Lecture 9
Analog to Digital Convertor

Embedded Systems

DCSE, UET Peshawar

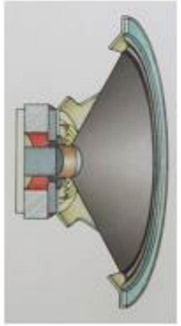
Bilal Habib

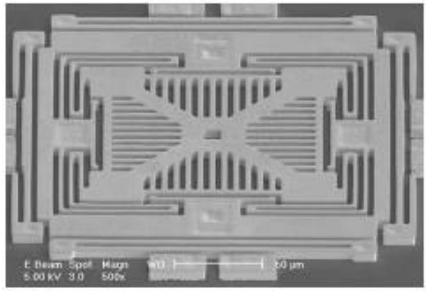
Some slides adopted from Prabal Dutta
University of Michigan

We live in an analog world

- Everything in the physical world is an analog signal
  - Sound, light, temperature, pressure

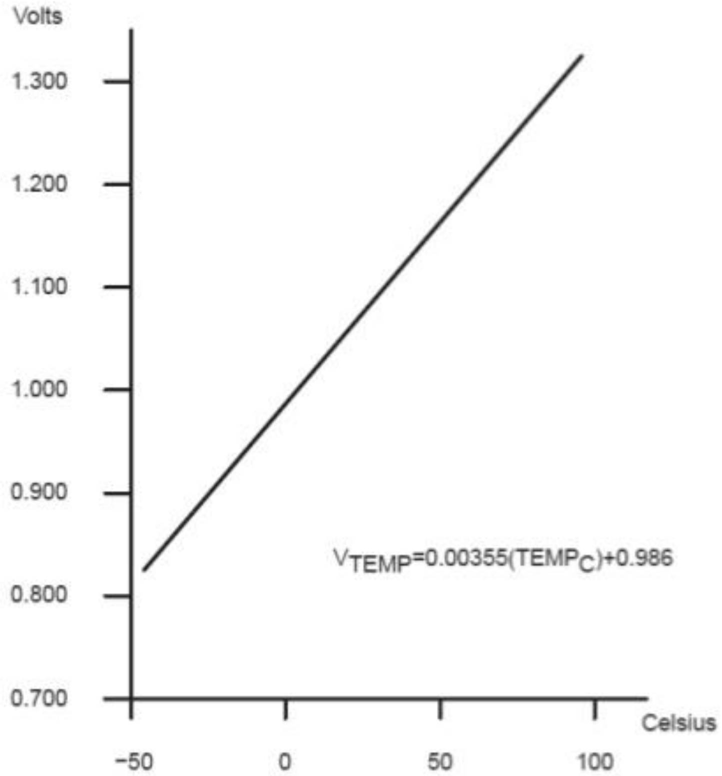
- Need to convert into electrical signals
  - Transducers: converts one type of energy to another
    - Electro-mechanical, Photonic, Electrical, …
  - Examples
    - Microphone/speaker
    - Thermocouples
    - Accelerometers





Transducers convert one form of energy into another

- Transducers
  - Allow us to convert physical phenomena to a voltage potential in a well-defined way.



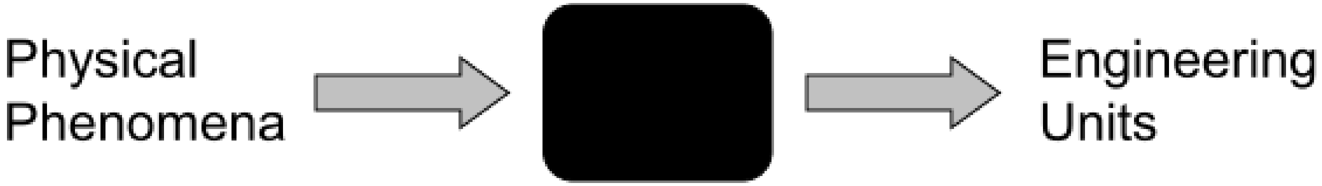
A transducer is a device that converts one type of energy to another. The conversion can be to/from electrical, electro-mechanical, electromagnetic, photonic, photovoltaic, or any other form of energy. While the term transducer commonly implies use as a sensor/detector, any device which converts energy can be considered a transducer. – *Wikipedia.*

Many other common sensors (some digital)

- Force
  - strain gauges - foil, conductive ink
  - conductive rubber
  - rheostatic fluids
    - Piezorestive (needs bridge)
  - piezoelectric films
  - capacitive force
    - Charge source
- Sound
  - Microphones
    - Both current and charge versions
  - Sonar
    - Usually Piezoelectric
- Position
  - microswitches
  - shaft encoders
  - gyros
- Acceleration
  - MEMS
  - Pendulum
- Monitoring
  - Battery-level
    - voltage
  - Motor current
    - Stall/velocity
  - Temperature
    - Voltage/Current Source
- Field
  - Antenna
  - Magnetic
    - Hall effect
    - Flux Gate
- Location
  - Permittivity
  - Dielectric

Going from analog to digital

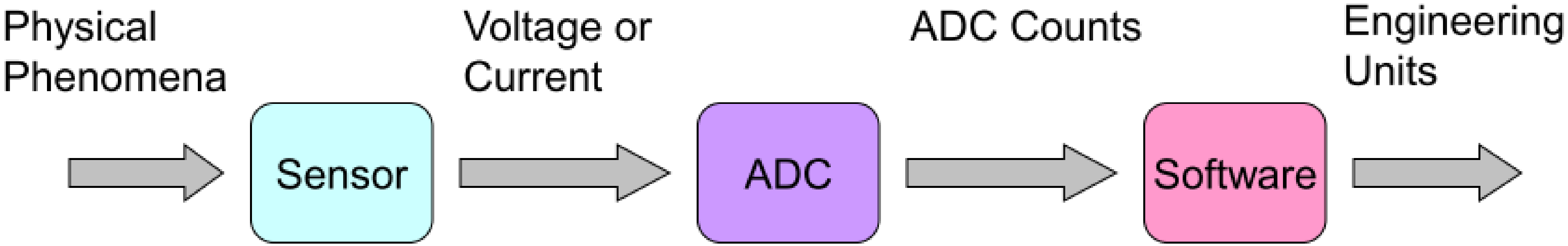
- What we want



Physical Phenomena

Engineering Units

- How we have to get there



Physical Phenomena

Sensor

Voltage or Current

ADC

ADC Counts

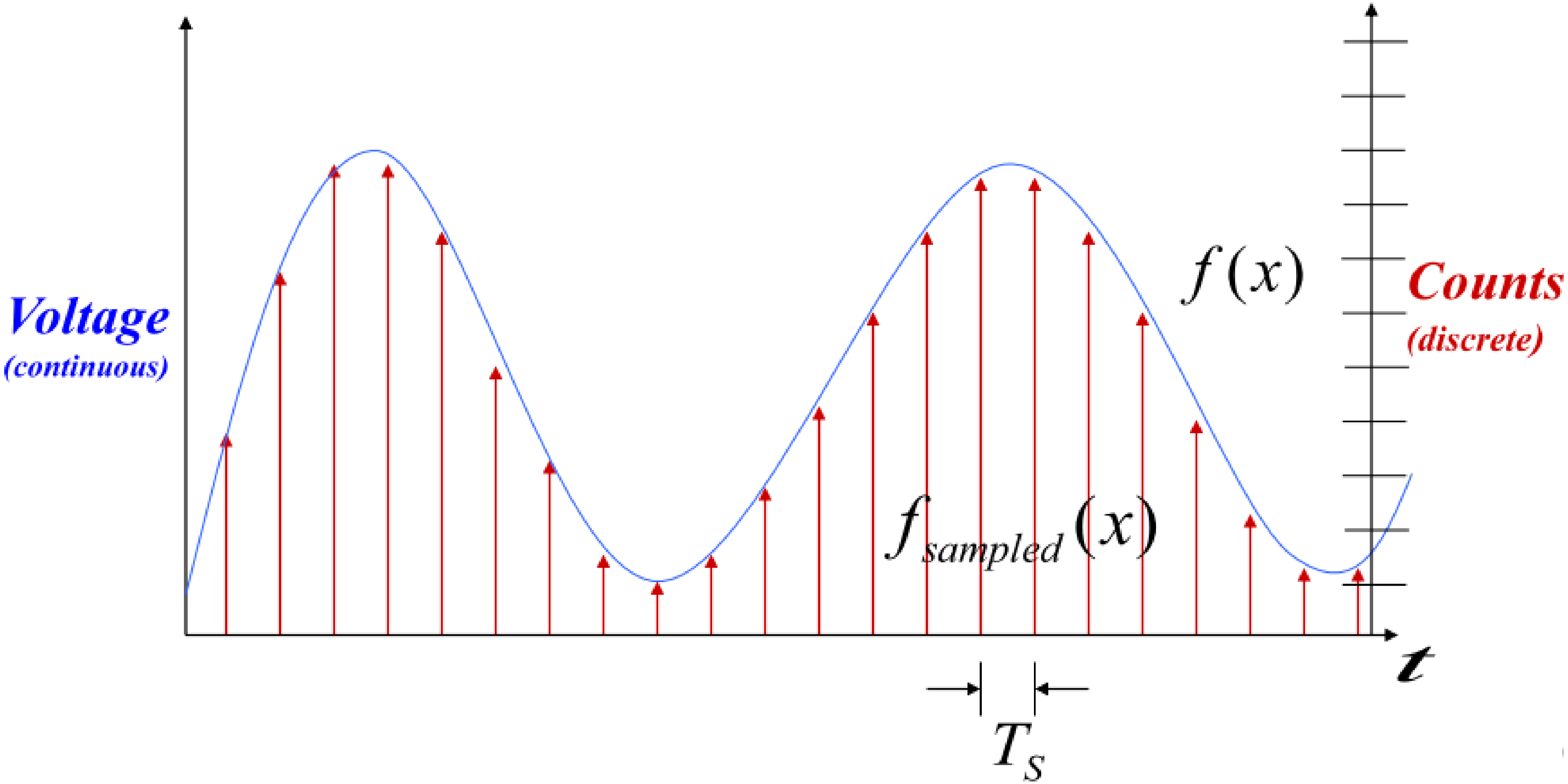
Software

Engineering Units

Representing an analog signal digitally

- How do we represent an analog signal (e.g. continuous voltage)?
  - As a time series of discrete values

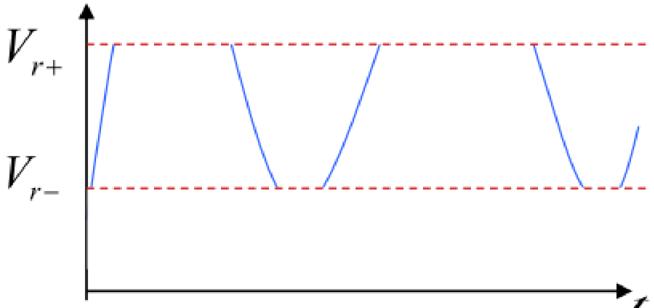
    → On MCU: read ADC data register (counts) periodically (T_s)



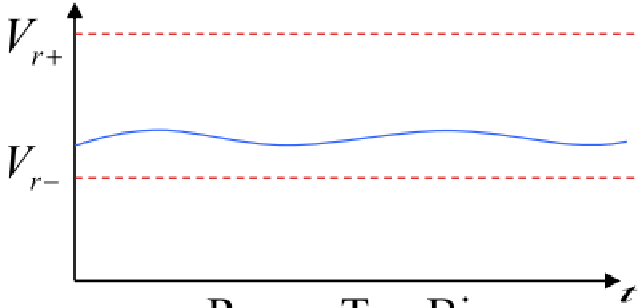
Choosing the range

- Fixed # of bits (e.g. 8-bit ADC)
- Span a particular input voltage range
- What do the sample values represent?
  - Some fraction within the range of values

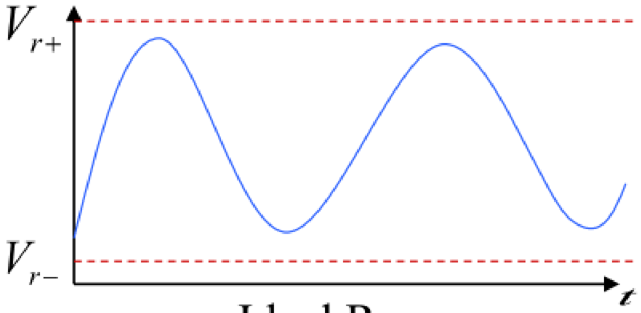
    → *What range to use?*



Range Too Small



Range Too Big



Ideal Range

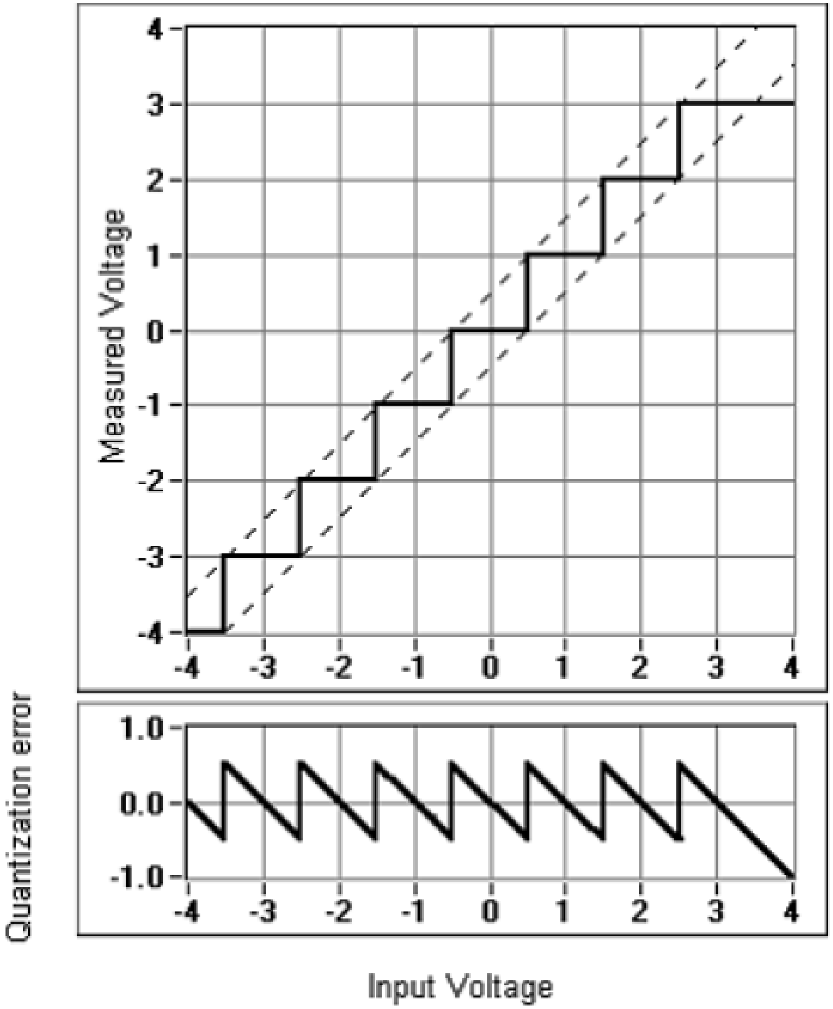
Choosing the granularity

- Resolution
  - Number of discrete values that represent a range of analog values
  - MSP430: 12-bit ADC
    - 4096 values
    - Range / 4096 = Step

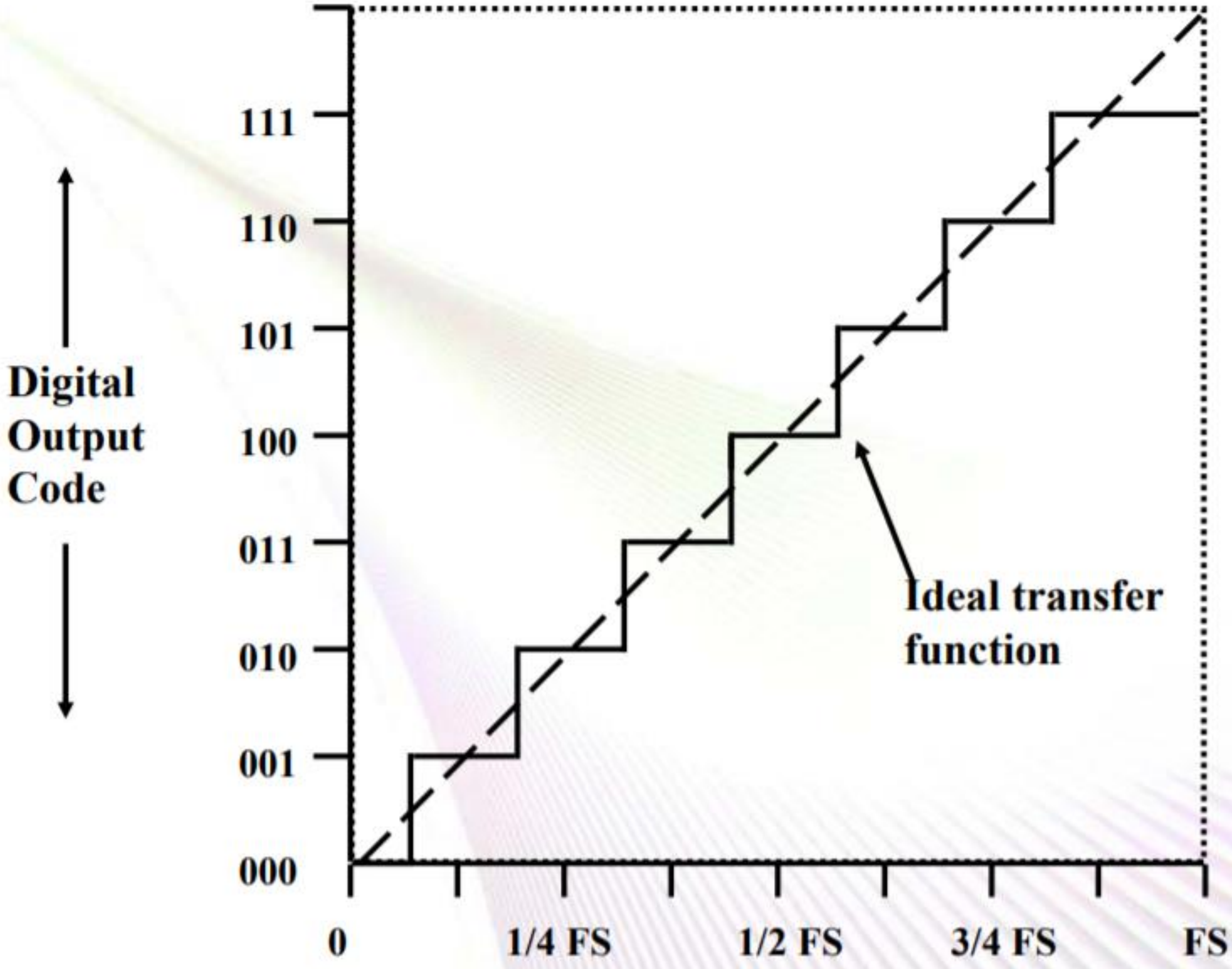
  Larger range ➔ less info / bit

- Quantization Error
  - How far off discrete value is from actual
  - ½ LSB → Range / 8192

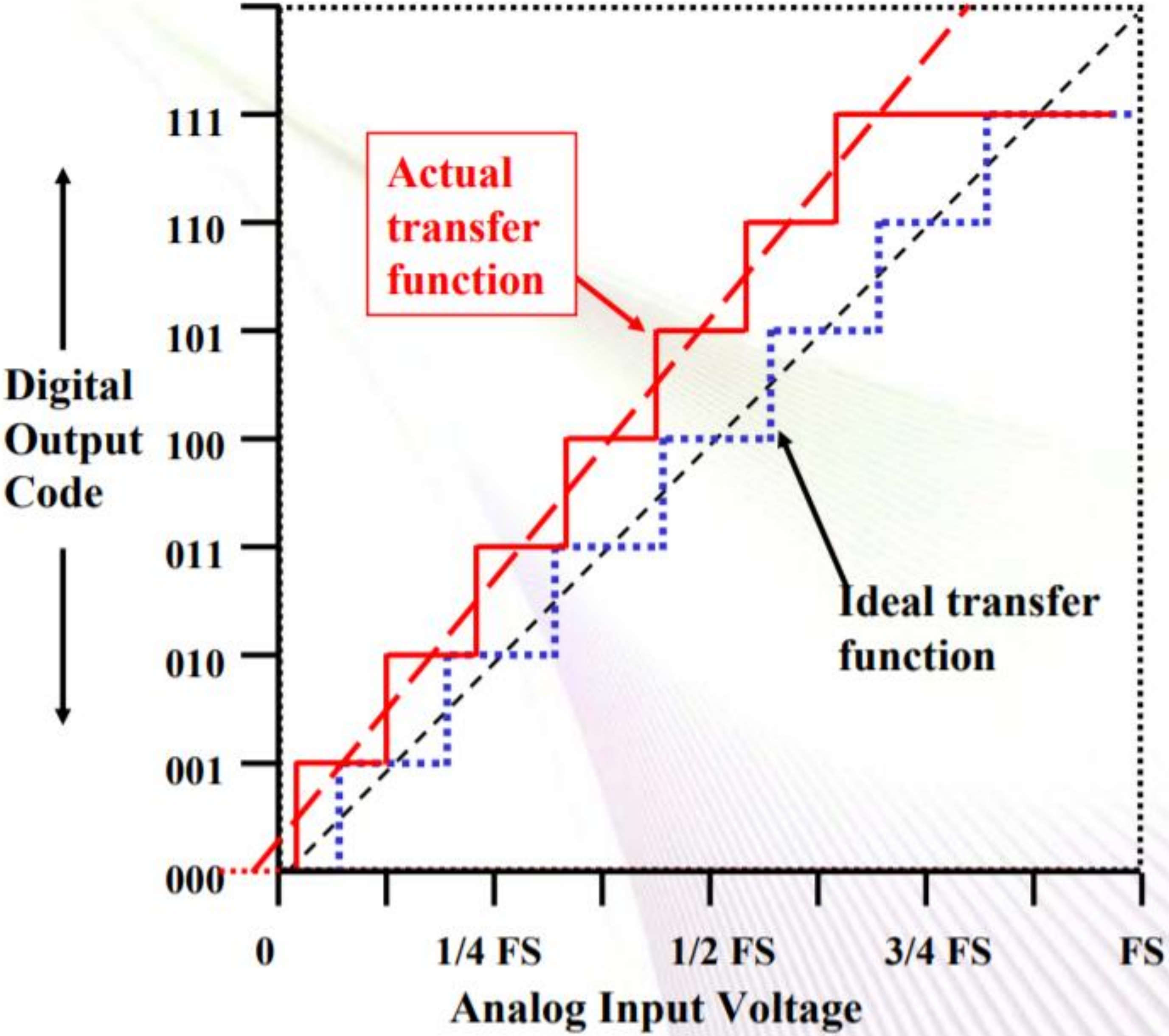
  Larger range ➔ larger error



ADC Ideal Transfer Function



ADC with Offset and Gain Error

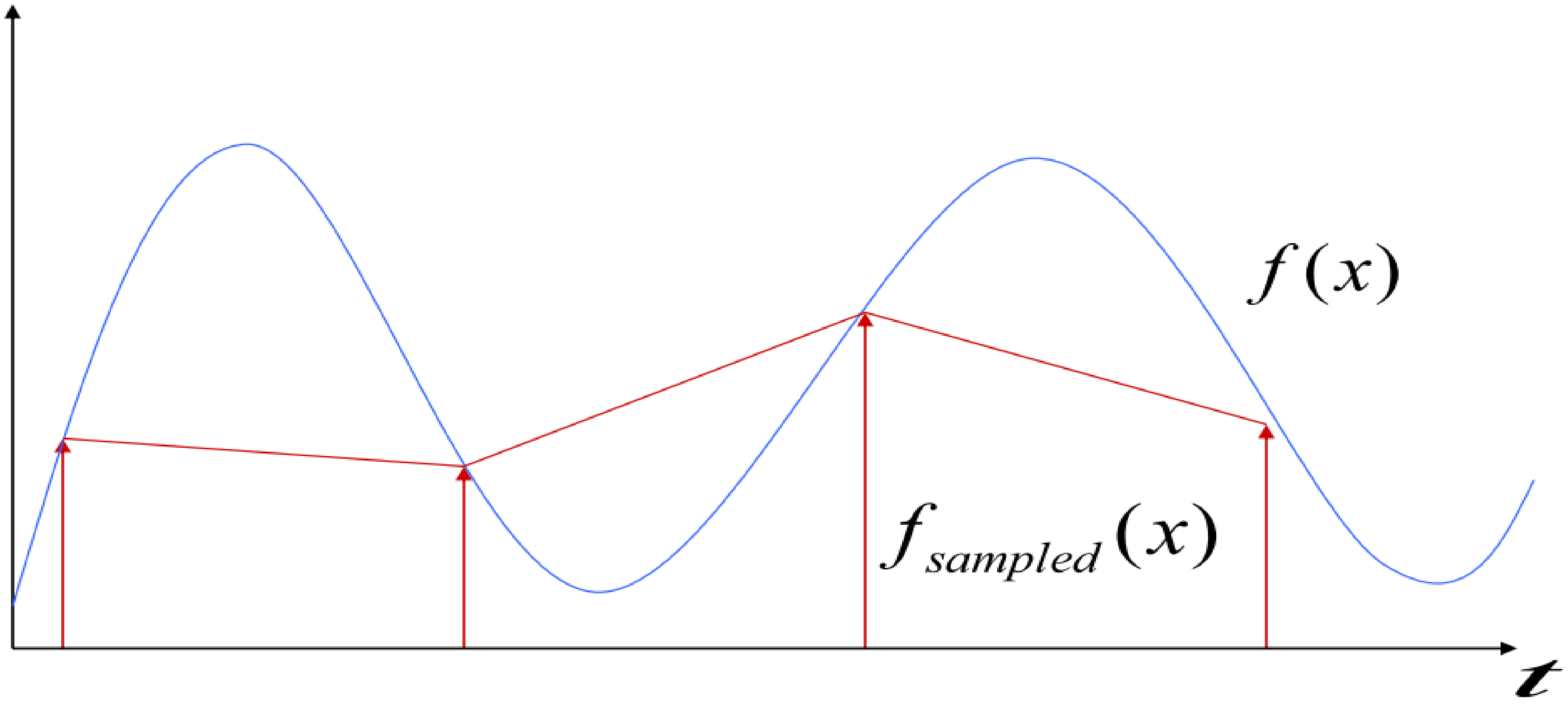


- y = a + (1+b)x

  where
  y=digital out
  x=analog in
  a=offset err
  b=gain err
- Every Ideal Code has Offset Error added
- Every ideal code is Multiplied by Gain Error

Choosing the sample rate

- What sample rate do we need?
  - Too little: we can't reconstruct the signal we care about
  - Too much: waste computation, energy, resources



Shannon-Nyquist sampling theorem

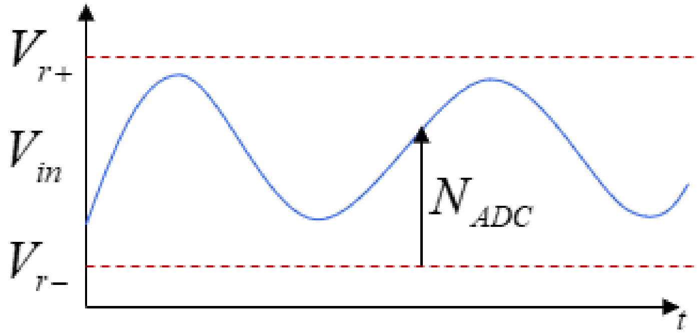
- *If a continuous-time signal $f(x)$ contains no frequencies higher than $f_{\max}$, it can be completely determined by discrete samples taken at a rate:*

$$f_{\text{samples}} > 2 f_{\max}$$

- Example:
  - Humans can process audio signals 20 Hz – 20 KHz
  - Audio CDs: sampled at 44.1 KHz

Converting between voltages, ADC counts, and engineering units

- Converting: ADC counts ⇔ Voltage



$$N_{ADC} = 4095 \times \frac{V_{in} - V_{r-}}{V_{r+} - V_{r-}}$$

$$V_{in} = N_{ADC} \times \frac{V_{r+} - V_{r-}}{4095}$$

- The ADC core uses two programmable/selectable voltage levels (V_R+ and V_R-) to define the upper and lower limits of the conversion.

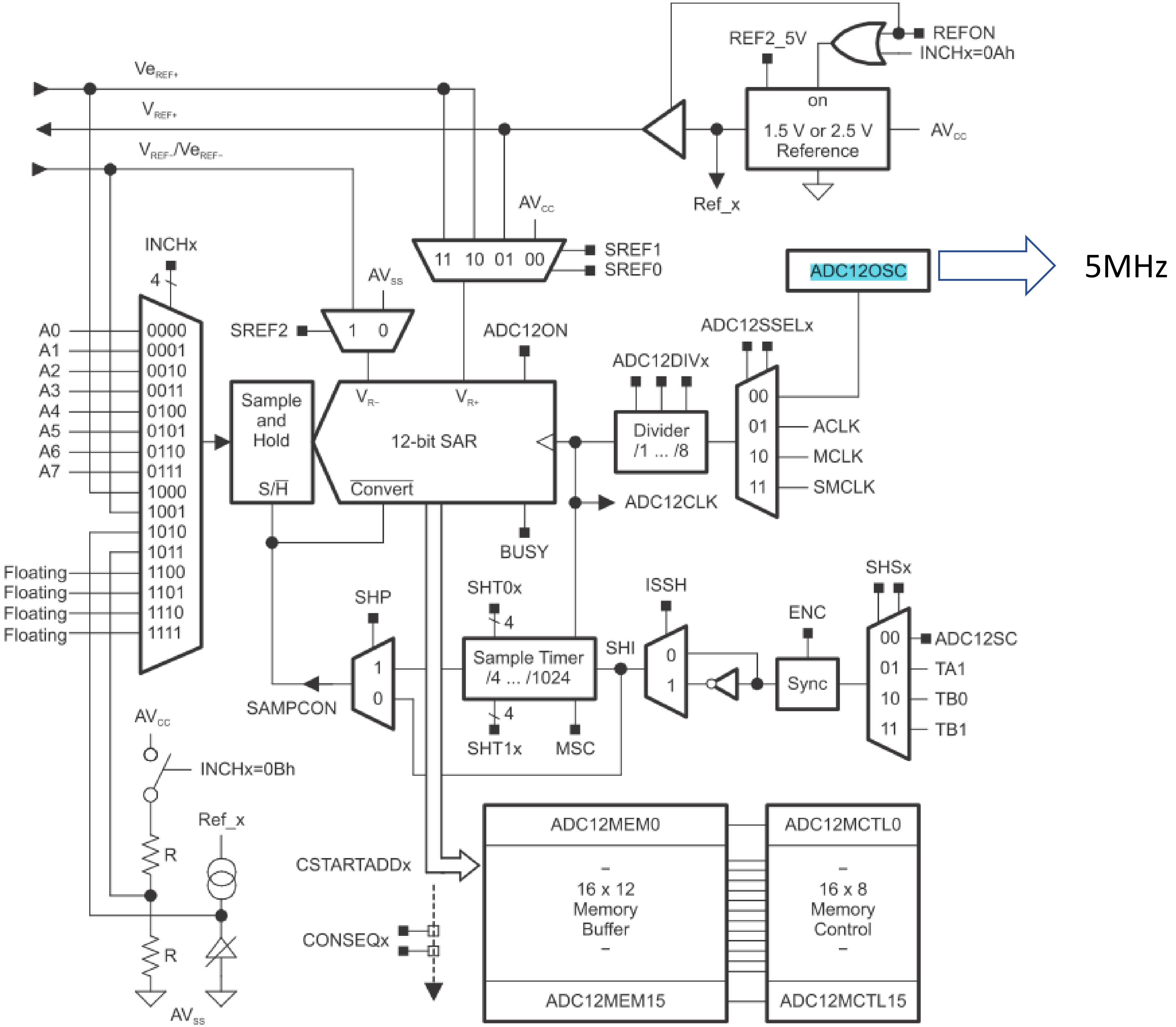
- N_{ADC} = 0xFFFh when Vin ≥ V_R+.
- N_{ADC} = 0 when Vin ≤ V_R-.

ADC12 Block Diagram

The **ADC12CLK** is used both as the conversion clock and to generate the sampling period when the pulse sampling mode is selected.

The ADC12 source clock is selected using the ADC12SSELx bits and can be divided from 1 through 8 using the ADC12DIVx bits.

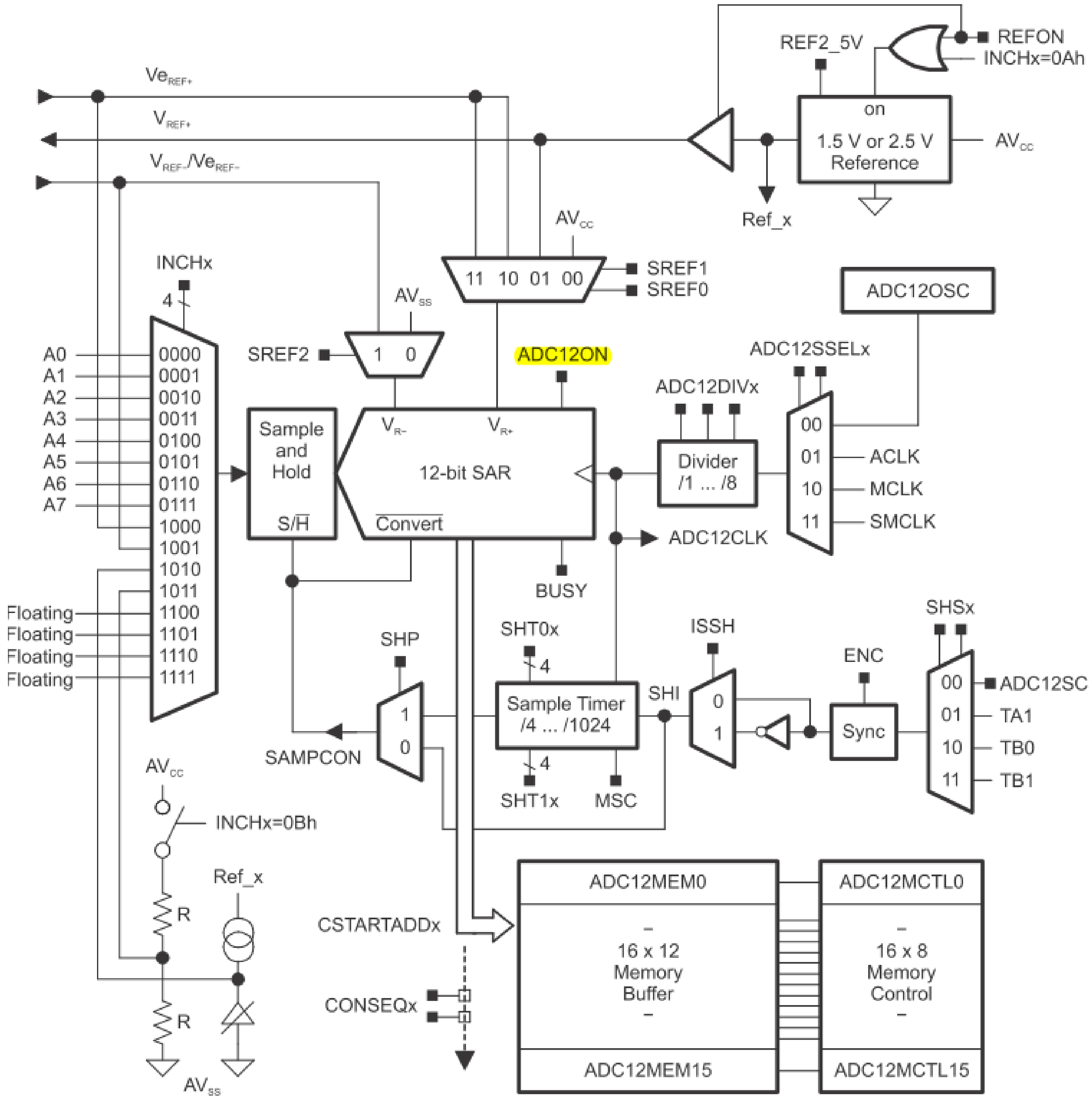
Possible ADC12CLK sources are SMCLK, MCLK, ACLK, and an internal oscillator ADC12OSC.



ADC12 Operation

The ADC core is enabled with the **ADC12ON** bit.

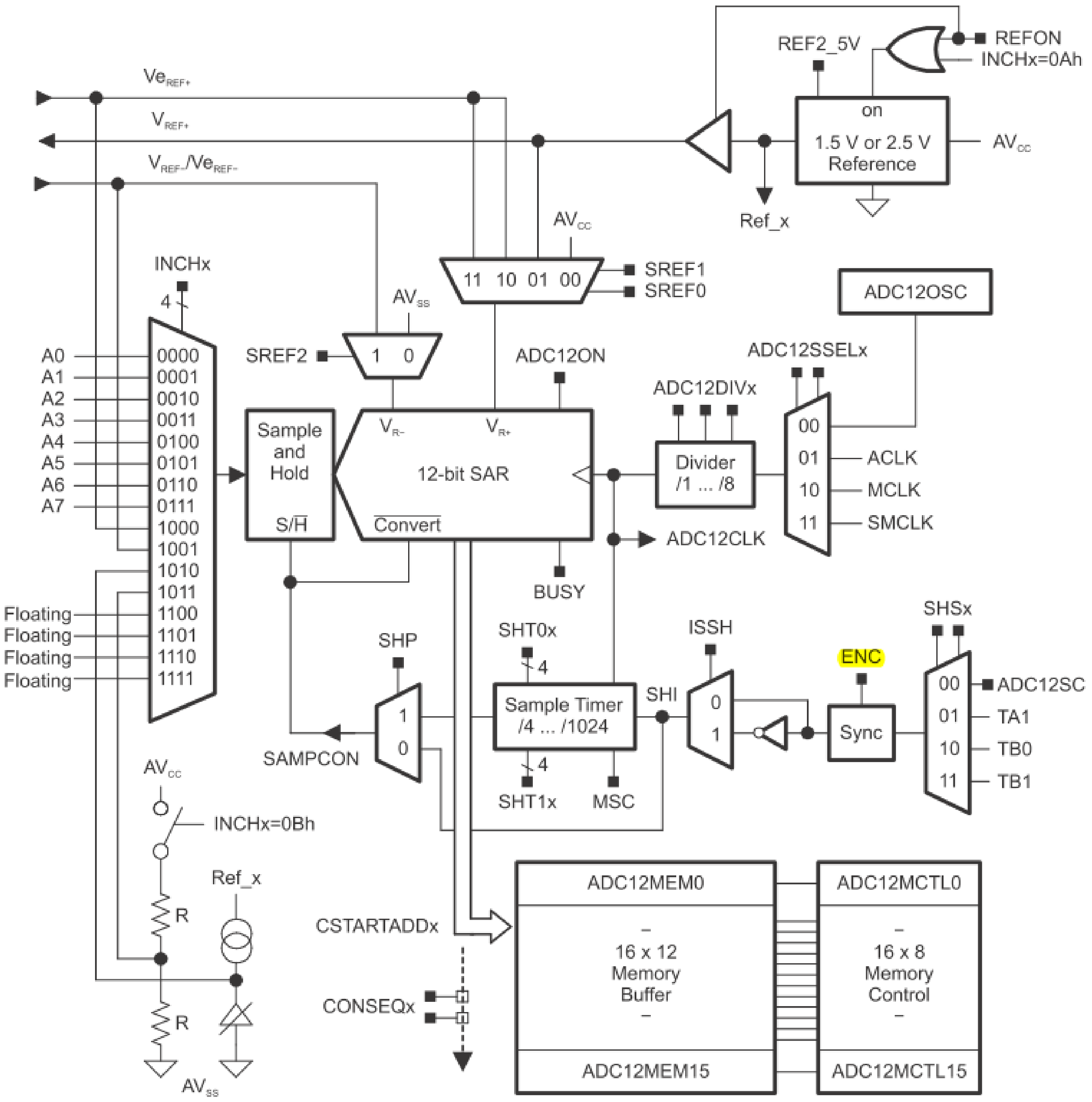
The ADC12 can be turned off when not in use to save power.



ADC12 Operation

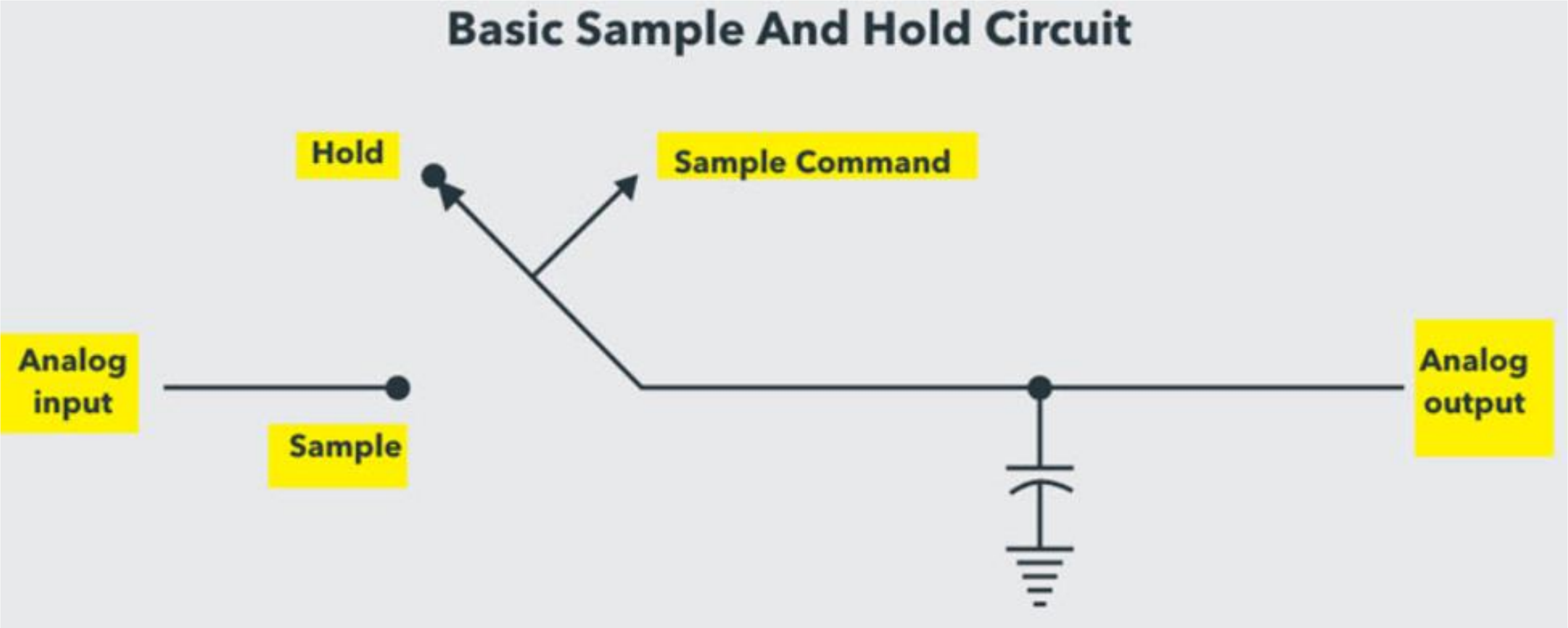
ADC12 control bits can only be modified when **ENC** = 0.

ENC must be set to 1 before any conversion can take place.



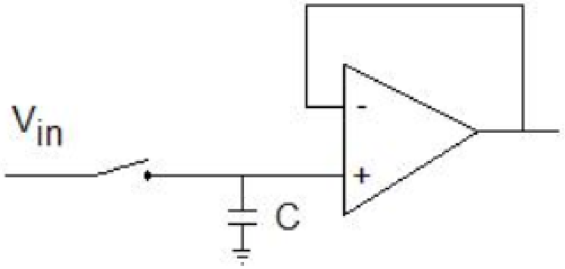
Sample and Hold Circuits
If the input analog voltage of an ADC changes more than ±1/2 LSB, then there is a severe chance that the output digital value is an error.

For the ADC to produce accurate results, the input analog voltage should be held constant for the duration of the conversion.



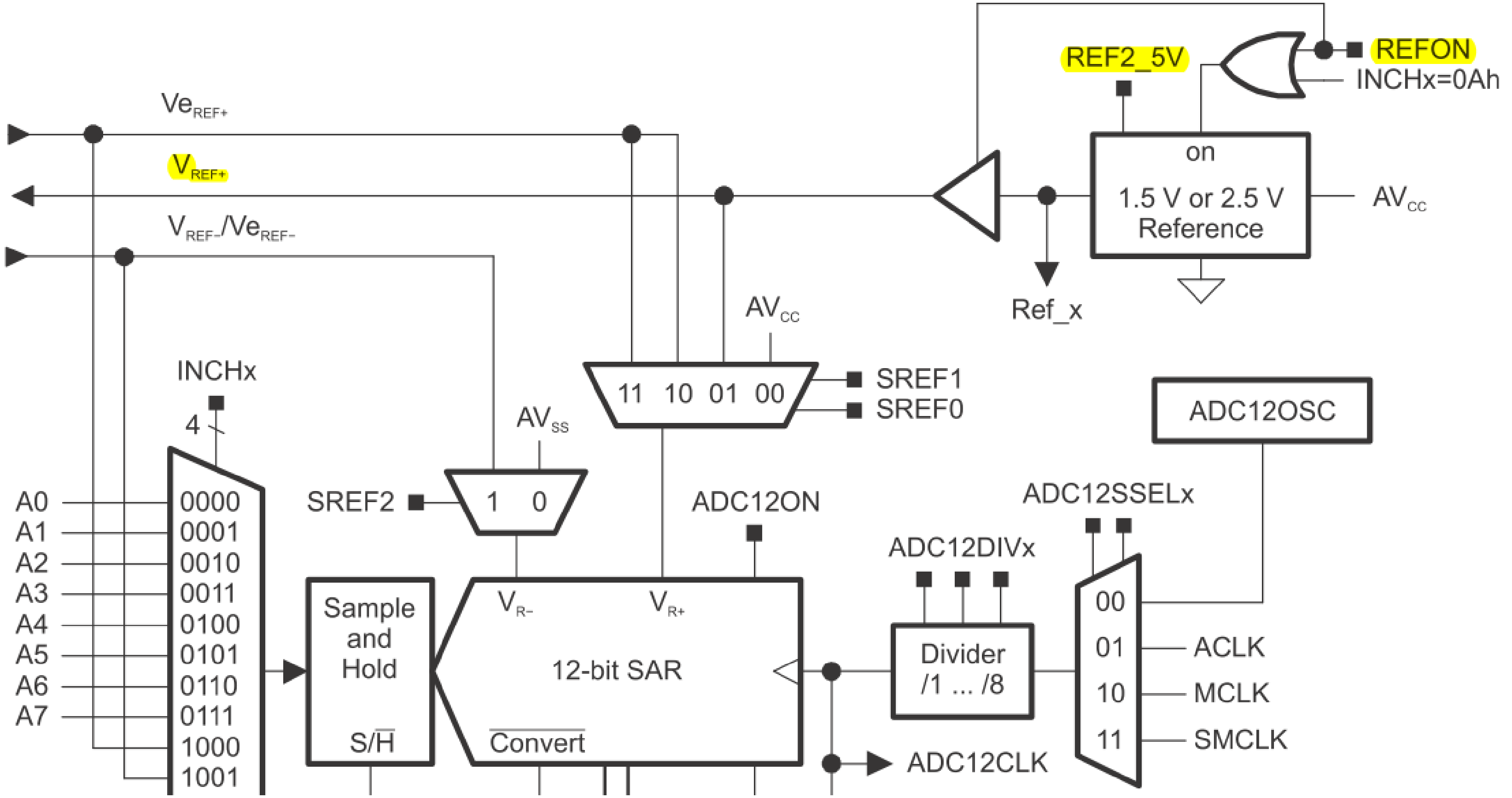
Source: Electronics Hub

Sample and Hold of ADC



- If the switch is left open, but momentarily close it when we want to grab a measurement, it is a sample and hold circuit.

ADC12 Operation



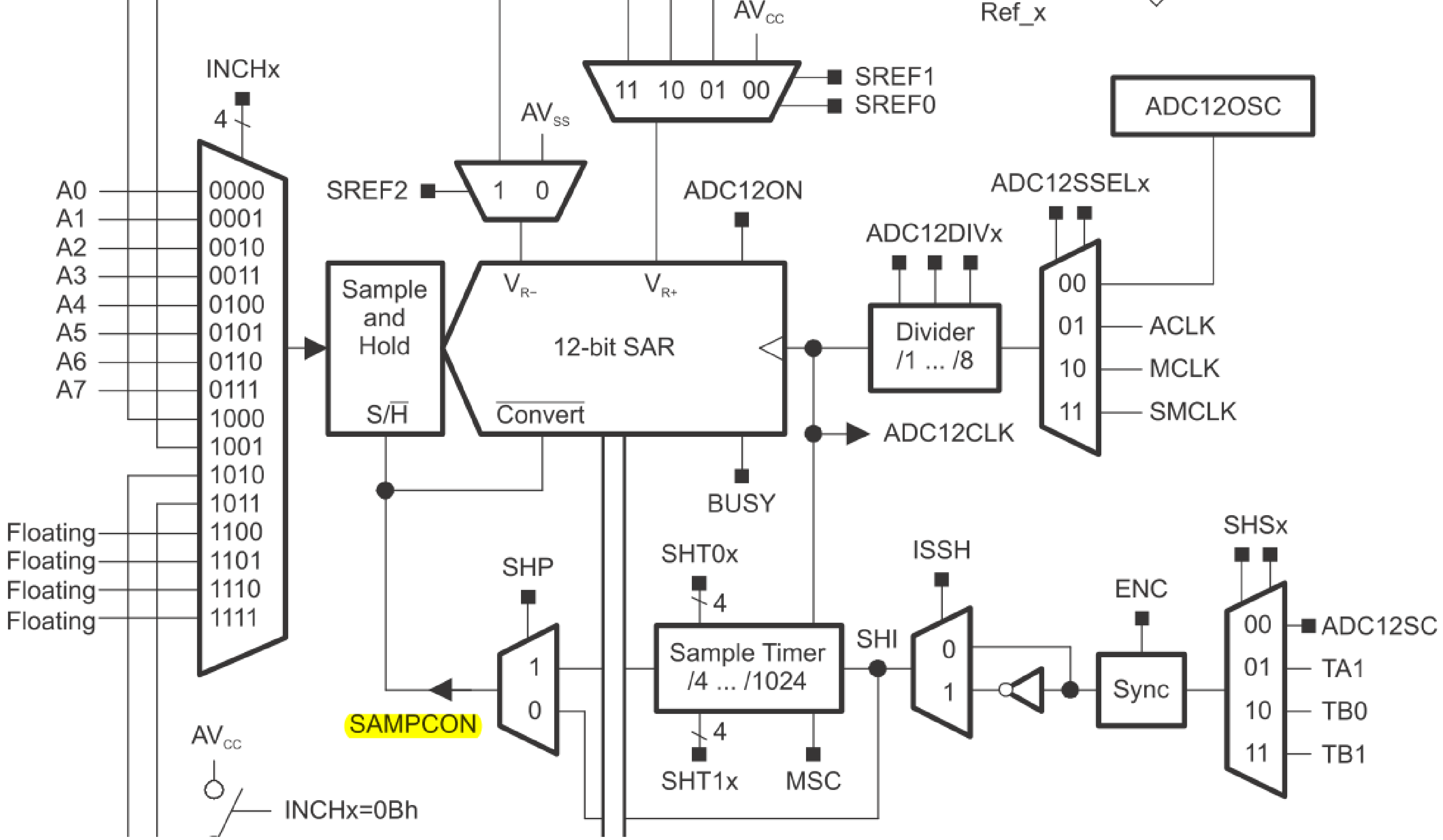
The ADC12 module contains a built-in voltage reference with two selectable voltage levels, 1.5 V and 2.5 V.

Either of these reference voltages may be used internally and externally on pin V$_{REF}$+ .

Setting REFON = 1 enables the internal reference.
When REF2_5V = 1, the internal reference is 2.5 V.
When REF2_5V = 0, the reference is 1.5 V.

The reference can be turned off to save power when not in use.

ADC12 Operation



The **SAMPCON** signal controls the sample period and start of conversion.

When SAMPCON is high, sampling is active. The high-to-low SAMPCON transition starts the analog-to-digital conversion, which requires 13 ADC12CLK cycles.

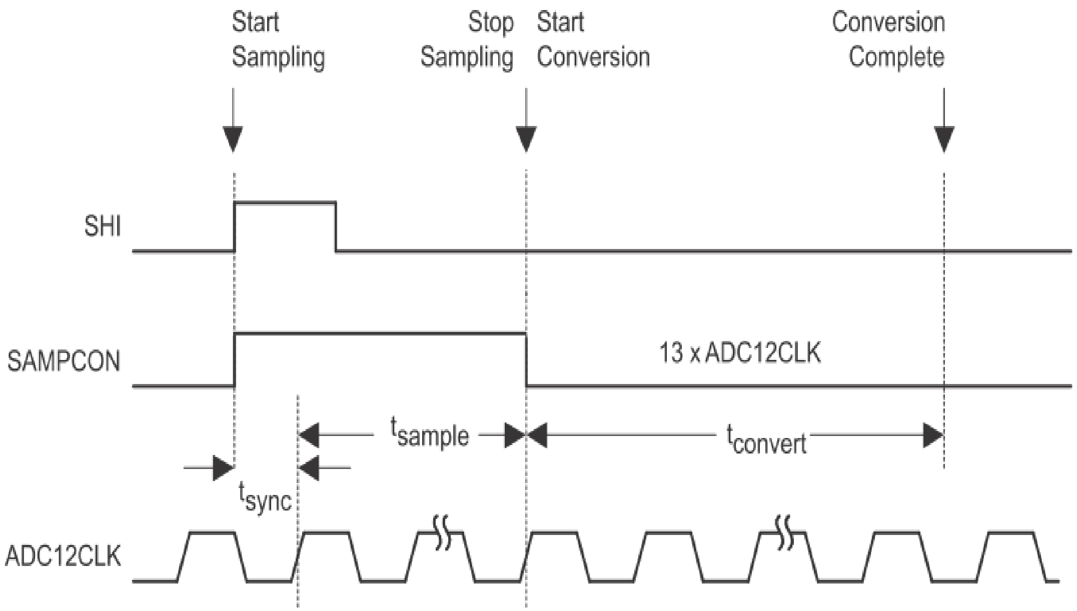
Two different sample-timing methods are defined by control bit SHP, **extended sample mode** and **pulse mode**.

ADC12 Operation

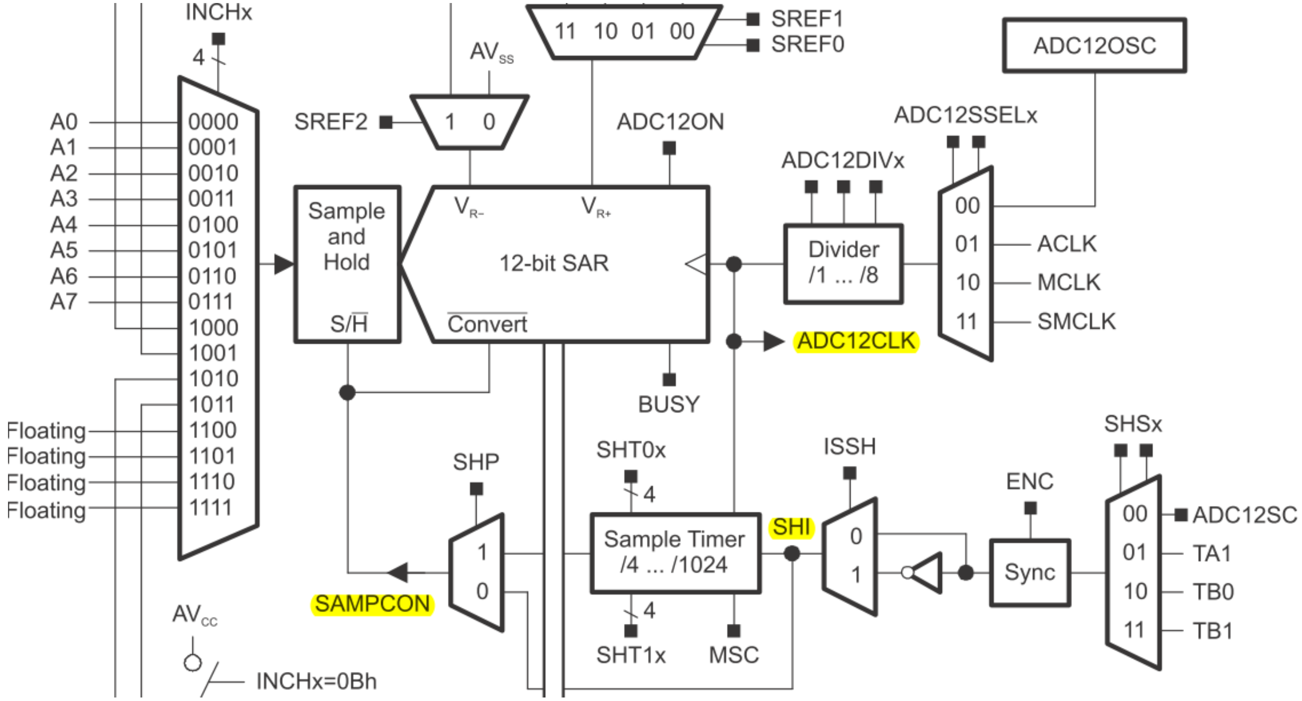
The pulse sample mode is selected when **SHP** = 1.

The **SHI** signal is used to trigger the sampling timer.

The SHT0x and SHT1x bits in ADC12CTL0 control the interval of the sampling timer that defines the SAMPCON sample period t sample .

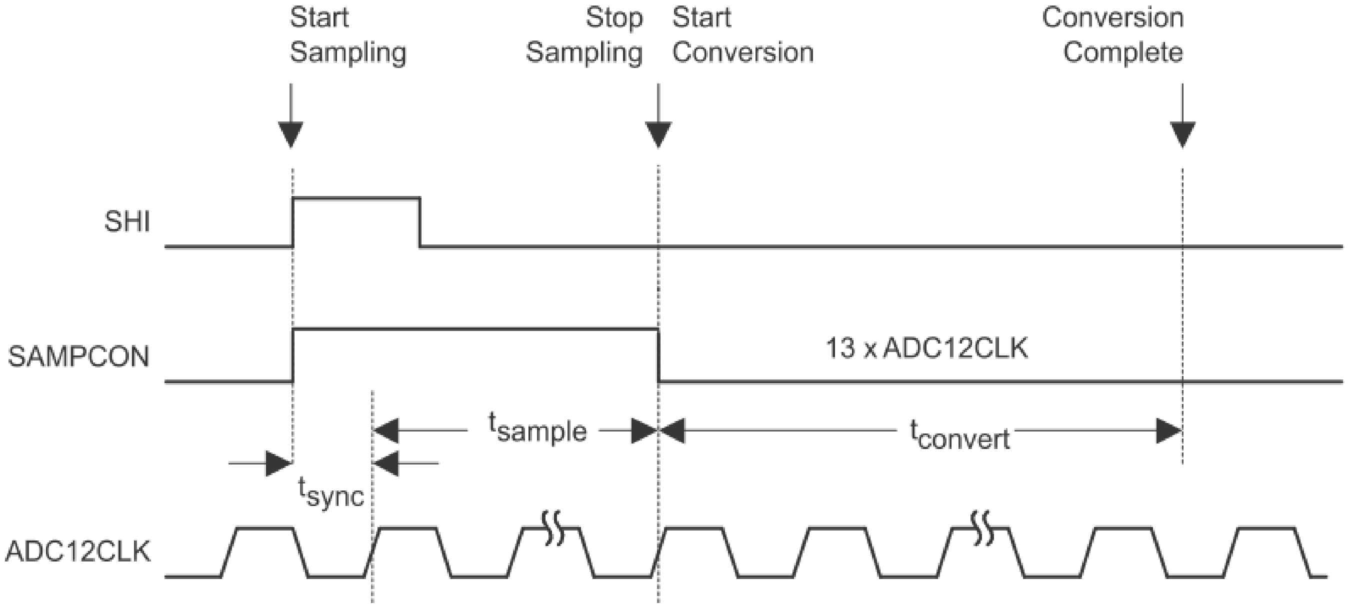


Pulse Sample Mode



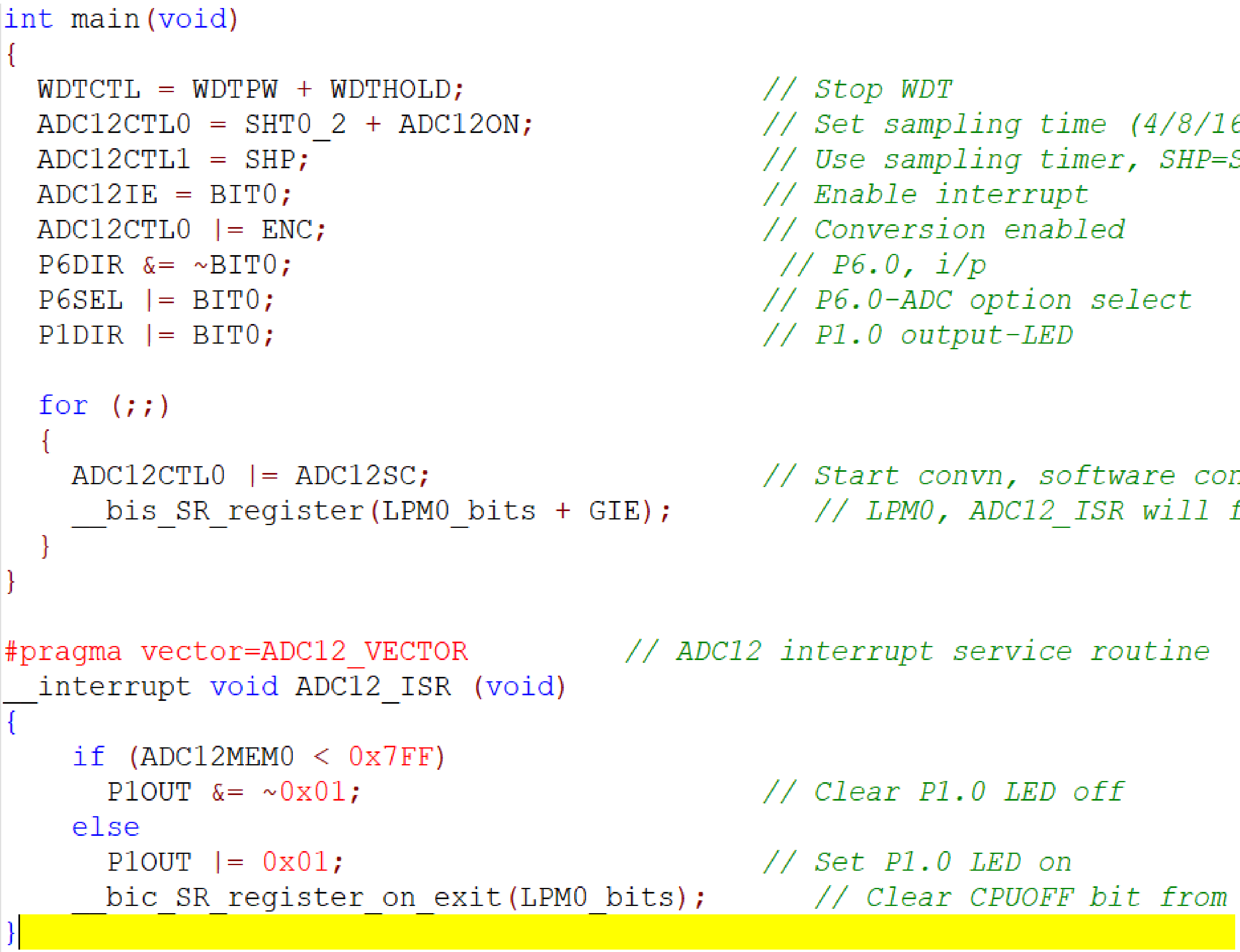
ADC12 Operation

The total sampling time is t_{sample} plus t_{sync}



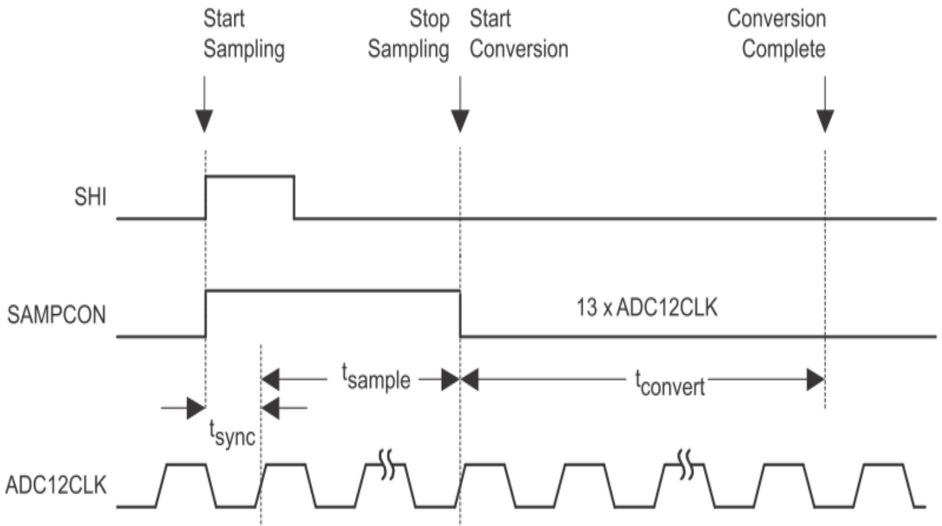
Pulse Sample Mode

ADC Example



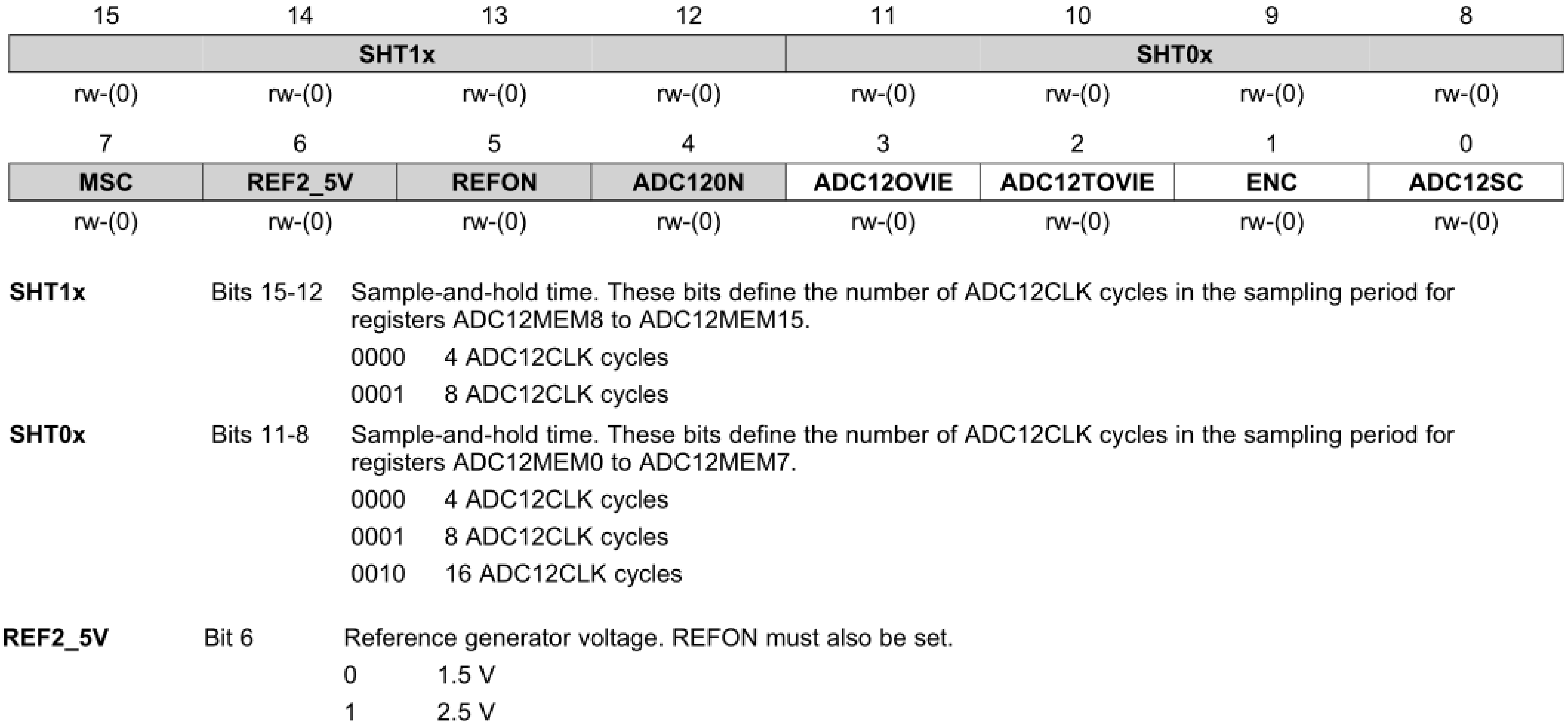
```
int main(void)
{
  WDTCTL = WDTPW + WDTHOLD;                 // Stop WDT
  ADC12CTL0 = SHT0_2 + ADC12ON;             // Set sampling time (4/8/16/64)*ADC12CLK, turn on ADC12
  ADC12CTL1 = SHP;                          // Use sampling timer, SHP=Sample/Hold Pulse Mode
  ADC12IE = BIT0;                           // Enable interrupt
  ADC12CTL0 |= ENC;                         // Conversion enabled
  P6DIR &= ~BIT0;                            // P6.0, i/p
  P6SEL |= BIT0;                            // P6.0-ADC option select
  P1DIR |= BIT0;                            // P1.0 output-LED

  for (;;)
  {
    ADC12CTL0 |= ADC12SC;                   // Start convn, software controlled
    __bis_SR_register(LPM0_bits + GIE);       // LPM0, ADC12_ISR will force exit
  }
}

#pragma vector=ADC12_VECTOR         // ADC12 interrupt service routine
__interrupt void ADC12_ISR (void)
{
    if (ADC12MEM0 < 0x7FF)
      P1OUT &= ~0x01;                       // Clear P1.0 LED off
    else
      P1OUT |= 0x01;                        // Set P1.0 LED on
    __bic_SR_register_on_exit(LPM0_bits);     // Clear CPUOFF bit from
}
```

Pulse Sample Mode



ADC12 Registers

ADC12CTL0



15	14	13	12	11	10	9	8
SHT1x				SHT0x			
rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)

7	6	5	4	3	2	1	0
MSC	REF2_5V	REFON	ADC120N	ADC12OVIE	ADC12TOVIE	ENC	ADC12SC
rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)

SHT1x Bits 15-12 Sample-and-hold time. These bits define the number of ADC12CLK cycles in the sampling period for registers ADC12MEM8 to ADC12MEM15.

0000 4 ADC12CLK cycles

0001 8 ADC12CLK cycles

SHT0x Bits 11-8 Sample-and-hold time. These bits define the number of ADC12CLK cycles in the sampling period for registers ADC12MEM0 to ADC12MEM7.

0000 4 ADC12CLK cycles

0001 8 ADC12CLK cycles

0010 16 ADC12CLK cycles

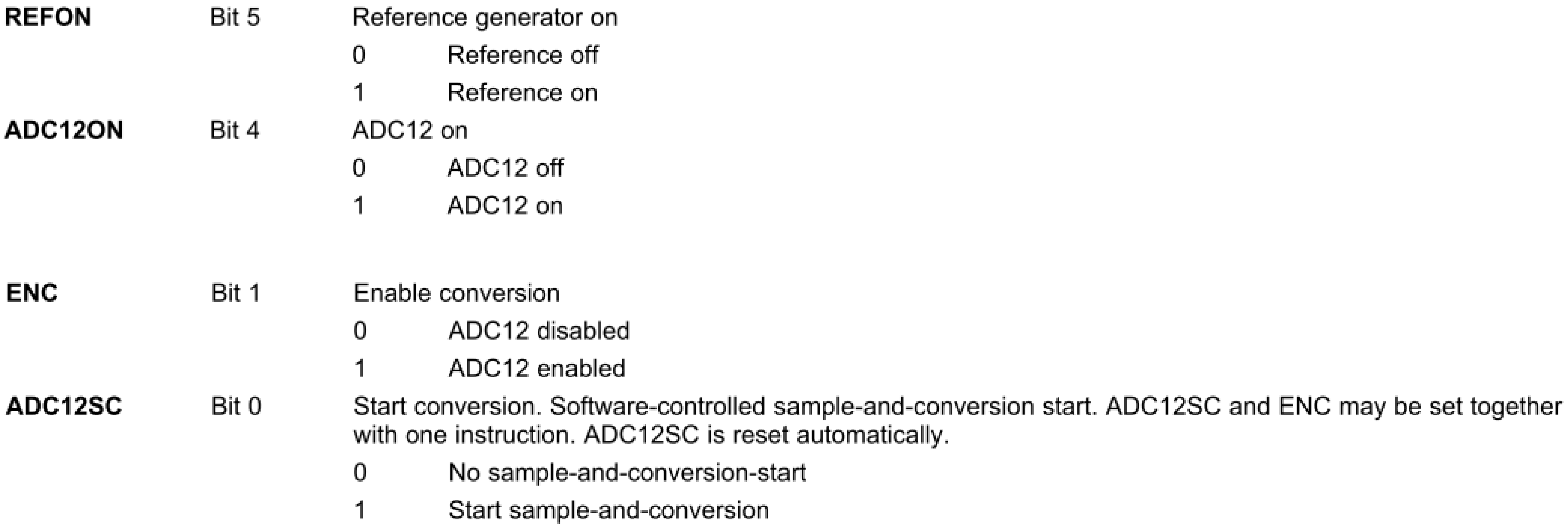
REF2_5V Bit 6 Reference generator voltage. REFON must also be set.

0 1.5 V

1 2.5 V

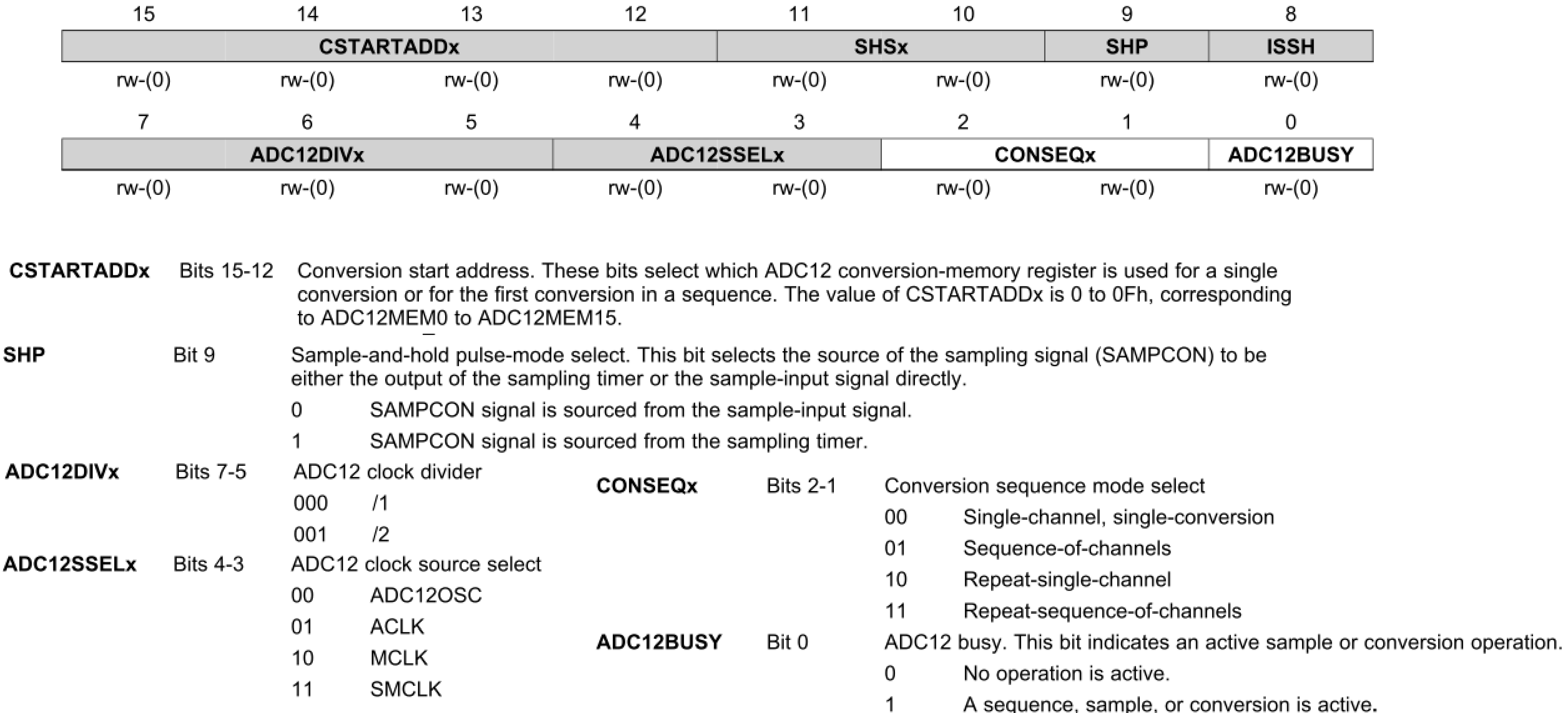
ADC12 Registers

ADC12CTL0



REFON	Bit 5	Reference generator on	
		0	Reference off
		1	Reference on
ADC12ON	Bit 4	ADC12 on	
		0	ADC12 off
		1	ADC12 on
ENC	Bit 1	Enable conversion	
		0	ADC12 disabled
		1	ADC12 enabled
ADC12SC	Bit 0	Start conversion. Software-controlled sample-and-conversion start. ADC12SC and ENC may be set together with one instruction. ADC12SC is reset automatically.	
		0	No sample-and-conversion-start
		1	Start sample-and-conversion

ADC12CTL1



15	14	13	12	11	10	9	8
CSTARTADDx				SHSx		SHP	ISSH
rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)

7	6	5	4	3	2	1	0
ADC12DIVx			ADC12SSELx		CONSEQx		ADC12BUSY
rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)	rw-(0)

CSTARTADDx Bits 15-12 Conversion start address. These bits select which ADC12 conversion-memory register is used for a single conversion or for the first conversion in a sequence. The value of CSTARTADDx is 0 to 0Fh, corresponding to ADC12MEM0 to ADC12MEM15.

SHP Bit 9 Sample-and-hold pulse-mode select. This bit selects the source of the sampling signal (SAMPCON) to be either the output of the sampling timer or the sample-input signal directly.

0	SAMPCON signal is sourced from the sample-input signal.
1	SAMPCON signal is sourced from the sampling timer.

ADC12DIVx Bits 7-5 ADC12 clock divider

000	/1
001	/2

ADC12SSELx Bits 4-3 ADC12 clock source select

00	ADC12OSC
01	ACLK
10	MCLK
11	SMCLK

CONSEQx Bits 2-1 Conversion sequence mode select

00	Single-channel, single-conversion
01	Sequence-of-channels
10	Repeat-single-channel
11	Repeat-sequence-of-channels

ADC12BUSY Bit 0 ADC12 busy. This bit indicates an active sample or conversion operation.

0	No operation is active.
1	A sequence, sample, or conversion is active.