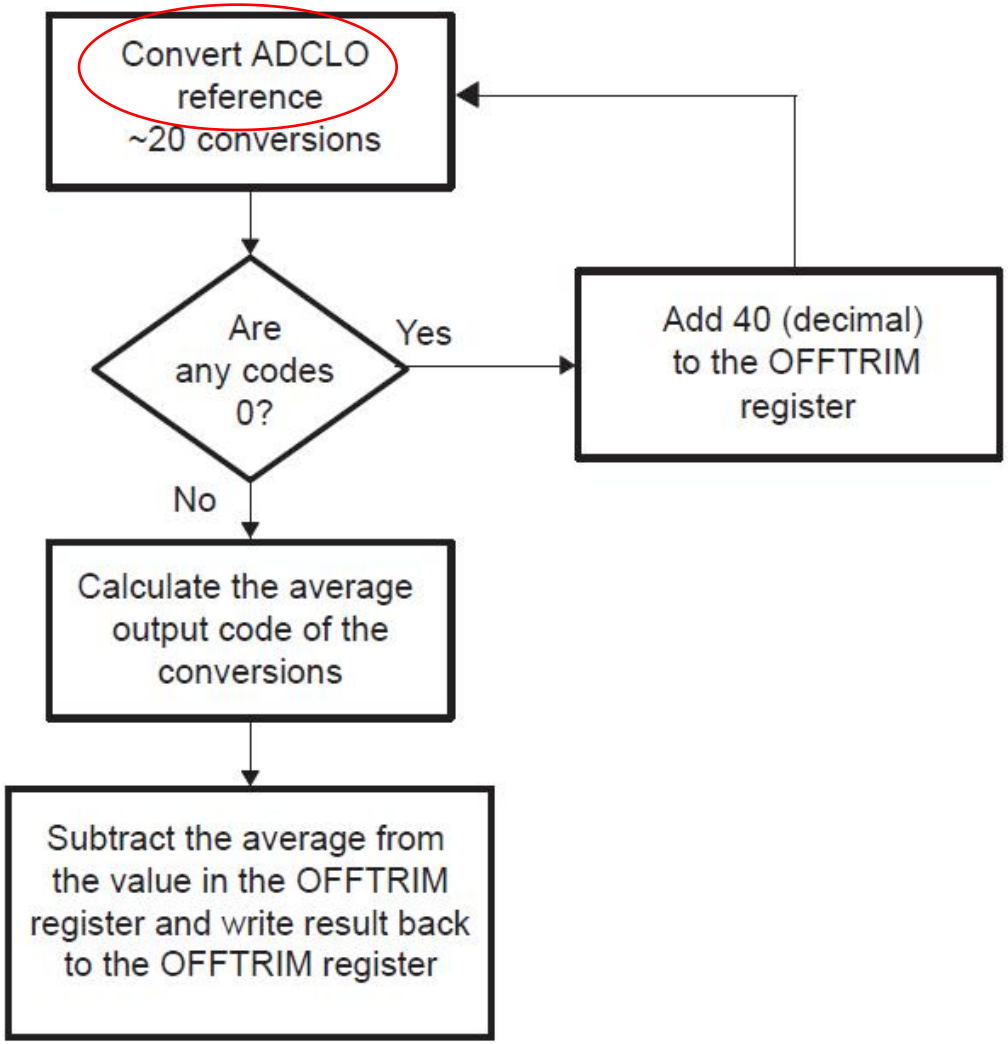
ADC Reference Select Register (ADCREFSEL) Field Descriptions

Bit(s)	Name	Value	Description
15-14	REF_SEL[1:0]		Reference select bits for ADC voltage generation circuit options are listed below:
		00	Internal reference selected (default)
		01	External reference, 2.048 V on ADCREFIN
		10	External reference, 1.500 V on ADCREFIN
		11	External reference, 1.024 V on ADCREFIN

2.5 ADC的输入校正

The 2833x ADC 通过ADC Offset Trim Register(ADCOFFTRIM)提供偏置校正。

ADCLO零电位



The 2833x ADC 通过ADC Offset Trim Register(ADCOFFTRIM)提供偏置校正。

At startup, most of the reference conversions yield a zero result. After writing the value 0x28 (40 decimal) into the OFFTRIM register, all of the reference conversions give a positive result and average out to 0x19 (25 decimal). The final value written to the OFFTRIM register should be 0x0F (15 decimal).

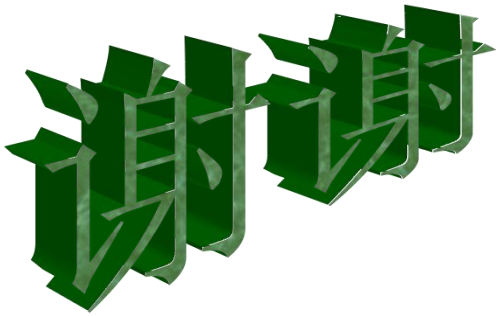
第七讲 ADC转换单元

1、A/D转换基本原理

2、F28335的ADC转换模块

3、ADC单元寄存器

已结合ADC模块介绍


谢谢