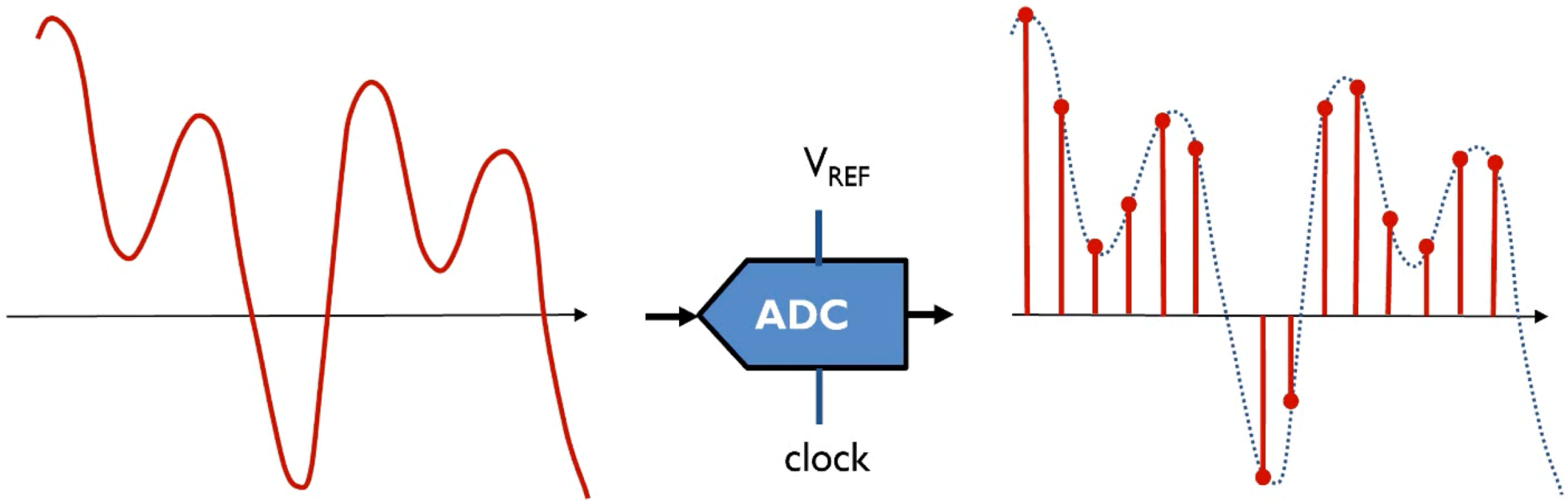
Analog-to-Digital Converter (ADC)

LAB – ADC - DMA

Zpracováno podle podkladů Dr. Yifeng Zhu

Analog-to-Digital Converter (ADC)



Analog signal $x(t)$

continuous time
continuous amplitude

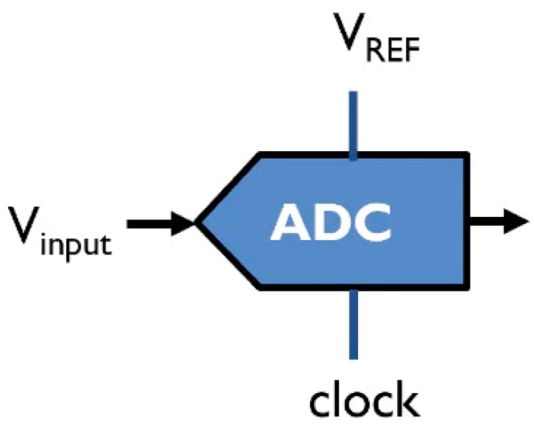
- **Resolution**
- **Quantization error**
- **Sampling rate**

Digital values $x(n)$

discrete time
discrete amplitude

ADC: Resolution

- Resolution: number of binary bits in ADC output.



$$ADC\ Output = round\left((2^N - 1) \times \frac{V_{input}}{V_{REF}}\right)$$

- **N**: resolution
- **V_{REF}**: the maximum input voltage that can be converted by the ADC

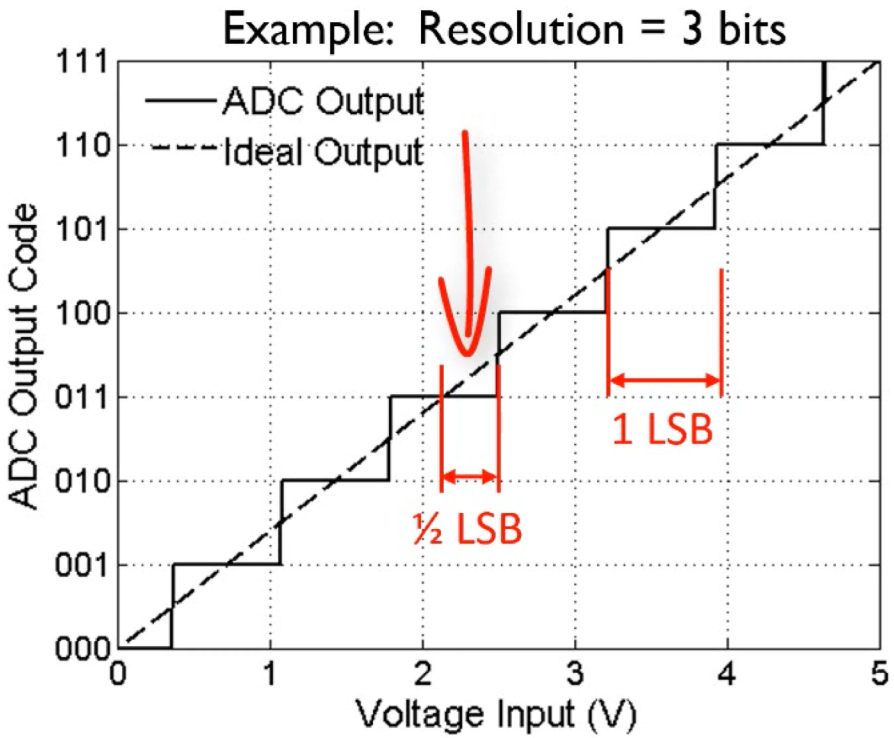
Example: for 10-bit ADC, V_{input} = 1V, V_{REF} = 5V

$$ADC\ Output = round\left(1023 \times \frac{1}{5}\right) = round(204.6) = 205 = 0xCD$$

ADC: Quantization Error

- Resolution N: number of binary bits in ADC output.

$$Digital\ Result = round\left((2^N - 1) \times \frac{V_{input}}{V_{REF}}\right)$$



Least significant bit (LSB) voltage

$$LSB = \frac{V_{REF}}{2^N - 1}$$

Max quantization error = ±½ LSB

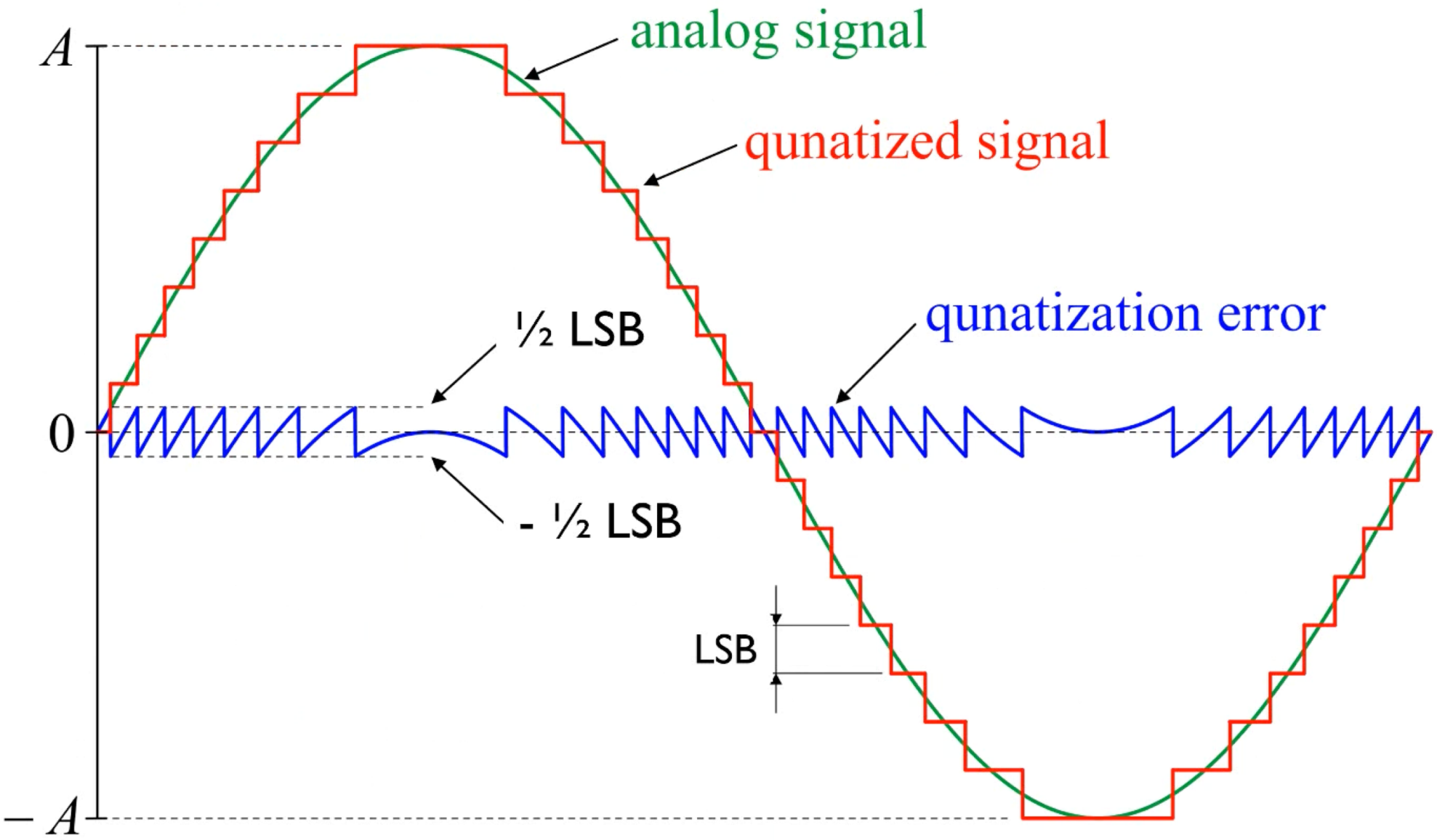
Example: 3-bit ADC and input range [0, 5V]

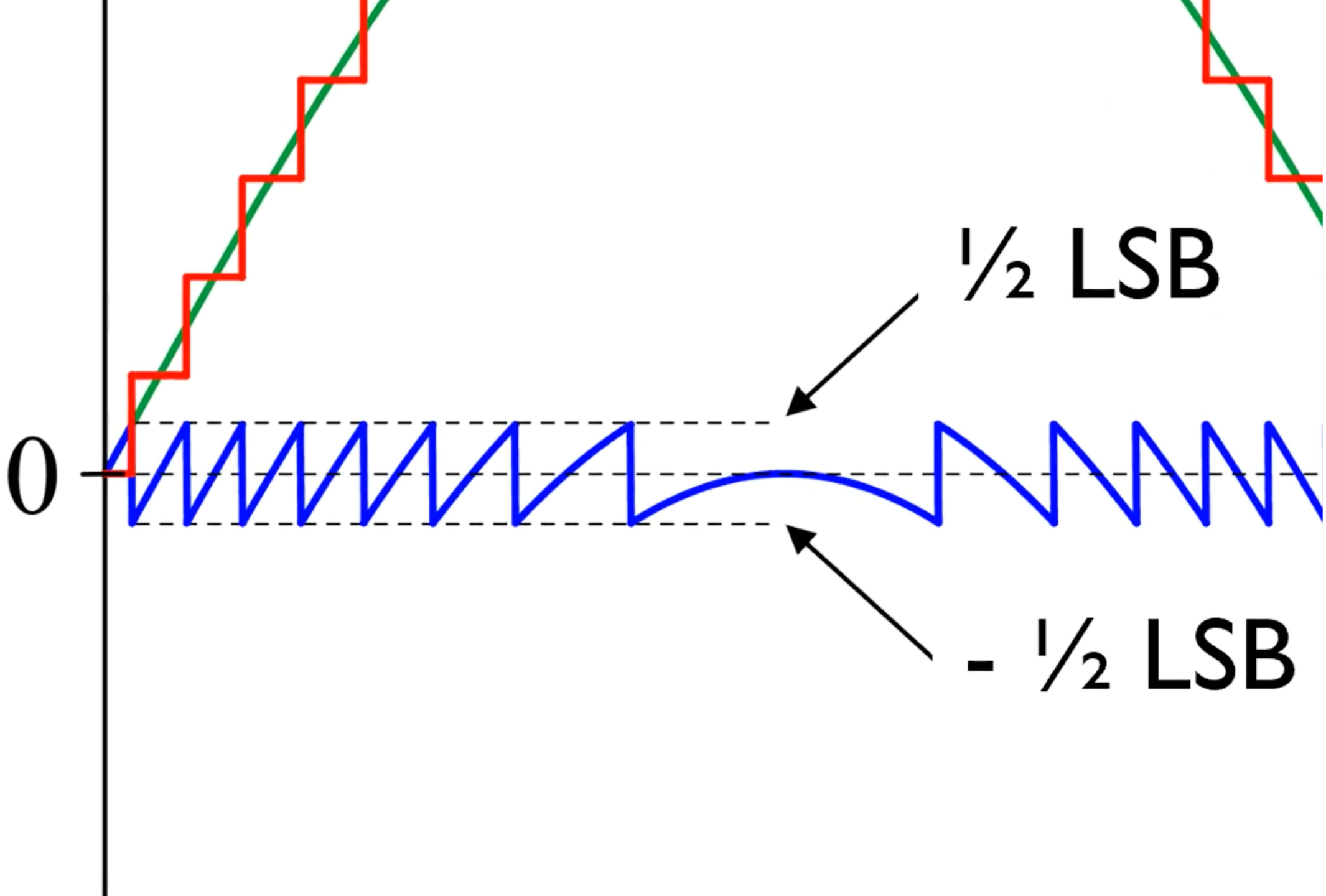
$$Max\ Quantization\ Error = \pm\frac{1}{2}LSB$$

$$= \pm\frac{5V}{2 \times (2^3 - 1)}$$

$$= \pm 0.357\ V$$

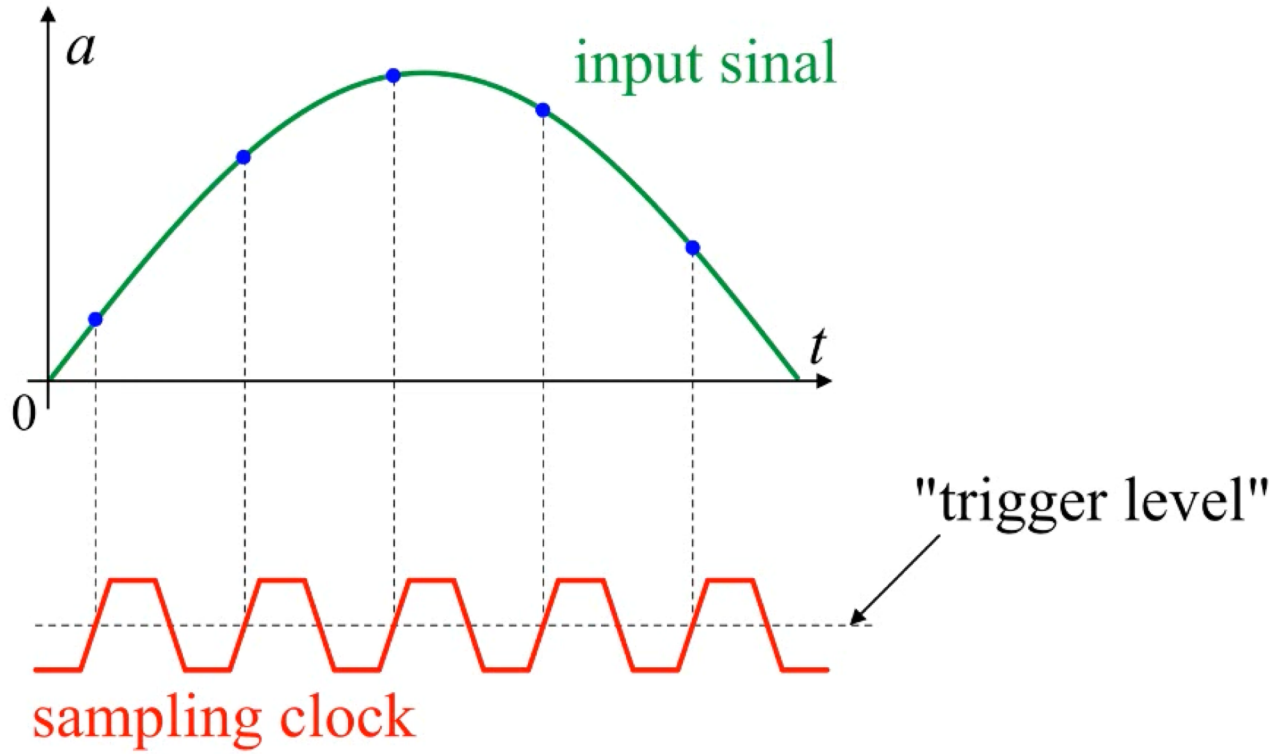
Example of Quantization Error



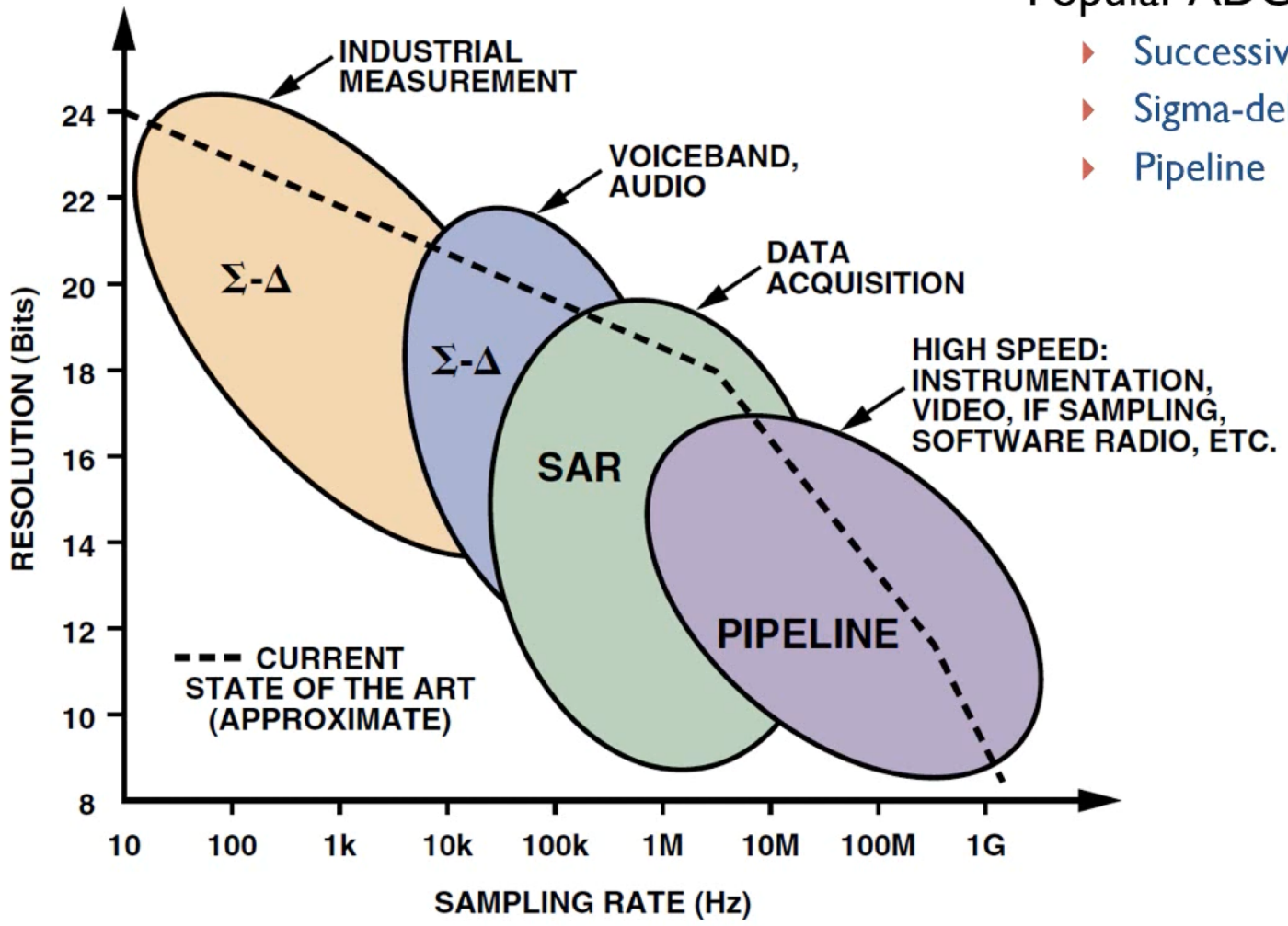

½ LSB
0
- ½ LSB

ADC: Sampling rate

- Number of ADC output samples available per unit time.



ADC Architecture

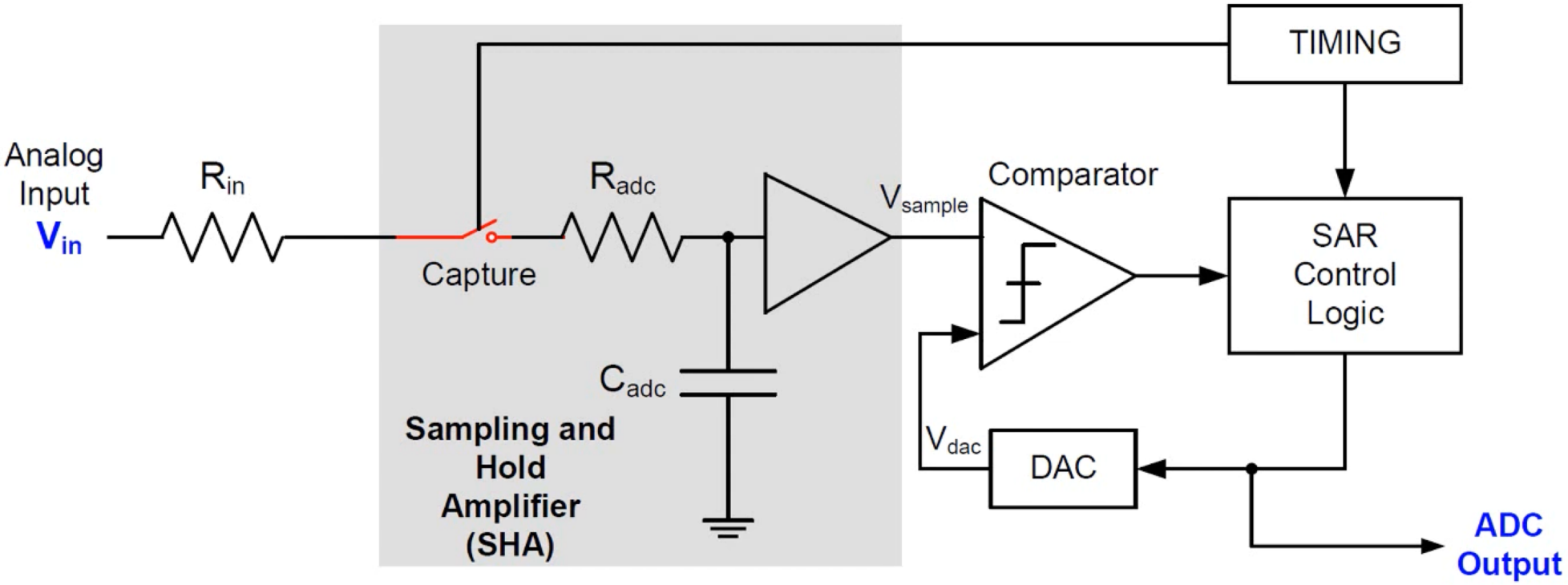


Popular ADC Architectures:

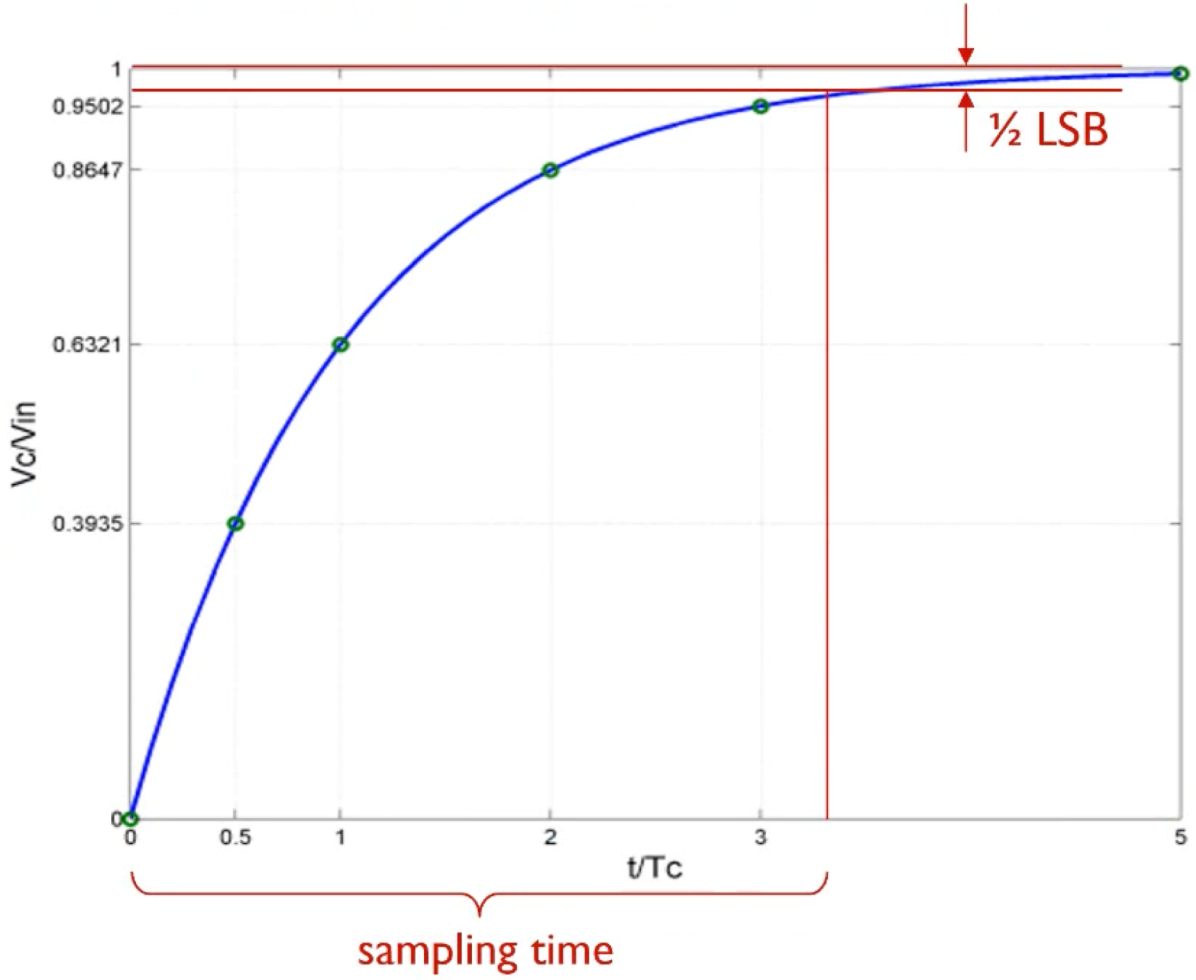
- Successive-approximation (SAR)
- Sigma-delta (Σ-Δ),
- Pipeline

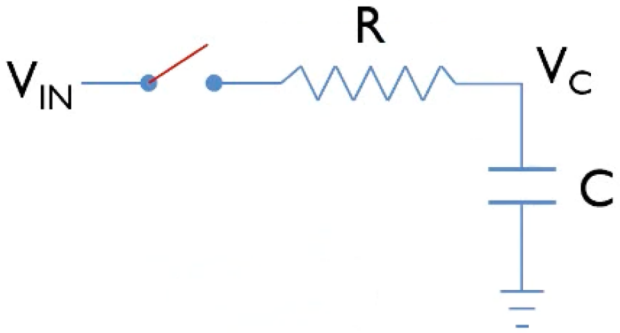
https://www.analog.com/en/analog-dialogue/articles/the-right-adc-architecture.html

Successive-approximation (SAR)



Determining Minimum Sampling Time



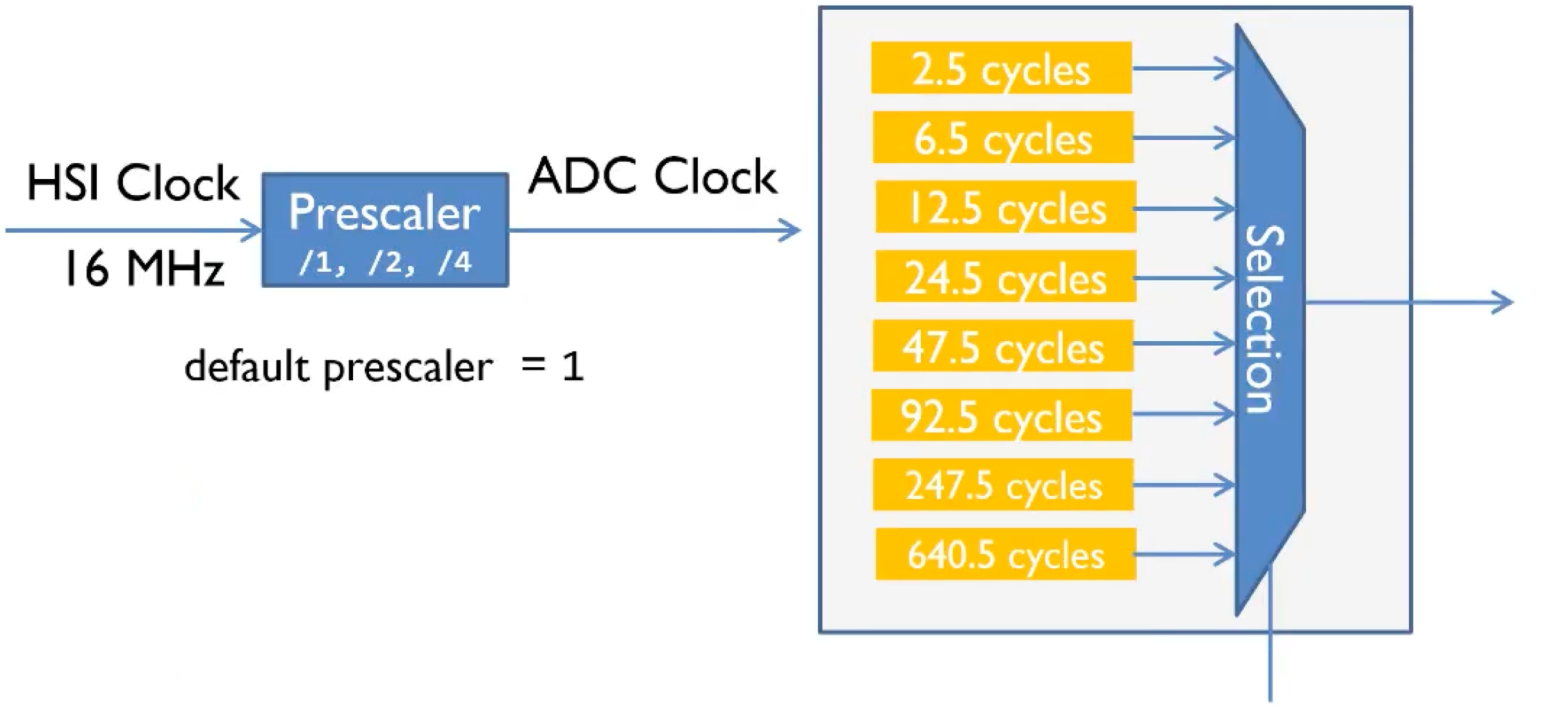


$$V_C(t) = V_{in} \times \left(1 - e^{-\frac{t}{T_c}}\right)$$

$$T_c = R \cdot C$$

- Sampling time is software programmable
- Sampling time must large enough to settle within ½ LSB

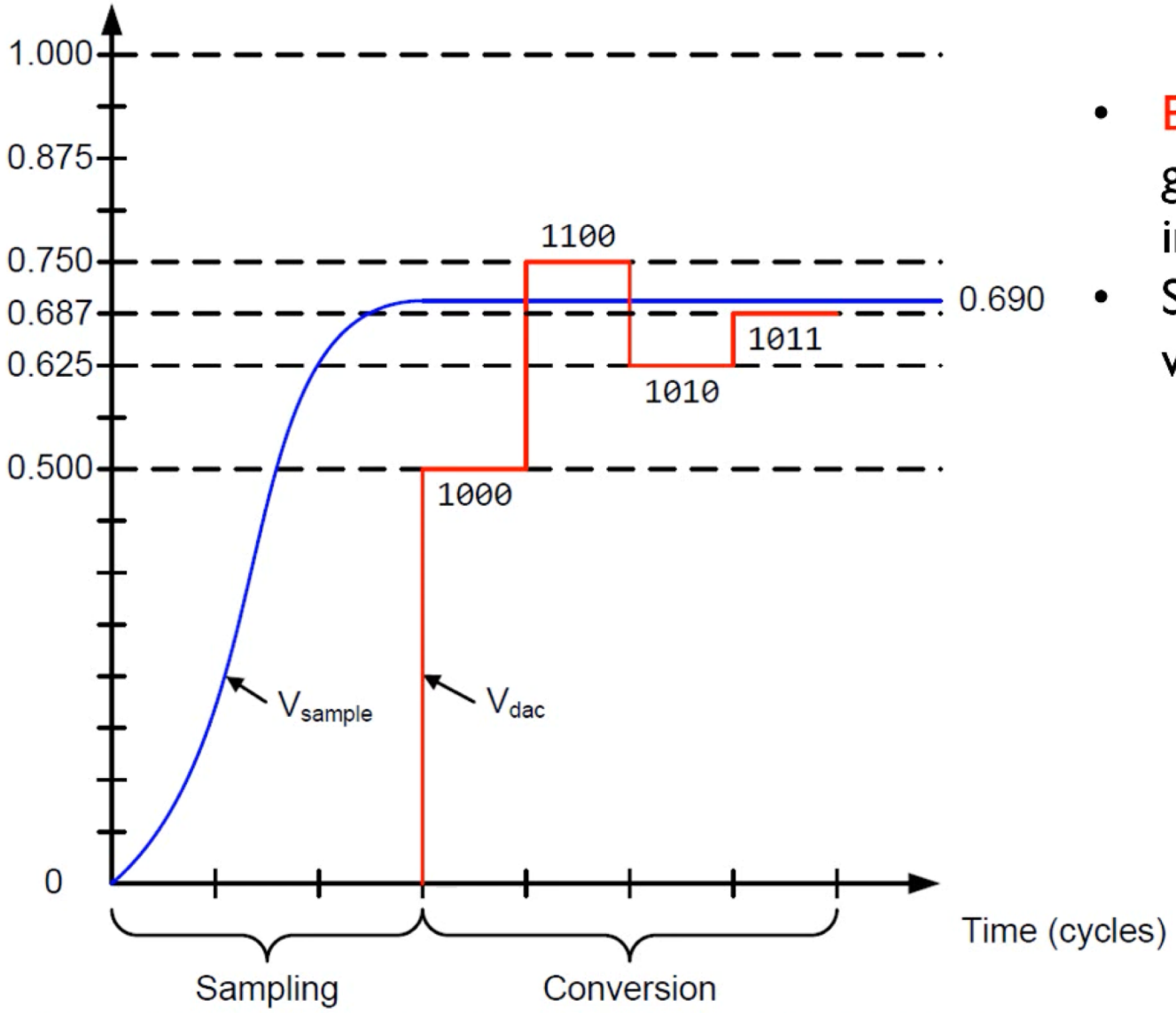
Programming ADC Sampling Time



default prescaler = 1

ADC sample time registers
(**SMPR1** and **SMPR2**)

Successive-approximation (SAR) ADC



- Binary search algorithm to gradually approaches the input voltage
- Settle into ±½ LSB bound within the time allowed

ADC Conversion Time

$$T_{ADC} = T_{sampling} + T_{Conversion}$$

Suppose ADC clock = 16 MHz, and Sampling time = 4 cycles

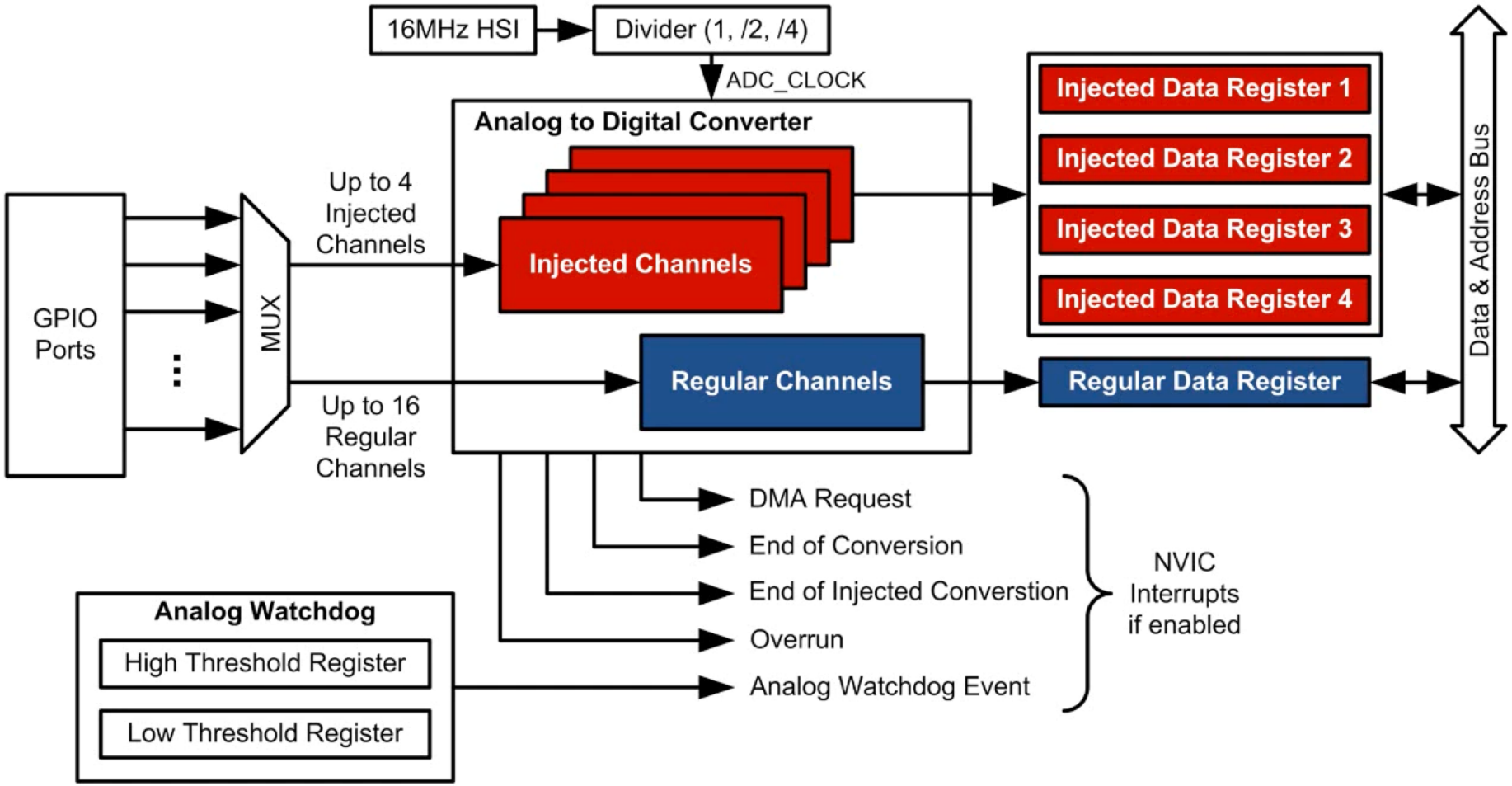
For 12-bit ADC

$$T_{ADC} = 4 + 12 = 16\ cycles = 1\ \mu s$$

For 6-bit ADC

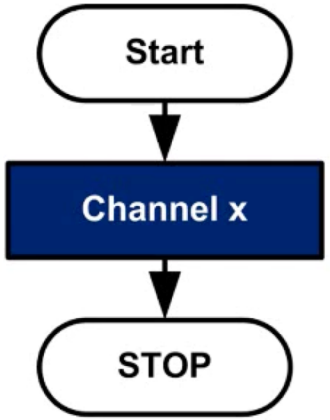
$$T_{ADC} = 4 + 6 = 10\ cycles = 0.625\ \mu s$$

ADC: Regular *vs* injected



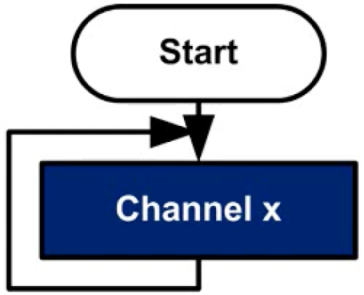
ADC Conversion Modes

Single Channel Mode



Single Channel, Single Conversion Mode

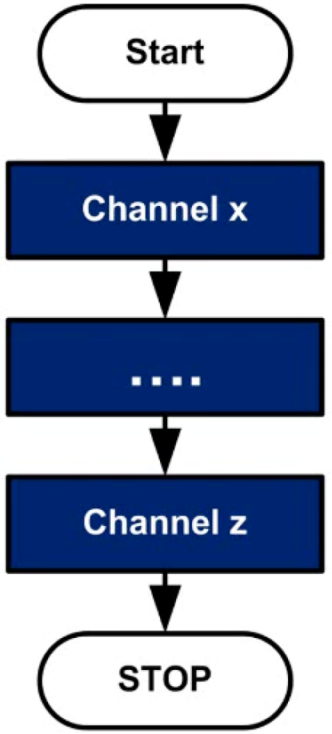
CONT in ADC_CR2 = 0
SCAN in ADC_CR2 = 0



Single Channel, Continuous Conversion Mode

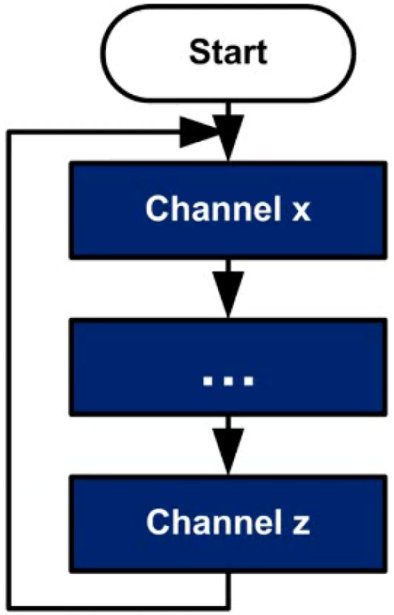
CONT in ADC_CR2 = 1
SCAN in ADC_CR2 = 0

Scan Mode



Scan Mode with Single Conversion

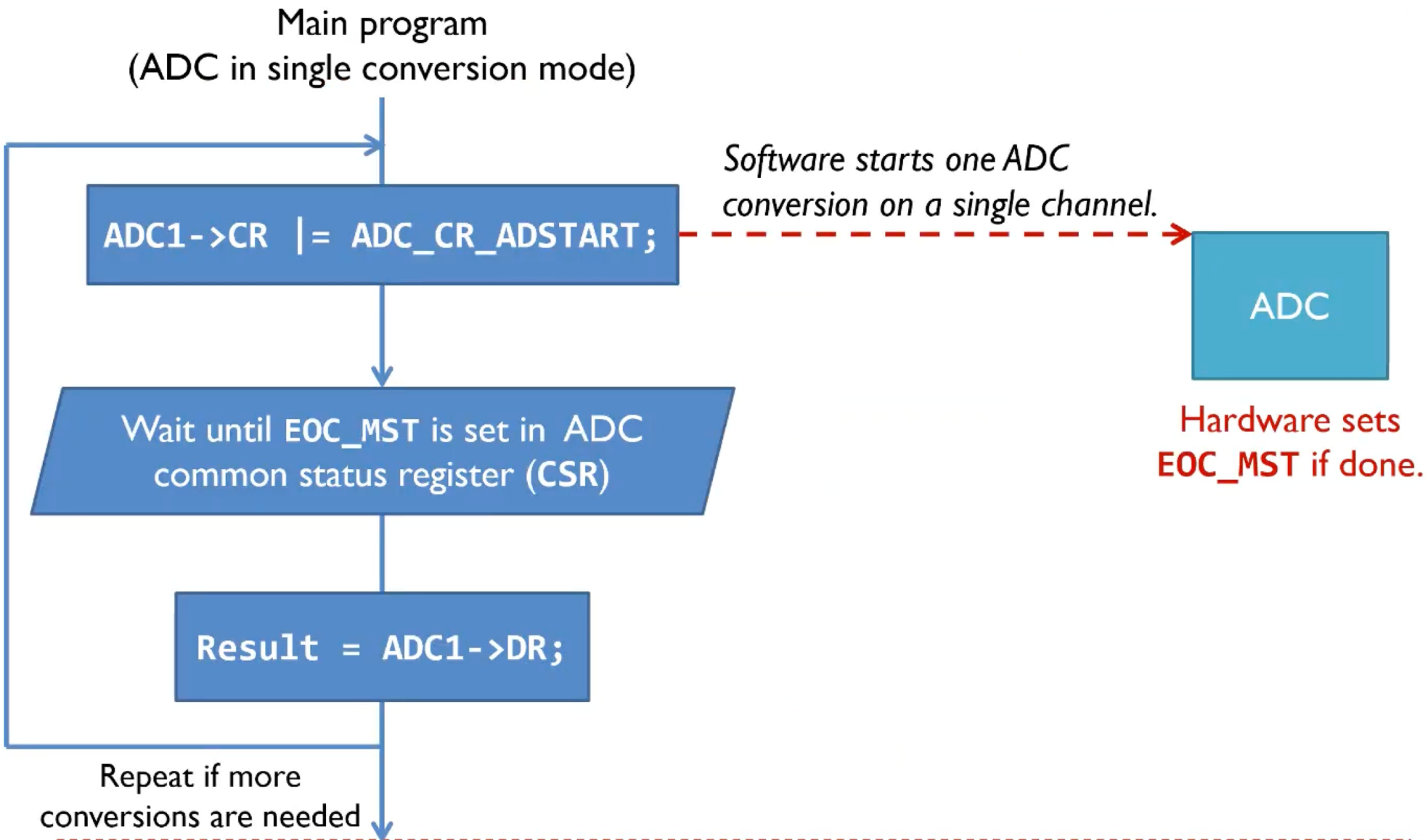
SCAN in ADC_CR2 = 1
CONT in ADC_CR2 = 0



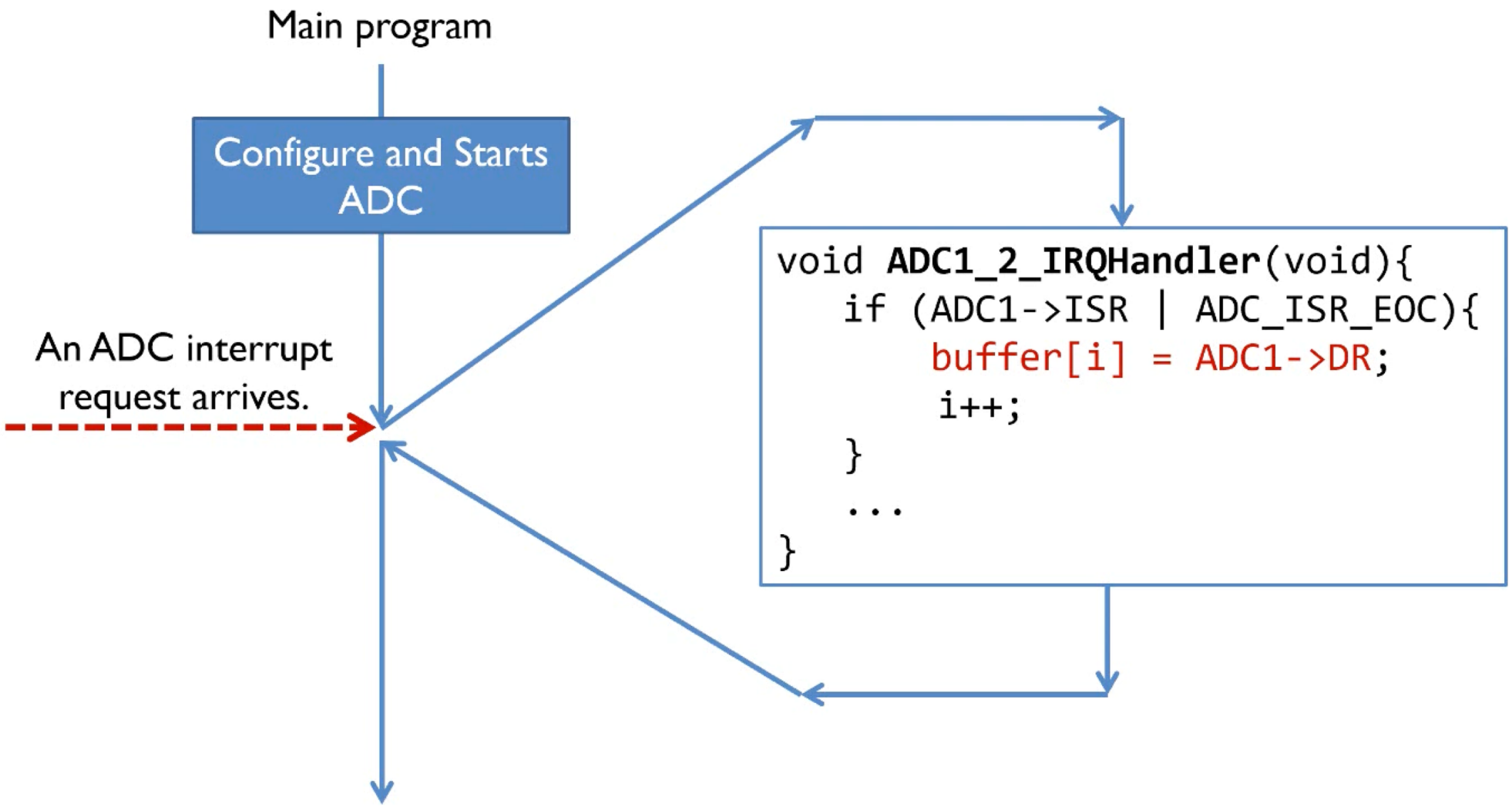
Scan Mode with Continuous Conversion

SCAN in ADC_CR2 = 1
CONT in ADC_CR2 = 1

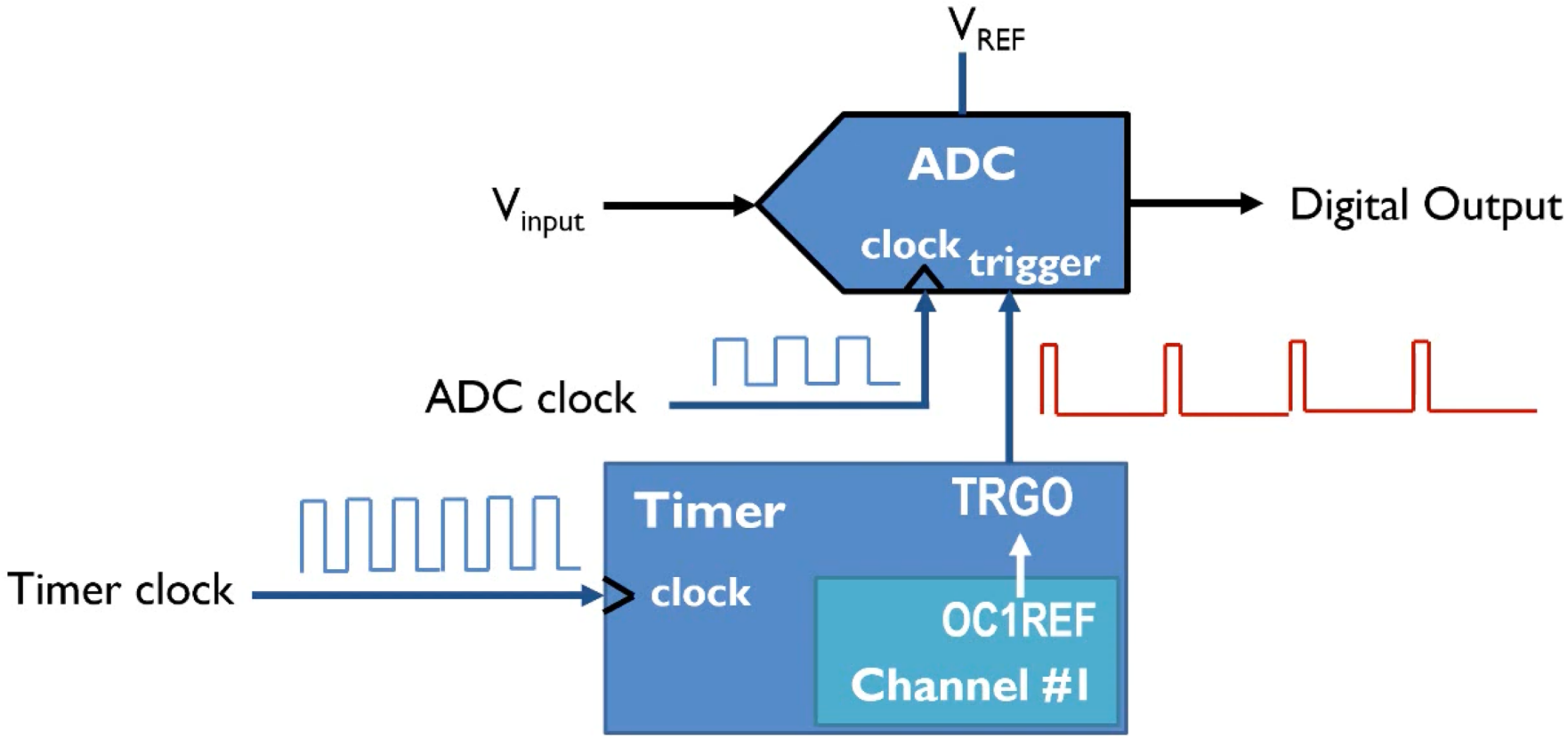
Example 1: ADC with Polling (Busy Waiting)



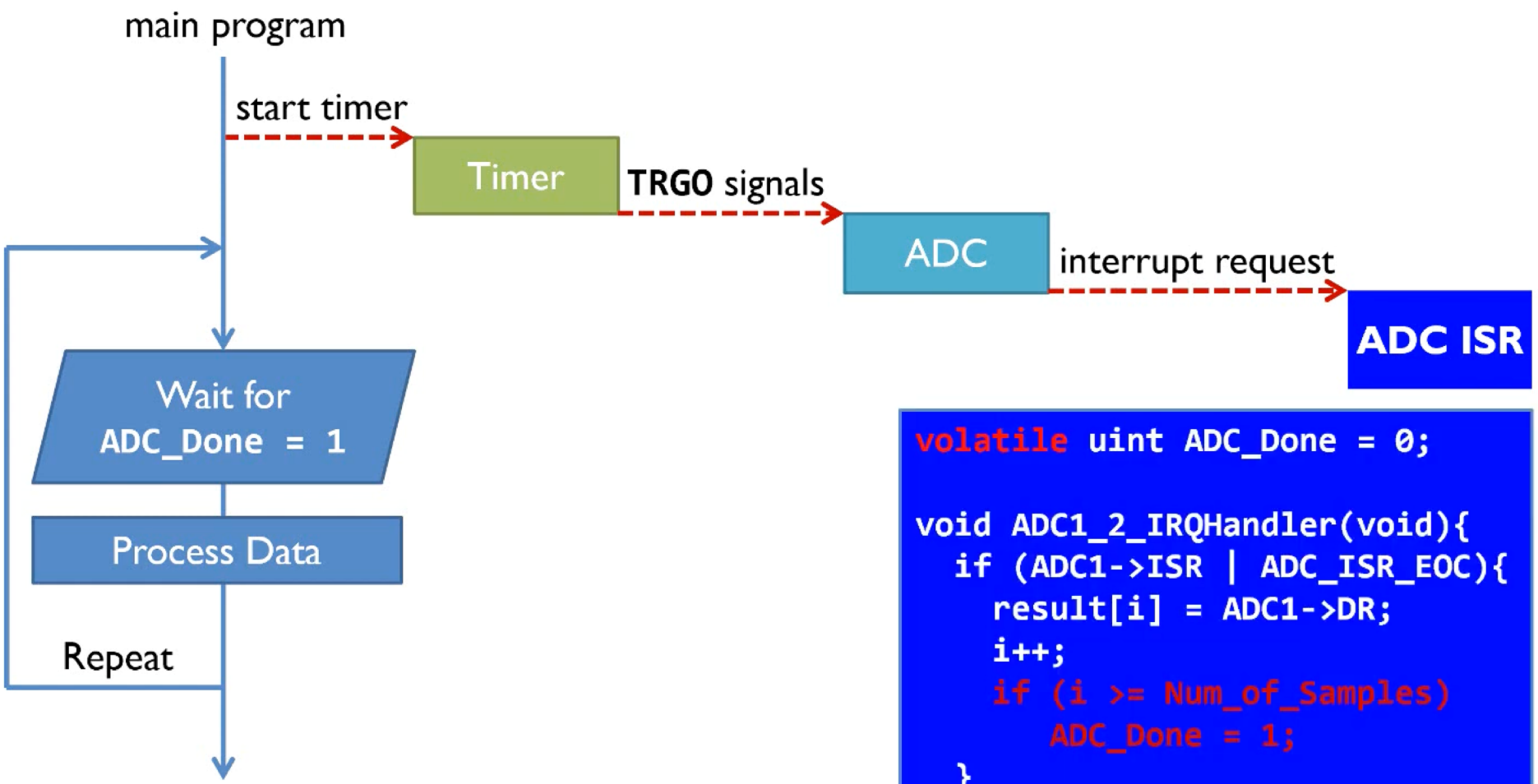
Example 2: ADC with Interrupts



Example 3: ADC Triggered by a Timer



Example 3: ADC Triggered by a Timer

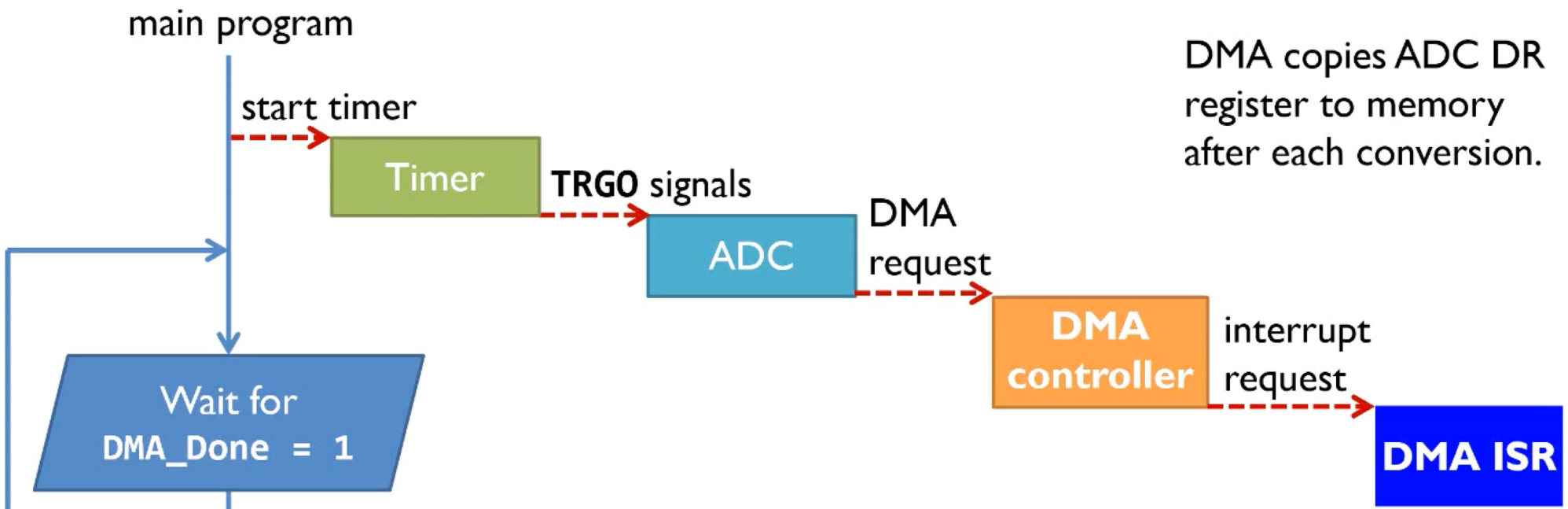



```
volatile uint ADC_Done = 0;

void ADC1_2_IRQHandler(void){
  if (ADC1->ISR | ADC_ISR_EOC){
    result[i] = ADC1->DR;
    i++;
    if (i >= Num_of_Samples)
       ADC_Done = 1;
  }
  ...
}
```

Make N sampling

Example 4: ADC Triggered by a Timer with DMA



DMA copies ADC DR register to memory after each conversion.

```
volatile uint DMA_Done = 0;

void DMA1_Channel1_IRQHandler(void){
  ...
  DMA_Done = 1;
  ...
}
```

Make N sampling

KONEC