GUC German University in Cairo الجامعة الألمانية بالقاهرة

Embedded System Architecture - CSEN 701

Module 3: *Embedded Hardware*

Lecture 06: *Sensors Fundamentals Cont. & ADC*

Dr. Eng. Catherine M. Elias

catherine.elias@guc.edu.eg

Lecturer, Computer Science and Engineering,
Faculty of Media Engineering and Technology, German University in Cairo

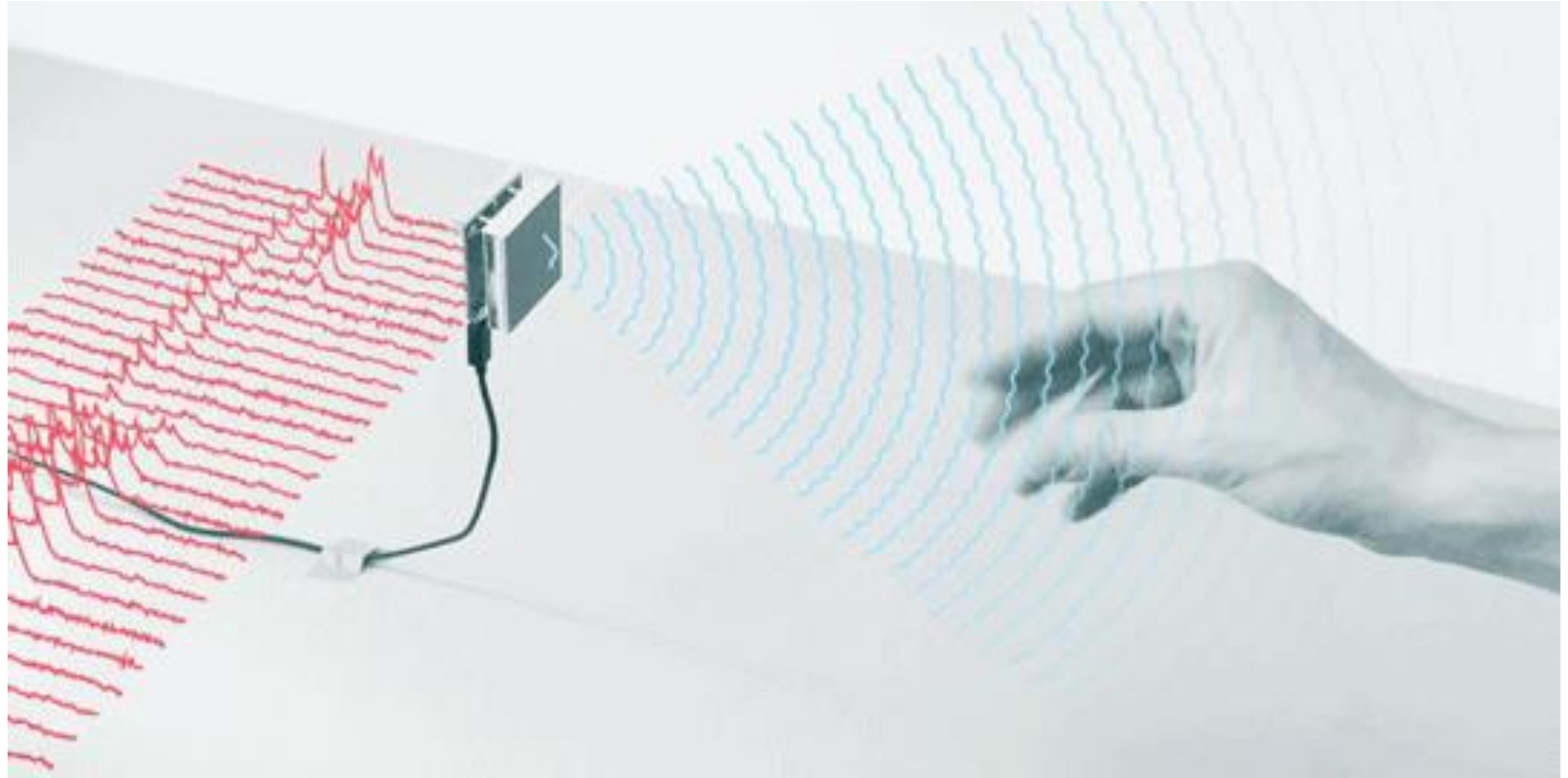
© Dr. Eng. Catherine M. Elias

Outline

- Sensors
- Interface Techniques

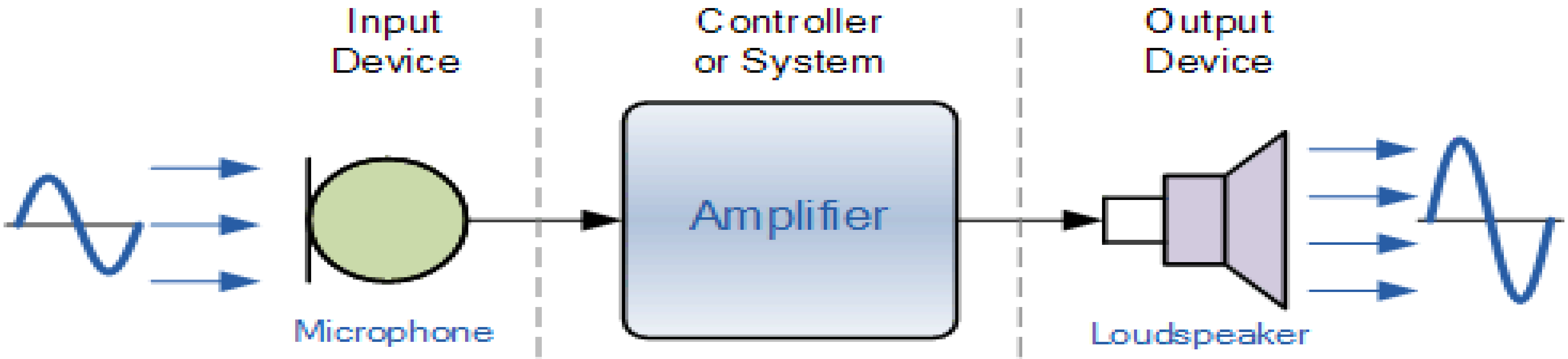
What is a sensor?

- A sensor is a device or component that is used to detect or measure physical properties or changes in the environment and convert them into electrical signals or other readable outputs.
- Sensors are widely used in various fields, including electronics, engineering, automation, and science, to gather data and enable control systems to respond to changes in their surroundings.

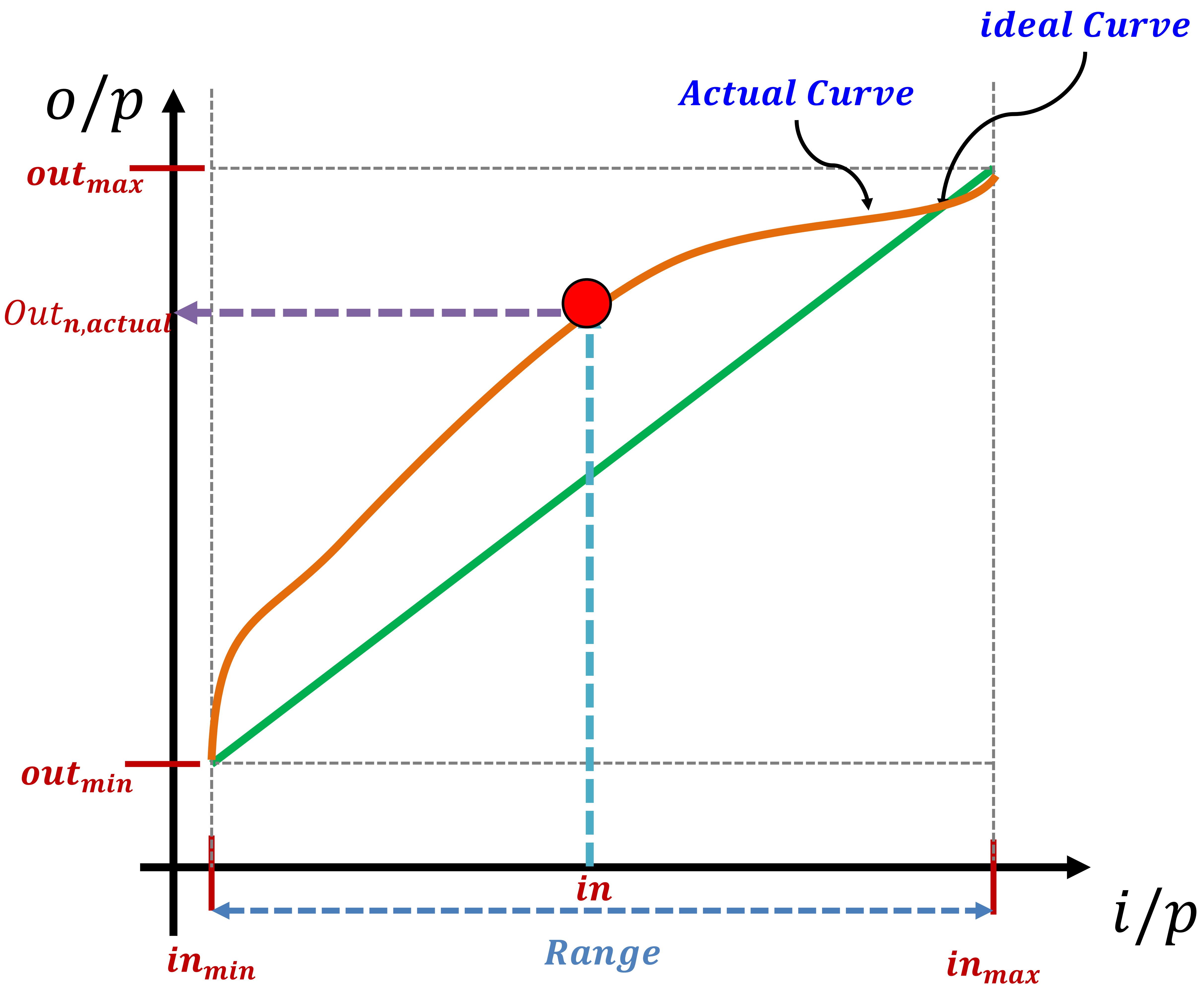


The Process

Detection → Transduction → Output

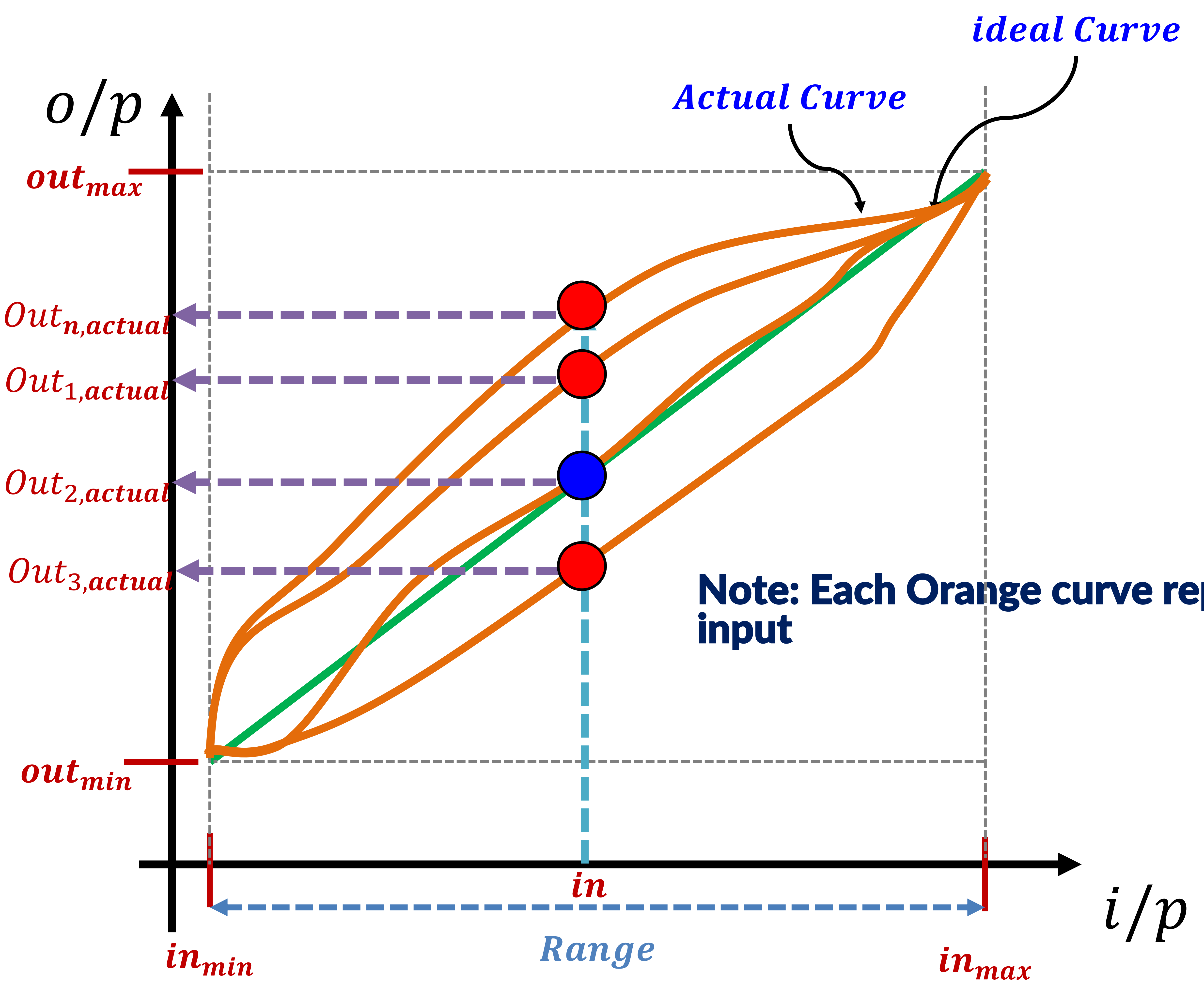


Sensor Characteristics: *Calibration Curve*



German University in Cairo
الجامعة الألمانية بالقاهرة

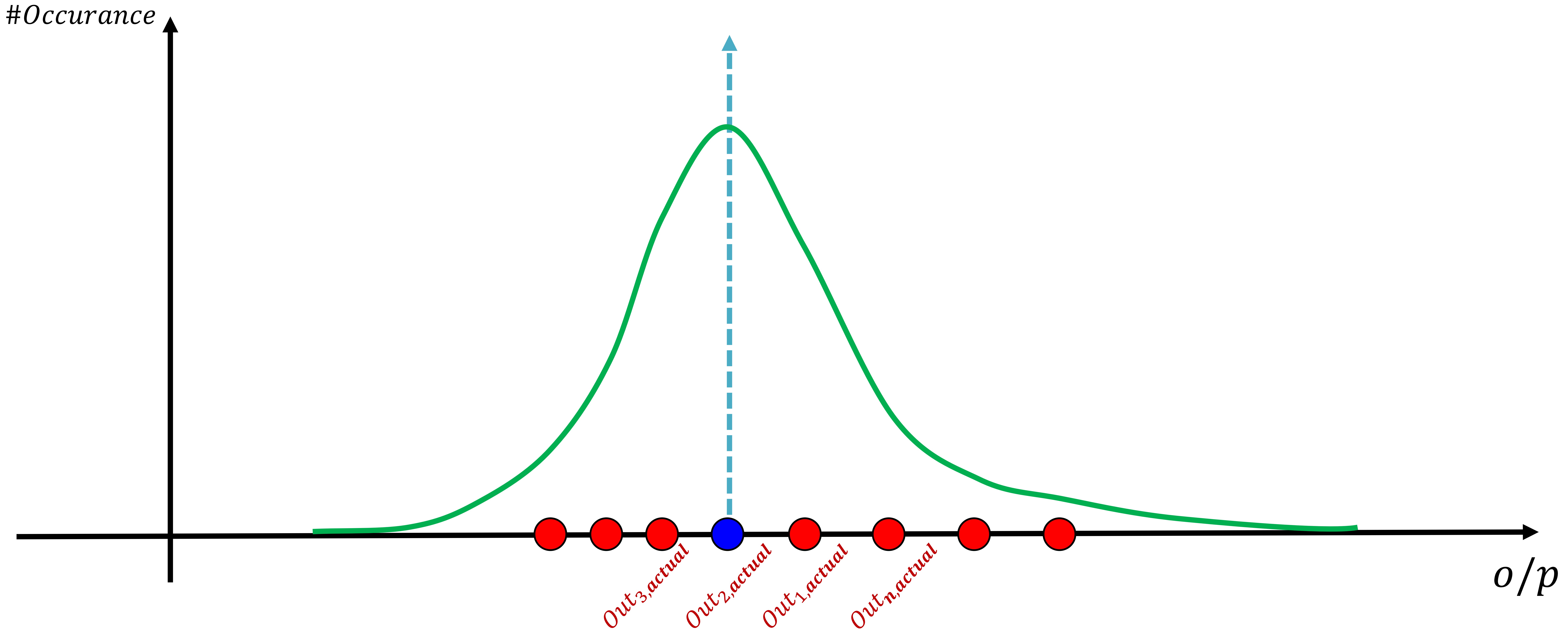
Sensor Characteristics: *Accuracy and Precision*



Note: Each Orange curve represents a reading fixing the input

Sensor Characteristics: *Gaussian Distribution Curve*

Repeat the same experiment for 10000 times and take the same measurement then count the histogram (#Occurance)



Sensor Characteristics: Mean and Standard Deviation

- The **mean** of any set of data, the **mean** can be calculated as:

$$\mu = \bar{X} = \frac{1}{N}\sum_{n=1}^{N} X_{n,meas}$$

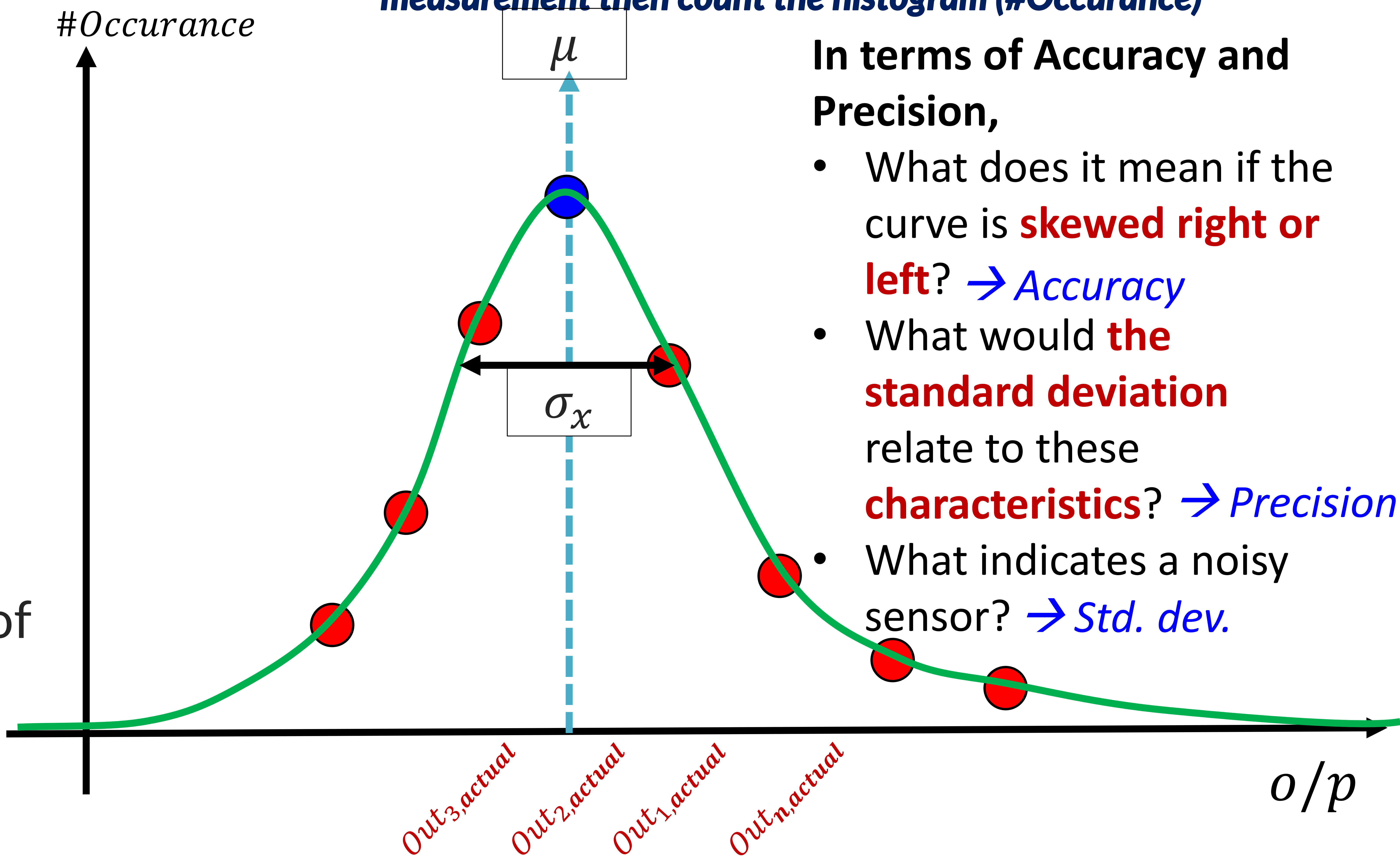
- While the **variance** can be calculated from:

$$\sigma_x^2 = \frac{1}{N}\sum_{n=1}^{N} (X_n - \bar{X})^2$$

- The **std. dev.** is the square-root of the variance.

$$\sigma_x = \sqrt{\sigma_x^2}$$

Repeat the same experiment for 10000 times and take the same measurement then count the histogram (#Occurance)



In terms of Accuracy and Precision,

- What does it mean if the curve is **skewed right or left**? → *Accuracy*
- What would **the standard deviation** relate to these **characteristics**? → *Precision*
- What indicates a noisy sensor? → *Std. dev.*

Sensor Characteristics: *Resolution*

- Resolution refers to the **smallest detectable change** in the property being measured.
- **High-resolution** sensors can detect **small changes** with **precision**.

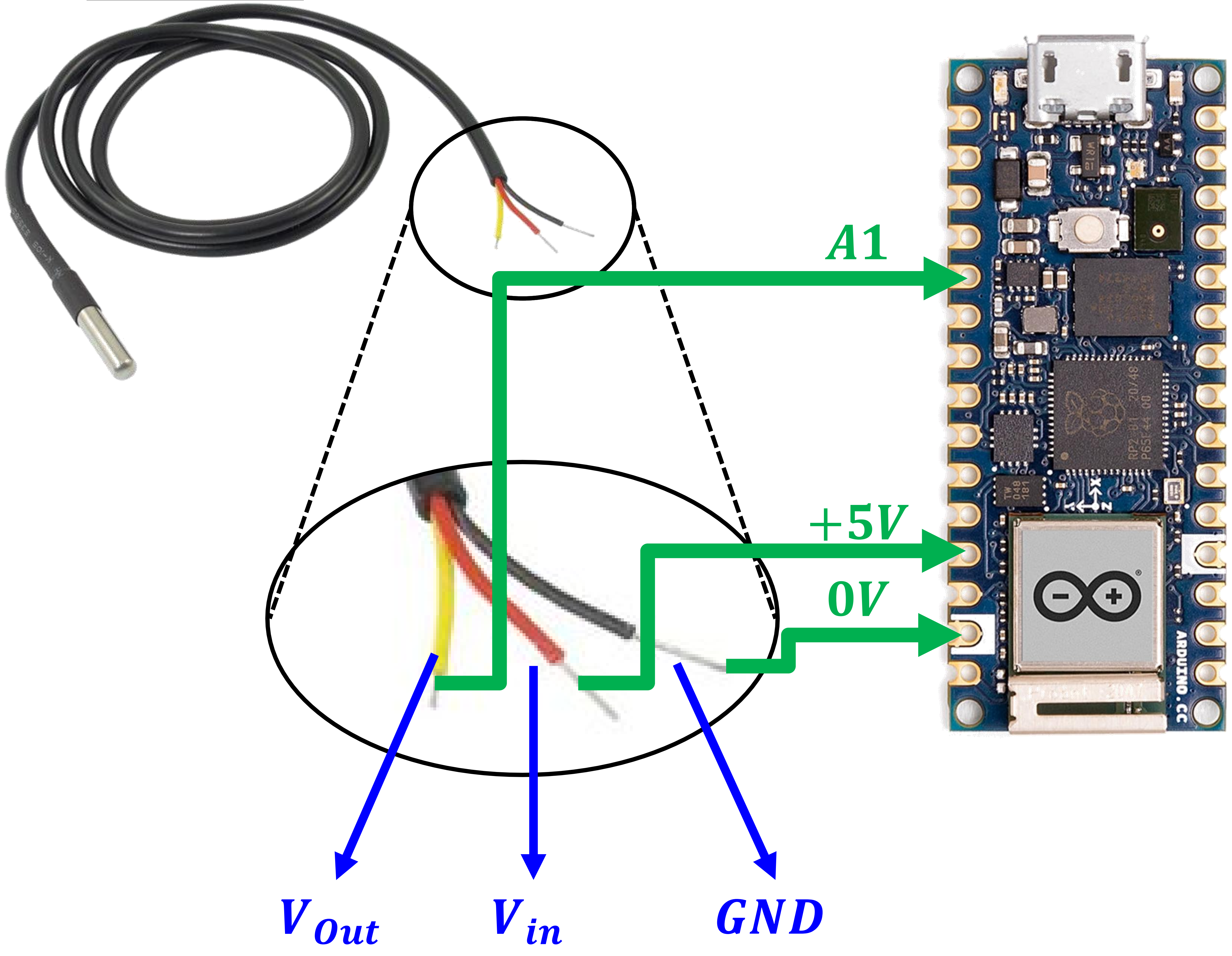
- The sensor **resolution** is highly coupled with the previously discussed concept of **signals**.
- It is also coupled with the **microcontroller** that processes the measured signal by the sensor.
- In order to understand the resolution, let's discuss one important peripheral (interface technique) in any microcontroller which is the **Analog-to-Digital Converter (ADC).**

Analog-to-Digital Converter (ADC)

- It is an electronic component or subsystem that is used to **convert analog signals into digital representations**.
- In other words, an ADC takes continuous or analog voltage or current inputs and transforms them into discrete digital values that can be **processed, stored, or manipulated** by digital systems, such as **microcontrollers**, computers, and digital signal processors (DSPs).

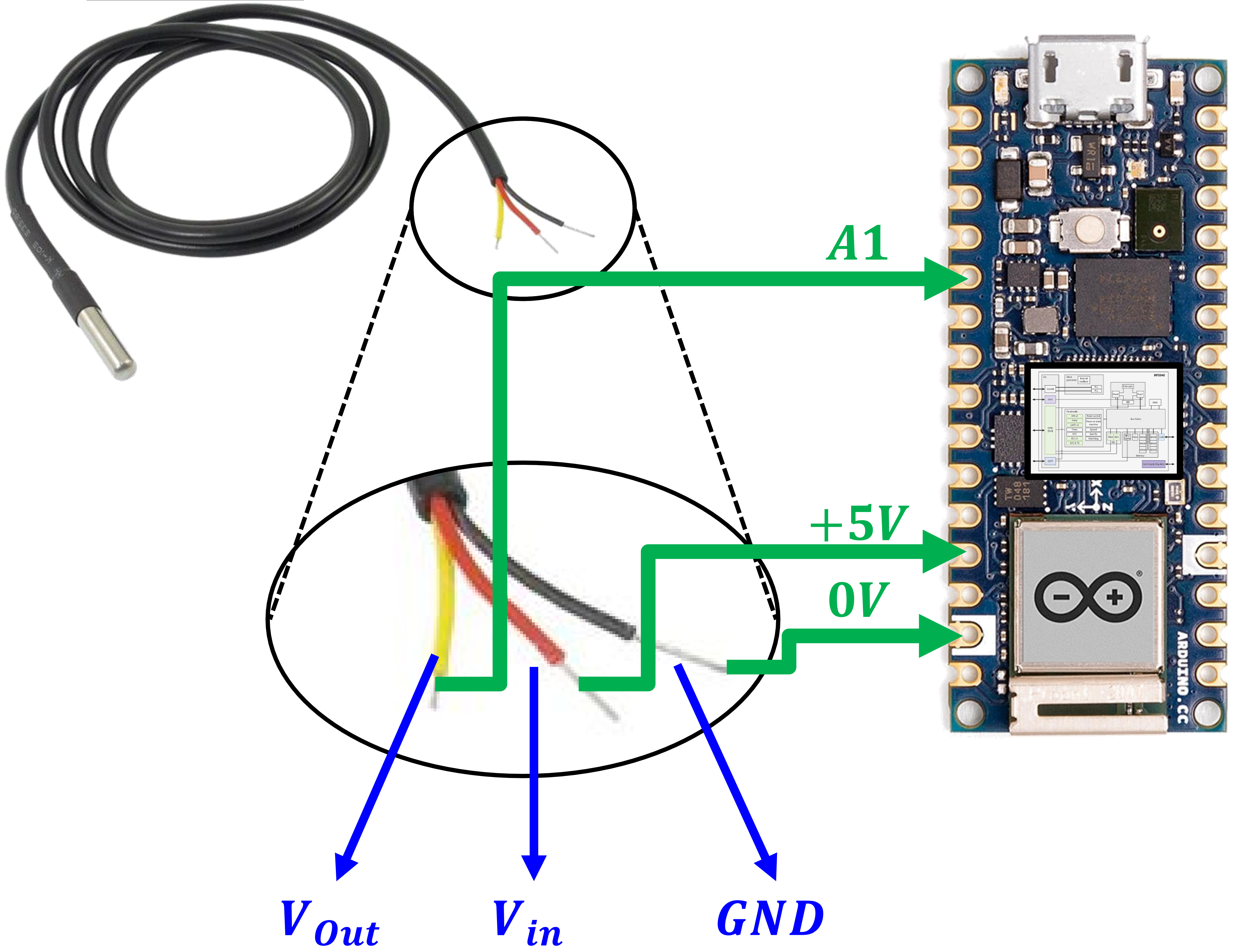
Analog-to-Digital Converter (ADC): *The Process*

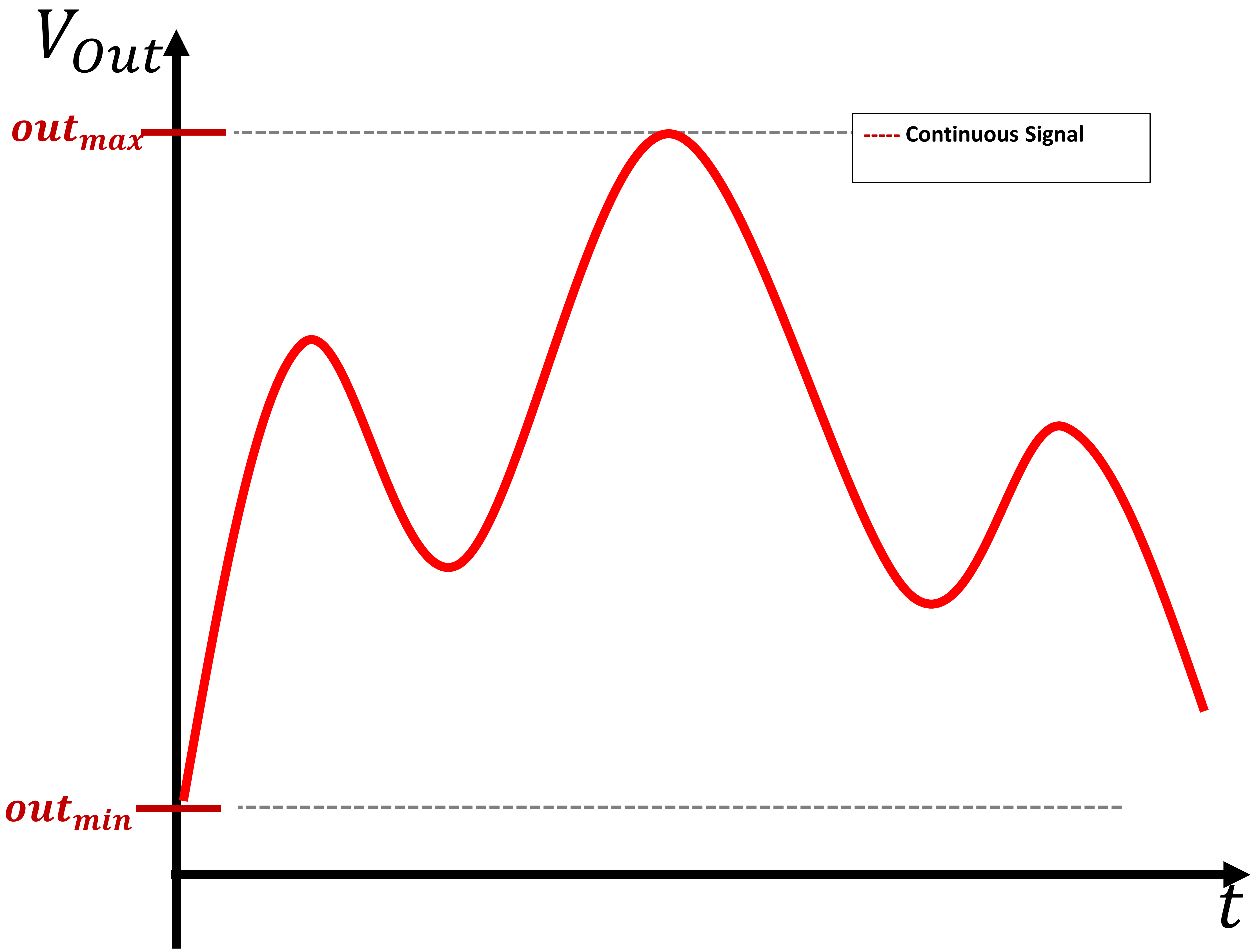
- ***Stage 1:*** Interface with the microcontroller



Analog-to-Digital Converter (ADC): *The Process*

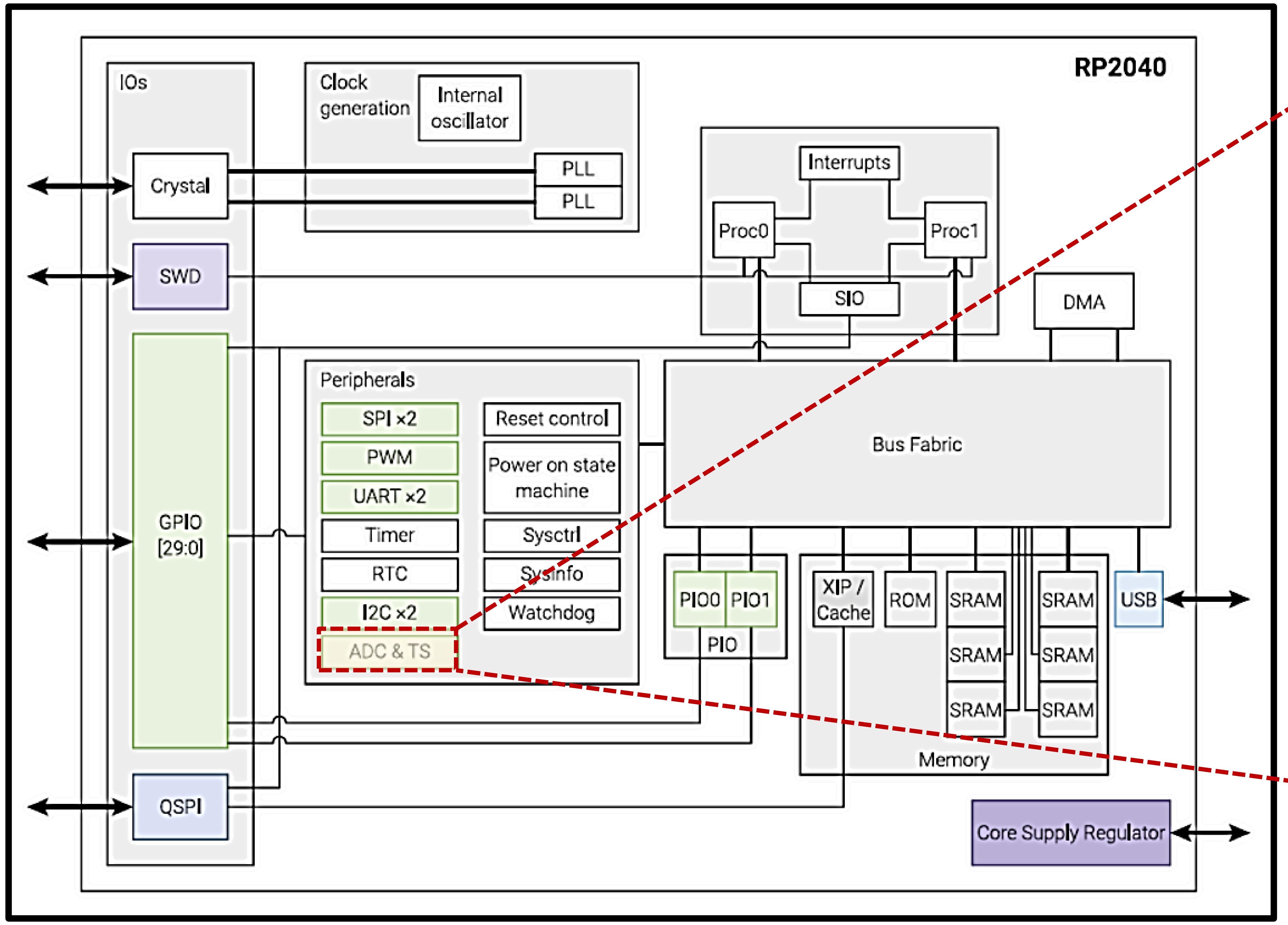
- ***Stage 2:*** Continuous Sensory Data Acquiring

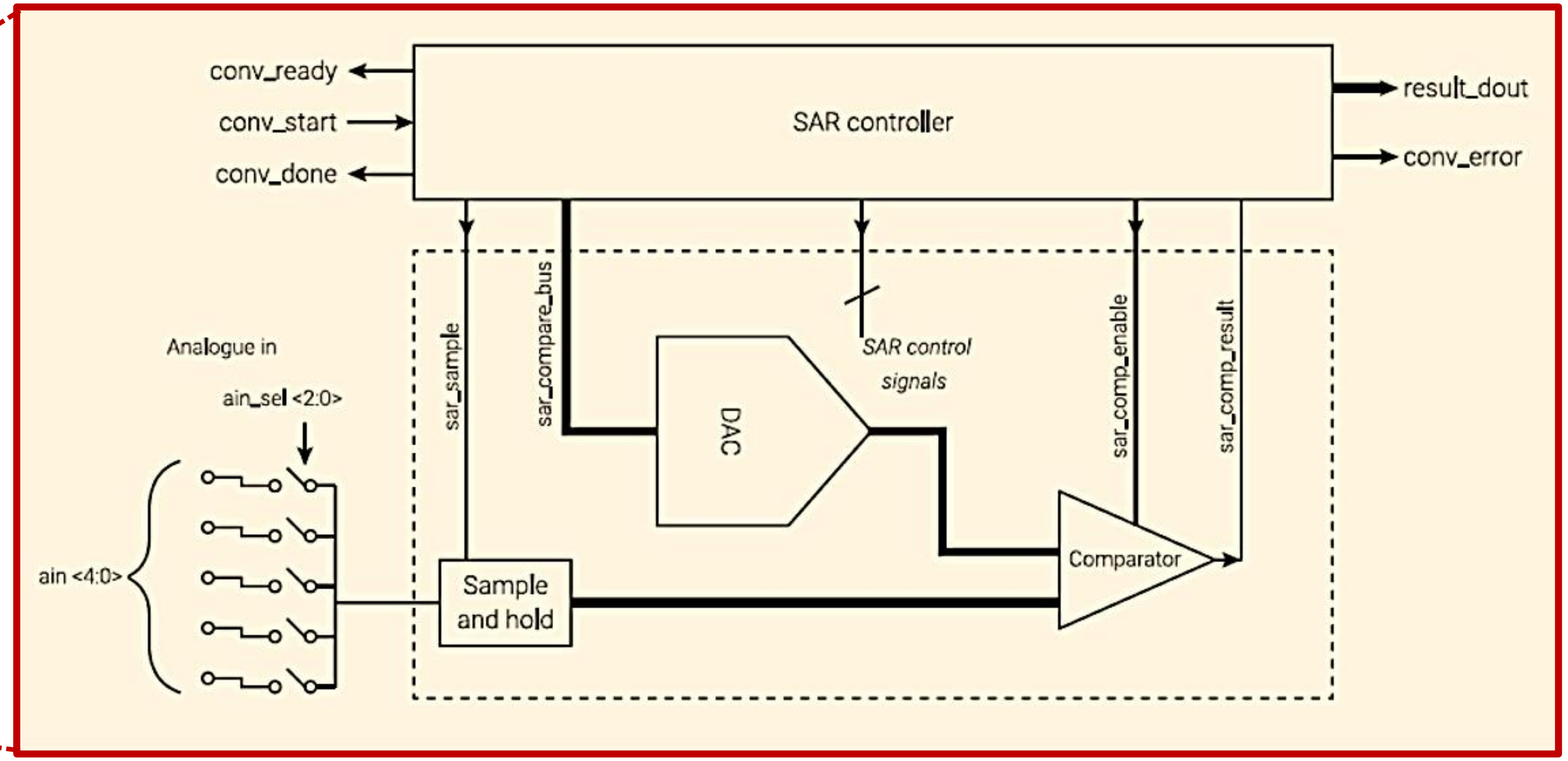




Analog-to-Digital Converter (ADC): *The Process*

- ***Stage 3:*** The ADC Process using Successive Approximation Register





Analog-to-Digital Converter (ADC): *The Process*

Sampling

The first step in the conversion process is sampling.

The ADC continuously measures the analog signal at specific time intervals.

Each measurement, or sample, represents the value of the analog signal at a particular point in time.

Quantization

After sampling, the analog voltage is quantized, which means it is mapped to a finite number of discrete digital values.

The number of possible digital values depends on the resolution of the ADC.

A higher-resolution ADC can represent the analog signal with more detail.

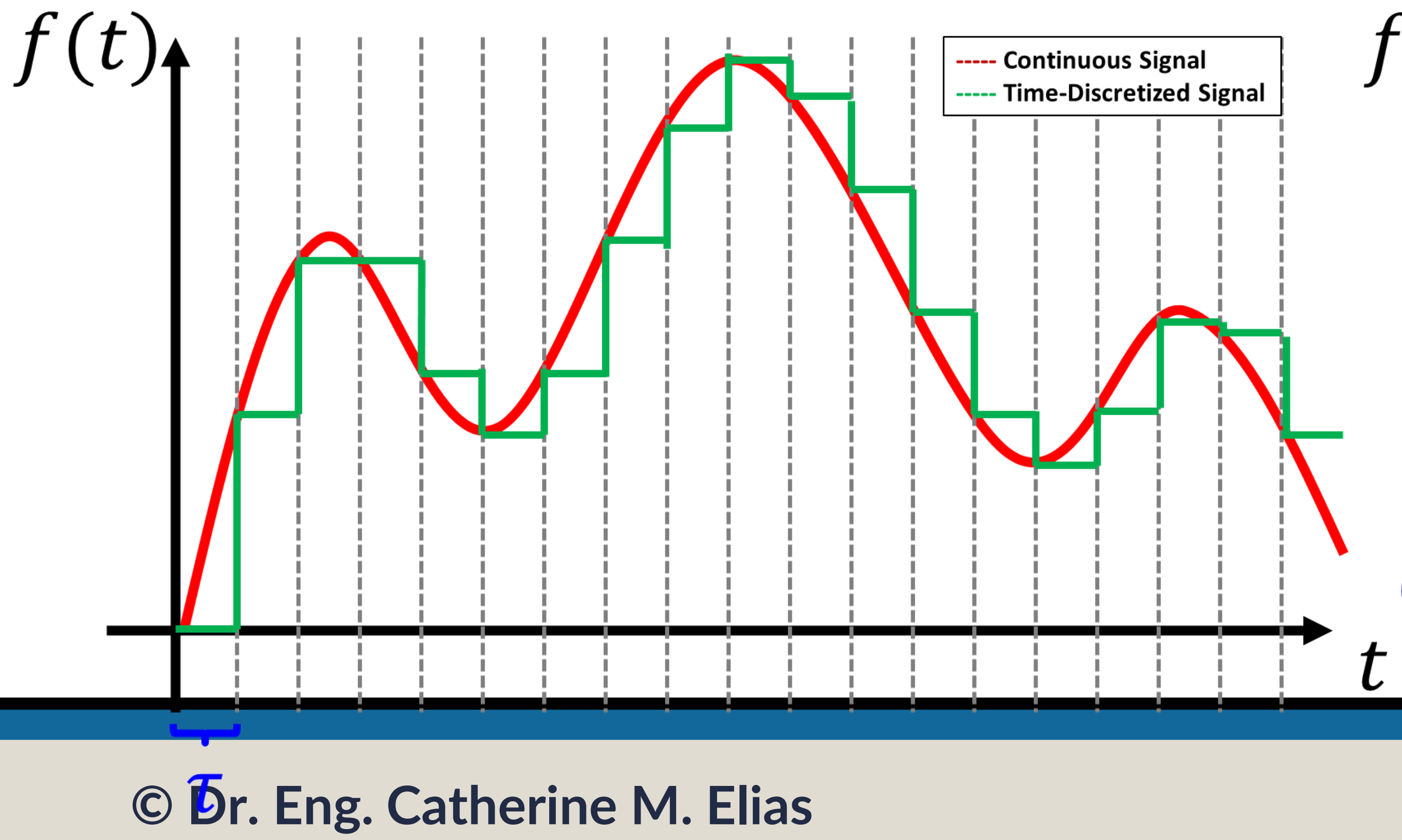
Conversion

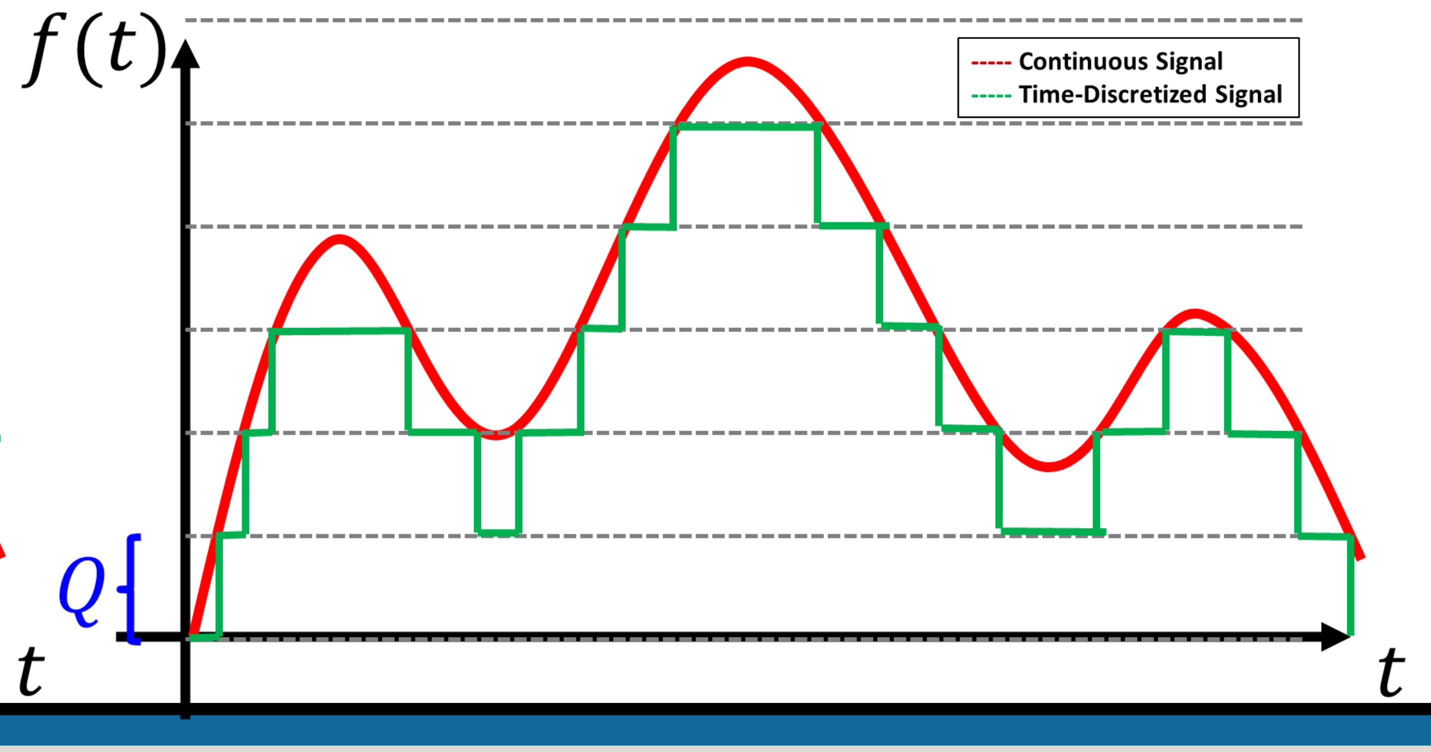
The ADC assigns a digital code to each sampled analog value. The digital code is usually binary and corresponds to the amplitude of the analog signal at that moment in time.

Output

The digital code is then made available as an output, often as a binary number or a series of binary digits (bits).

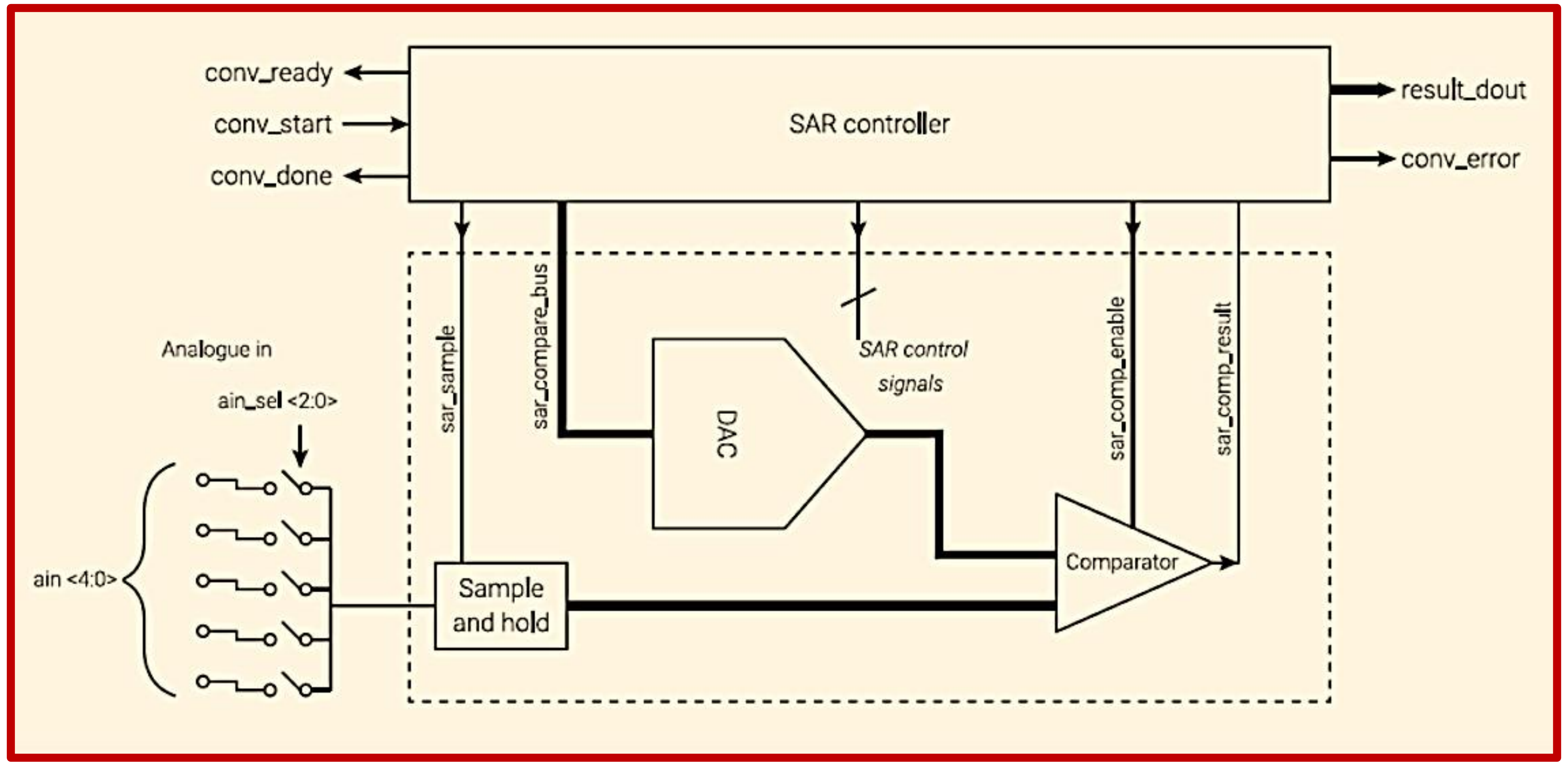
This digital representation can be further processed, analyzed, or transmitted by digital devices.





Analog-to-Digital Converter (ADC): *The Process*

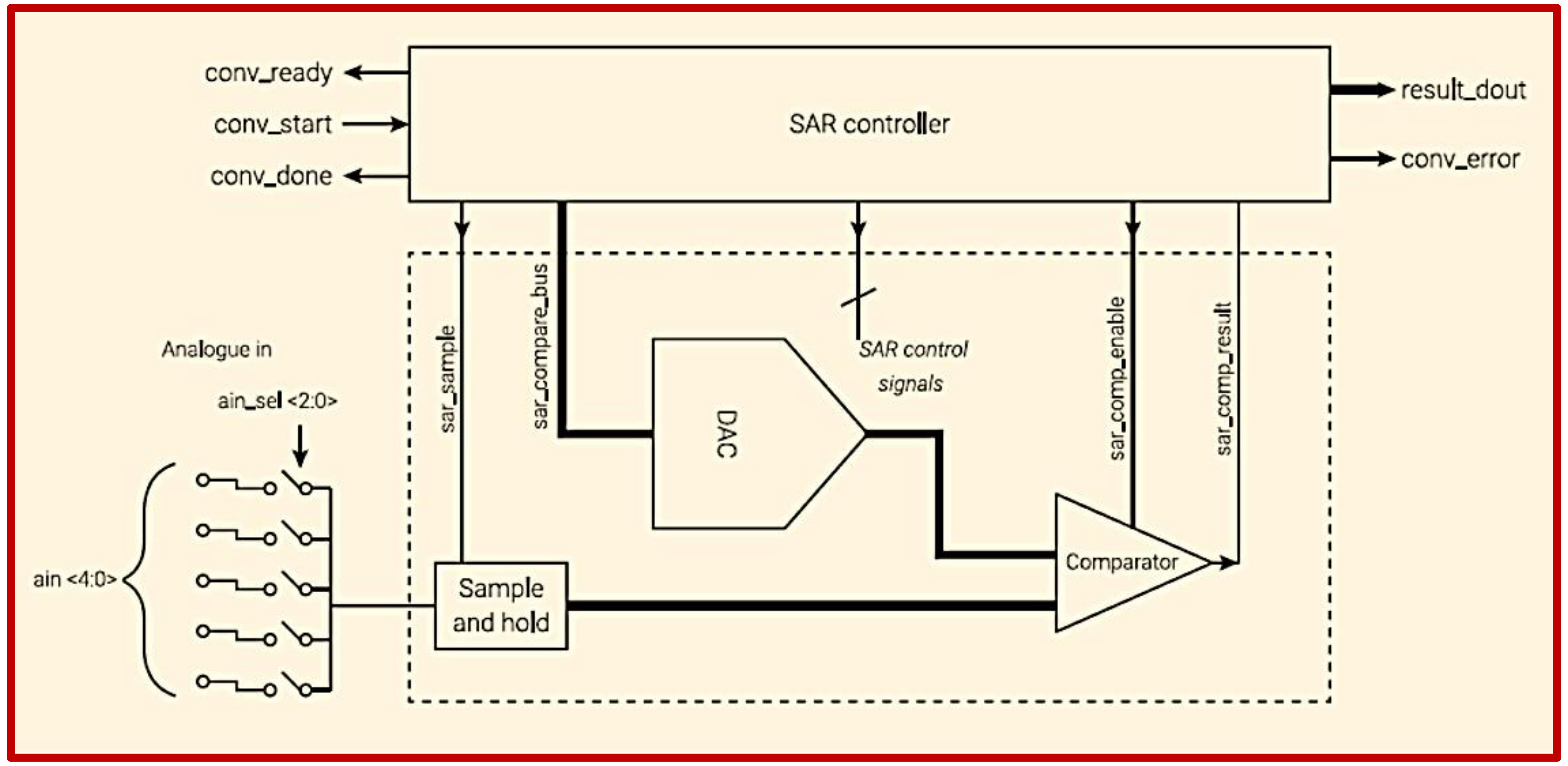
- ***Stage 3:*** The ADC Process using Successive Approximation Register



- ***The Goal:***
  - Knowing that the voltage range of the µC is $[V_{min}, V_{max}]$.
  - Knowing that it is a n-bit ADC
  - Find:
    - The digital value of V_{in}
    - The resolution of the sensor

Analog-to-Digital Converter (ADC): *The Process*

- ***Stage 3:*** The ADC Process using Successive Approximation Register



- ***Example:***
  - ➢ Knowing that the voltage range of the µC is $[0,10]\ Volt$.
  - ➢ Knowing that it is a 8-bit ADC
  - ➢ Find:
    - ▪ The digital value of $V_{in} = 7.65\ Volt$
    - ▪ The resolution of the sensor

But first, Let's see the story….

GUC German University in Cairo الجامعة الألمانية بالقاهرة

Analog-to-Digital Converter (ADC): *The Story*

- For a any microcontroller that is powered by a maximum voltage of$V_{max}\ volt$:

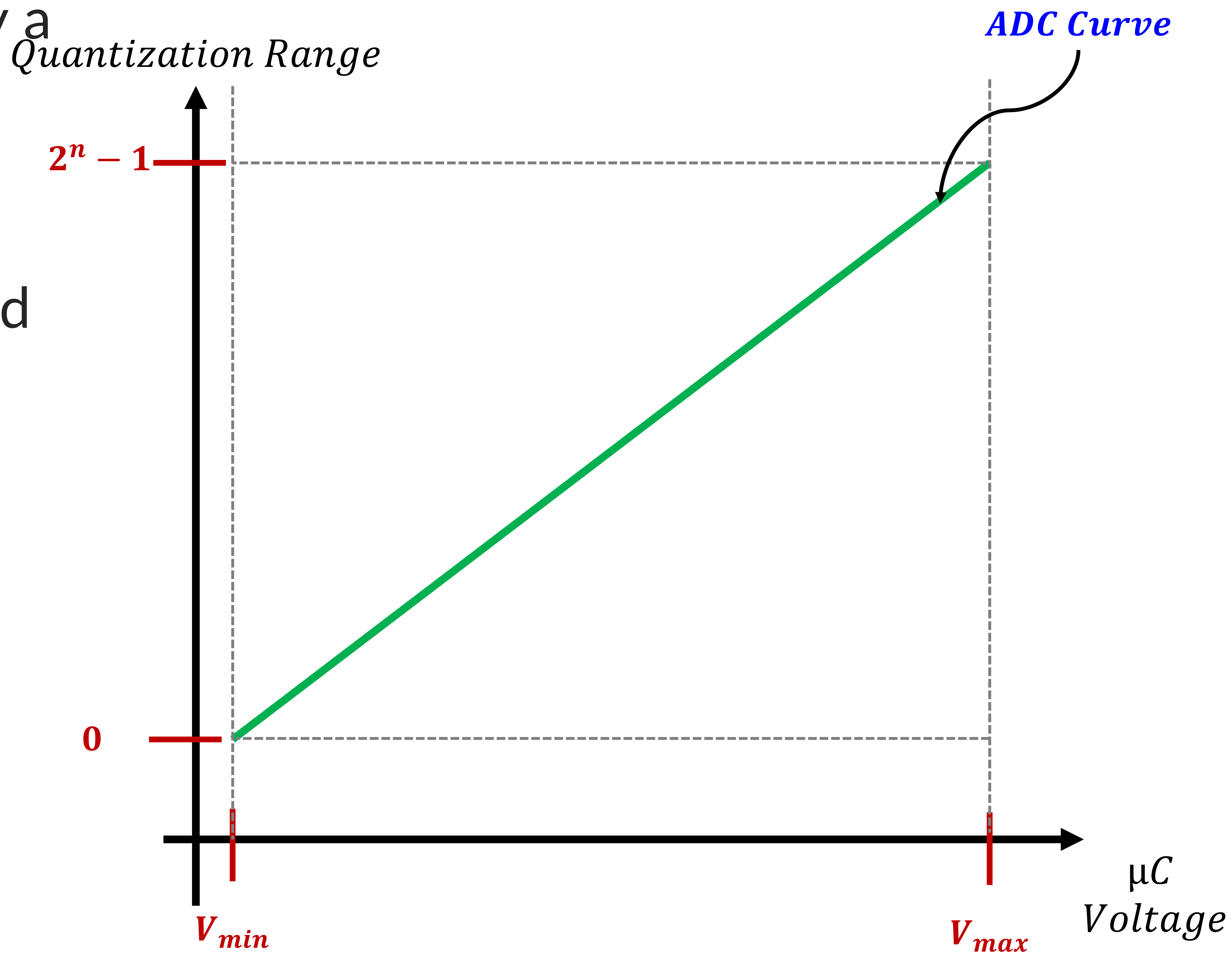
$$V_{range} = [V_{min}, V_{max}]volt$$

- Knowing that the microcontroller is equipped with a n-bit ADC convertor:

$$digital_{range} = [0\ , 2^n - 1]$$

- The resolution can be calculated as:

$$Resolution = \frac{\Delta V_{range}}{\Delta digital_{range}} = \frac{V_{max} - V_{min}}{2^n - 1 - 0}$$

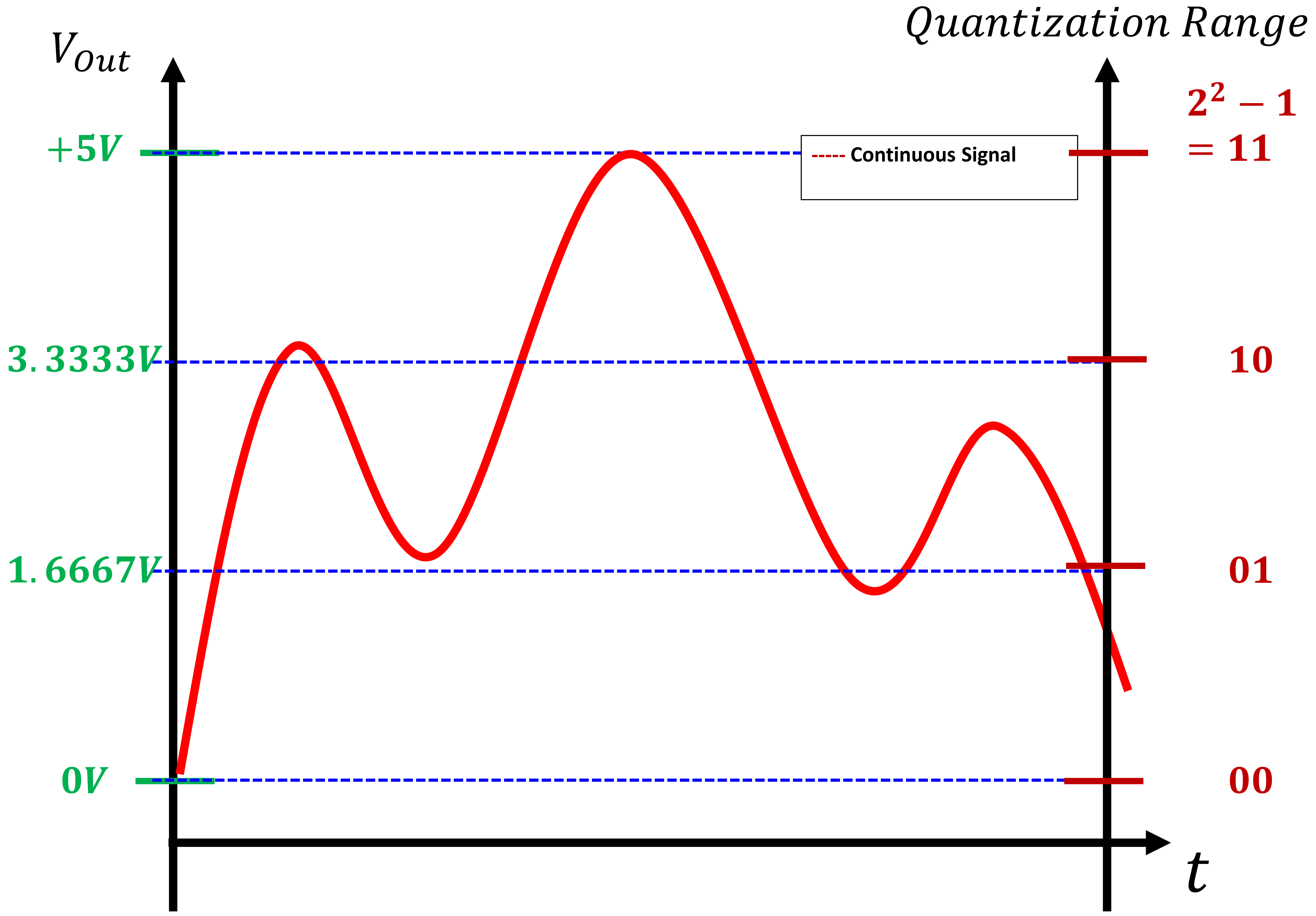


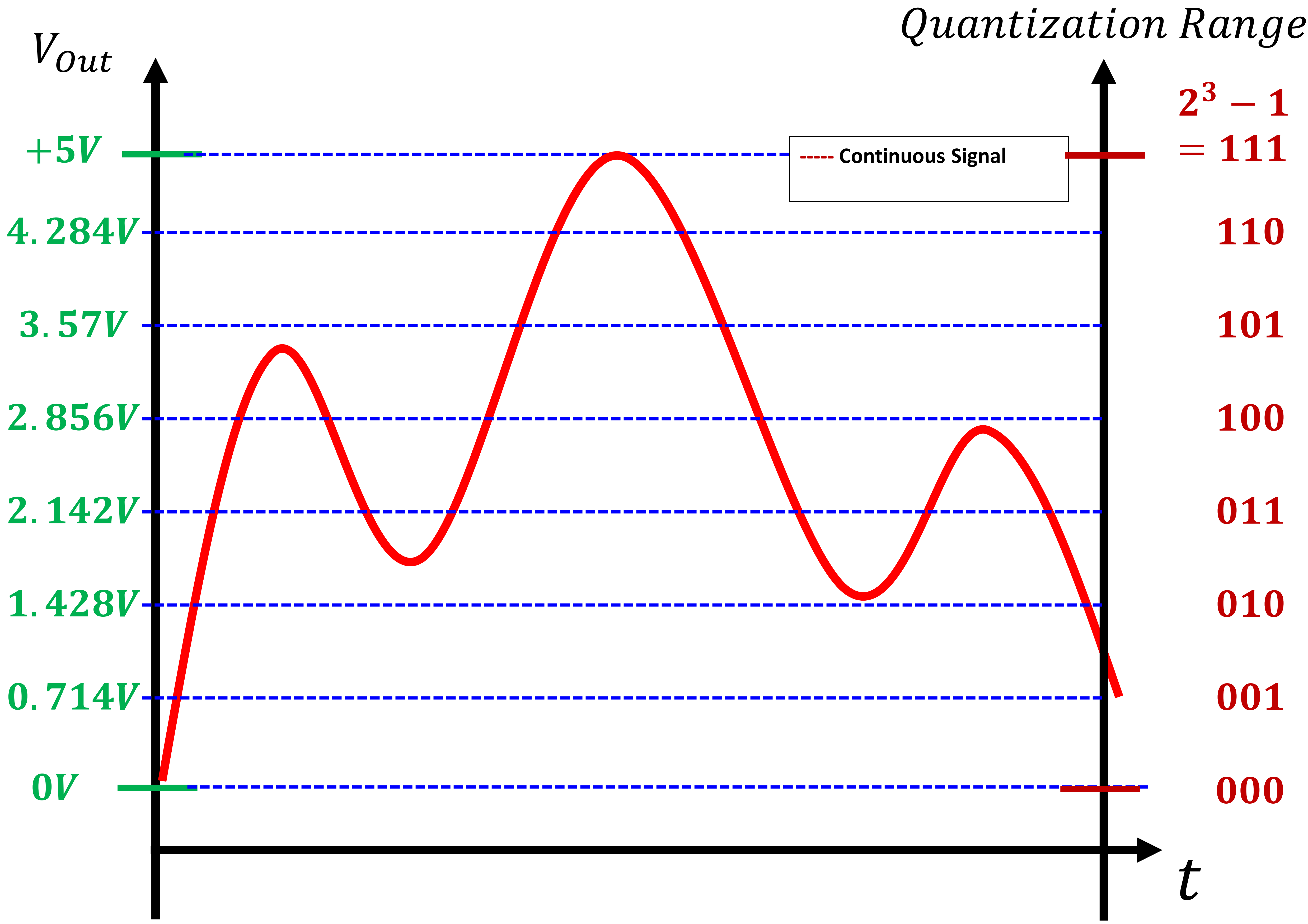
Analog-to-Digital Converter (ADC): *The Story*

How to do the quantization??

2-bits ADC: $$Res = \frac{5}{2^2 - 1} = 1.6667V$$

3-bits ADC: $$Res = \frac{5}{2^3 - 1} = 0.714V$$



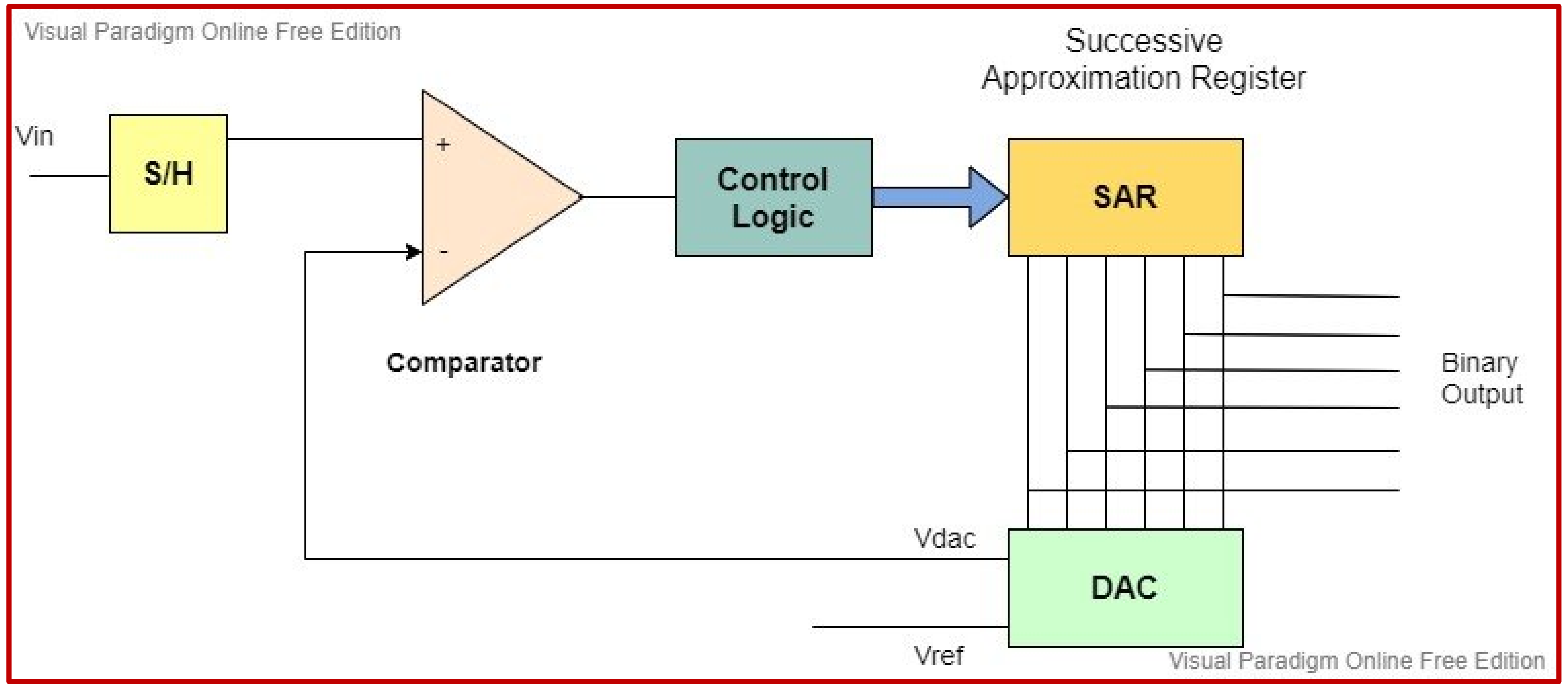


GUC German University in Cairo
الجامعة الألمانية بالقاهرة

Analog-to-Digital Converter (ADC): *The Process*

- ***Stage 3:*** The ADC Process using Successive Approximation Register

- ***Example:***
  - Knowing that the voltage range of the µC is $[0,10]\ Volt$.
  - Knowing that it is a 8-bit ADC
  - Find:
    - The digital value of $V_{in} = 7.65\ Volt$
    - The resolution of the sensor



$$Res = \frac{10}{2^8 - 1} = 0.392V$$

Now, let's see behind the scenes…

GUC German University in Cairo الجامعة الألمانية بالقاهرة

Analog-to-Digital Converter (ADC): *Behind The Scene*

The Successive Approximation Register

- For an 8-bit ADC,
- The process starts from the MSB till reaching the LSB

MSB	→	**LSB**

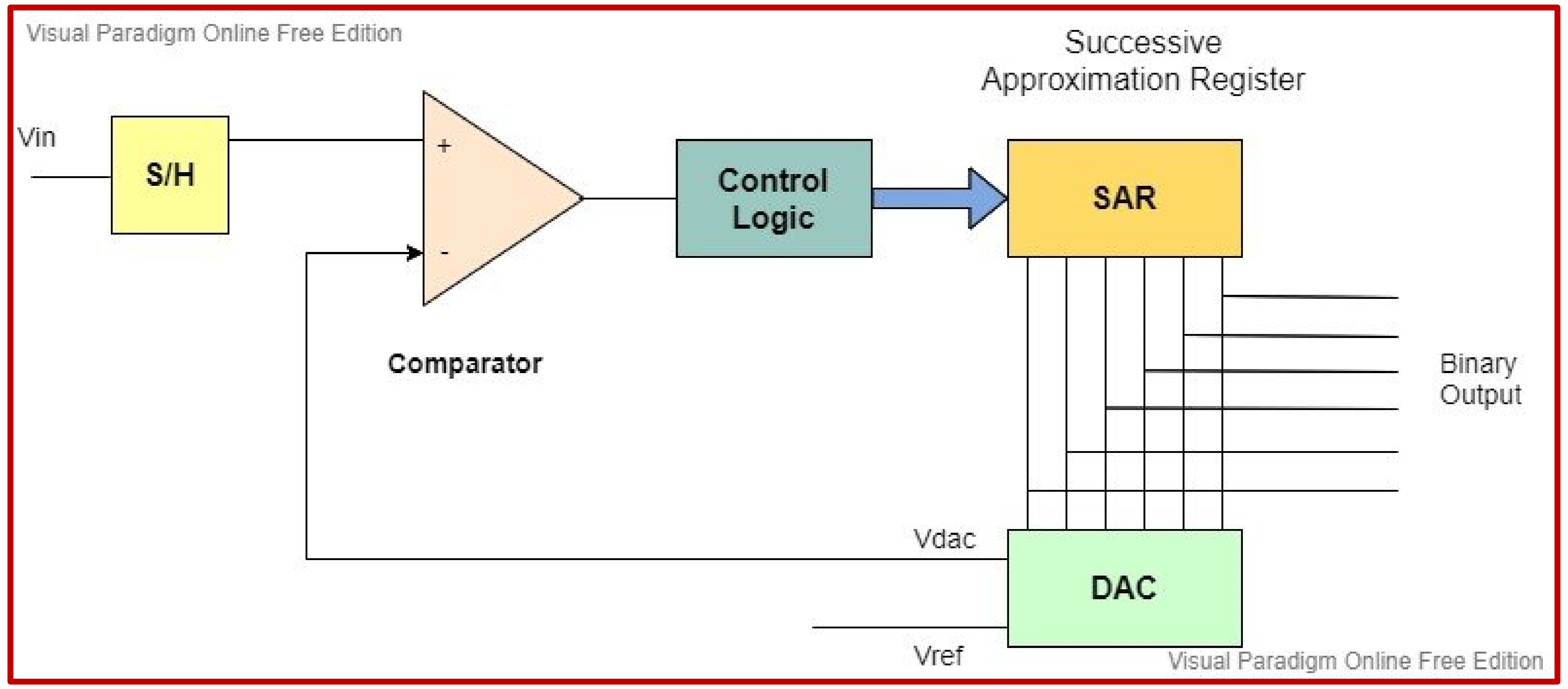
- For bit B_i, we calculate:

$$V_{avg_i} = \frac{V_{avg\_untog} + V_{avg_{i+1}}}{2}$$

V_{avg_untog} is the last untoggled average voltage

- and compare it V_{in} with V_{avg_i}

$$B_i = \begin{cases} 1, & V_{in} > V_{avg_i} \\ 0, & V_{in} > V_{avg_i} \end{cases}$$



Analog-to-Digital Converter (ADC): *Behind The Scene*

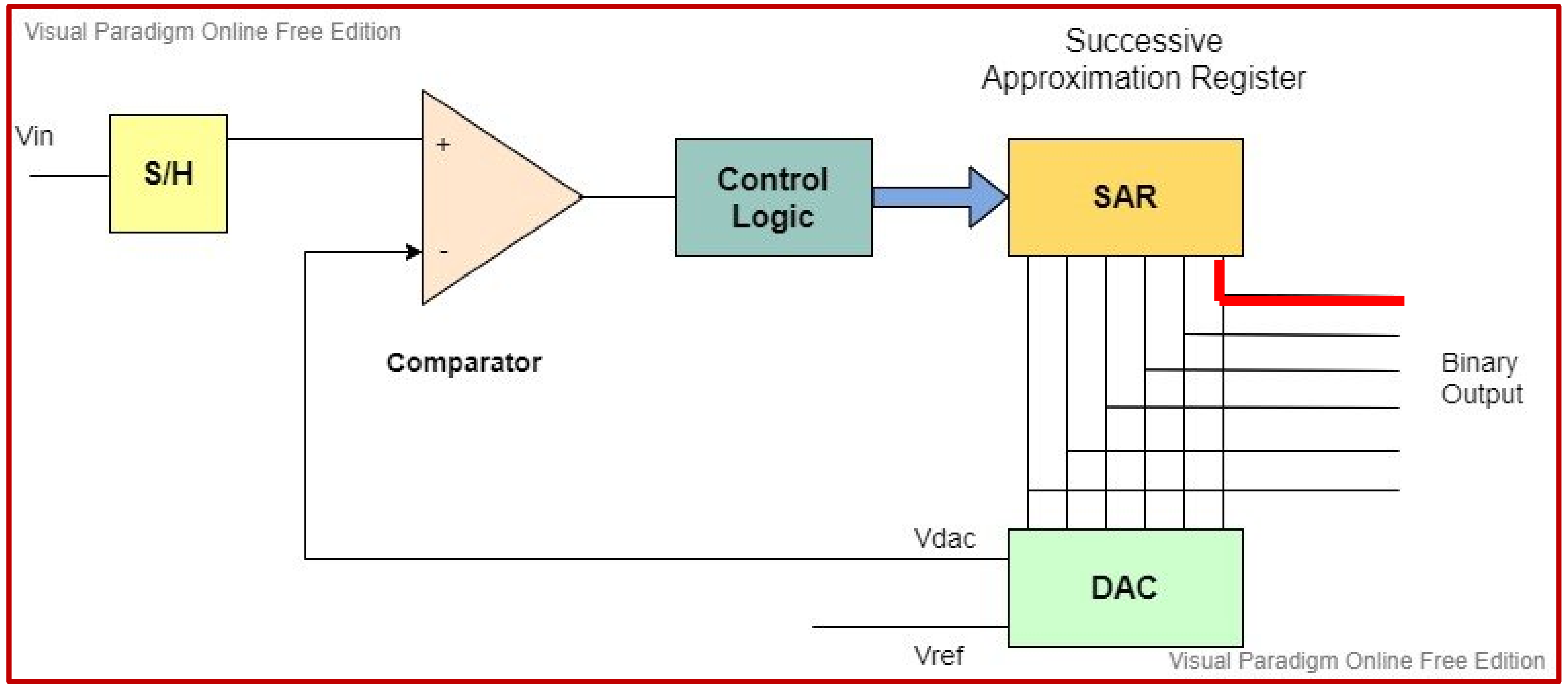
The Successive Approximation Register

- For an 8-bit ADC, $V_{max} = 10, V_{in} = 7.65$

- We Start with initial $V_{avg\_untog} = V_{max} = 10$ and $V_{avg_8} = 0$ *(note there is no 8th bit, it is just an initial value)*
- For the 7th bit:

$$V_{avg_7} = \frac{10+0}{2} = 5$$

$$(V_{in} = 7.65) > (V_{avg_7} = 5) \therefore B_7 = 1$$



Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1							

Analog-to-Digital Converter (ADC): *Behind The Scene*

The Successive Approximation Register

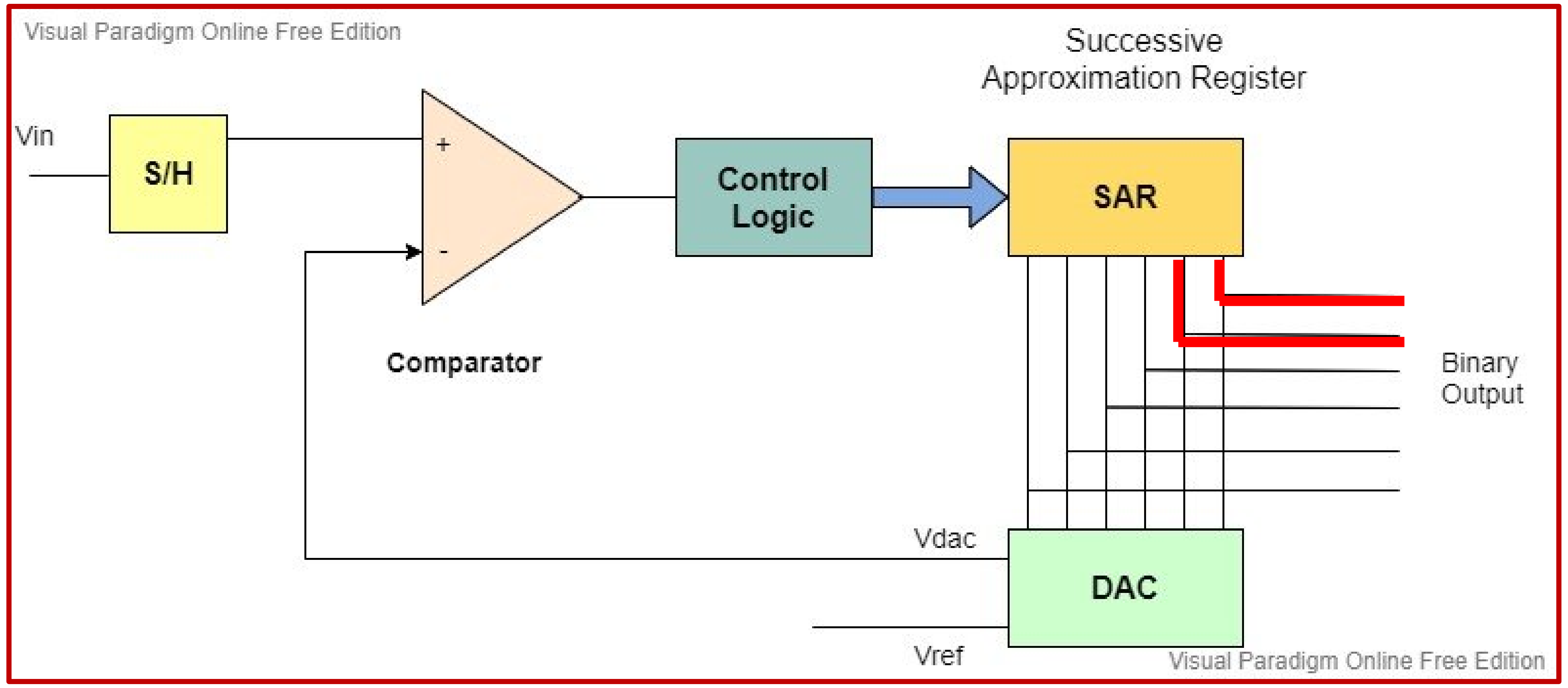
- For an 8-bit ADC, $V_{max} = 10, V_{in} = 7.65$

- Now, $V_{avg_7} = 5, V_{avg\_untog} = 10$

- For the 6th bit:

$$V_{avg_6} = \frac{10+5}{2} = 7.5$$

$$(V_{in} = 7.65) > \left(V_{avg_6} = 7.5\right) \therefore B_6 = 1$$



Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	1						

GUC German University in Cairo الجامعة الألمانية بالقاهرة

Analog-to-Digital Converter (ADC): *Behind The Scene*

The Successive Approximation Register

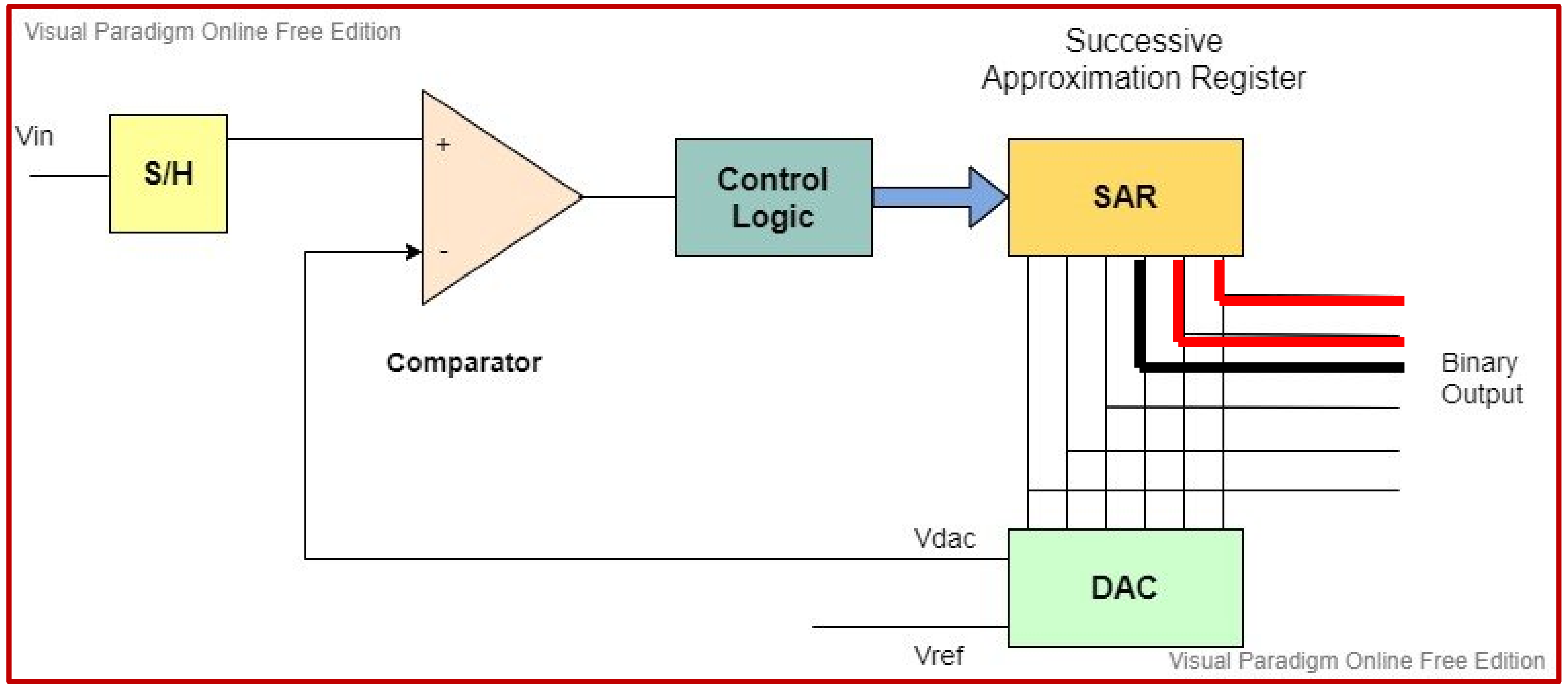
- For an 8-bit ADC, $V_{max} = 10, V_{in} = 7.65$
- Now, $V_{avg_6} = 7.5, V_{avg\_untog} = 10$
- For the 5th bit:

$$V_{avg_5} = \frac{10 + 7.5}{2} = 8.75$$

$$(V_{in} = 7.65) < \left(V_{avg_5} = 8.75\right) \therefore B_5 = 0$$

- Since a toggle occurred then

$$V_{avg\_untog} = V_{avg_6} = 7.5$$



Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	1	0					

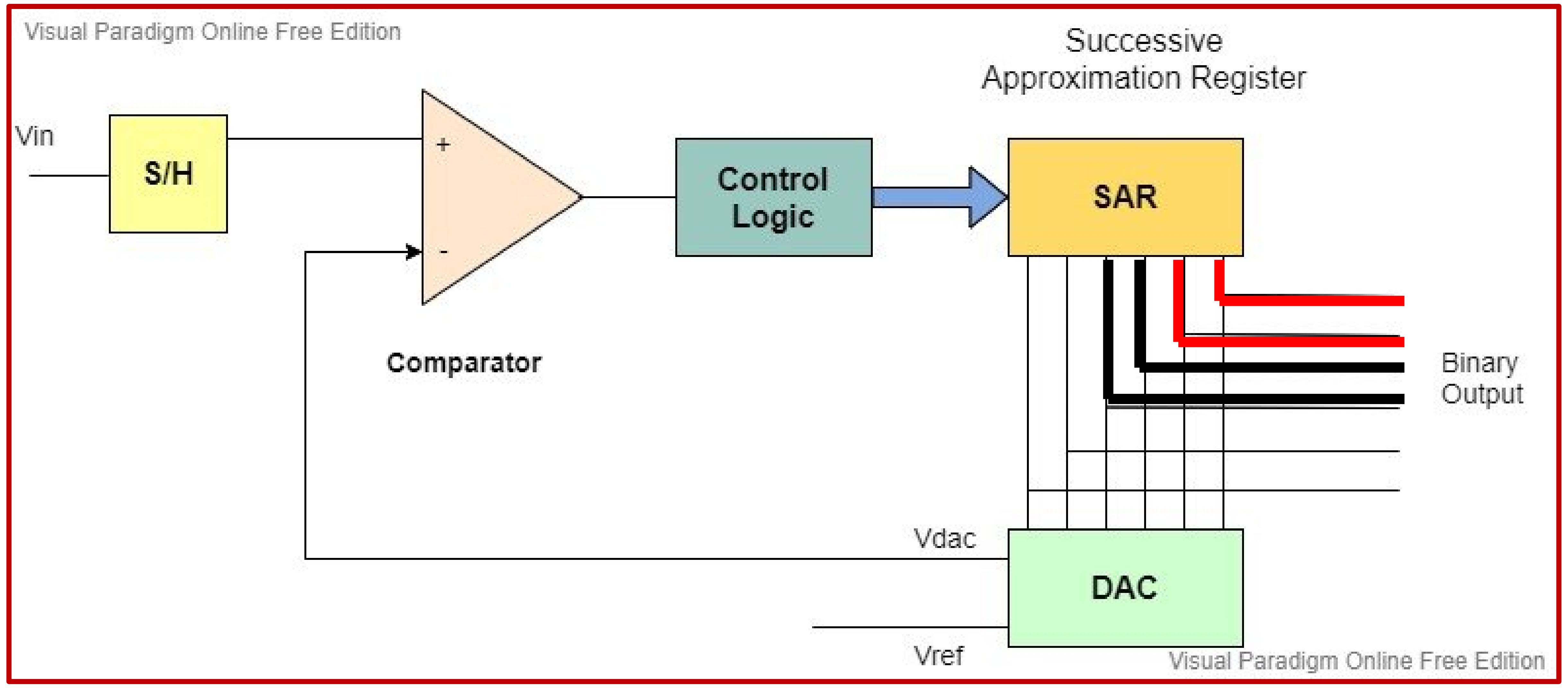
Analog-to-Digital Converter (ADC): *Behind The Scene*

The Successive Approximation Register

- For an 8-bit ADC, $V_{max} = 10, V_{in} = 7.65$
- Now, $V_{avg_5} = 8.75, V_{avg\_untog} = 7.5$
- For the 4th bit:

$$V_{avg_4} = \frac{7.5 + 8.75}{2} = 8.125$$

$$(V_{in} = 7.65) < \left(V_{avg_4} = 8.125\right) \therefore B_4 = 0$$



Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	1	0	0				

Analog-to-Digital Converter (ADC): *Behind The Scene*

The Successive Approximation Register

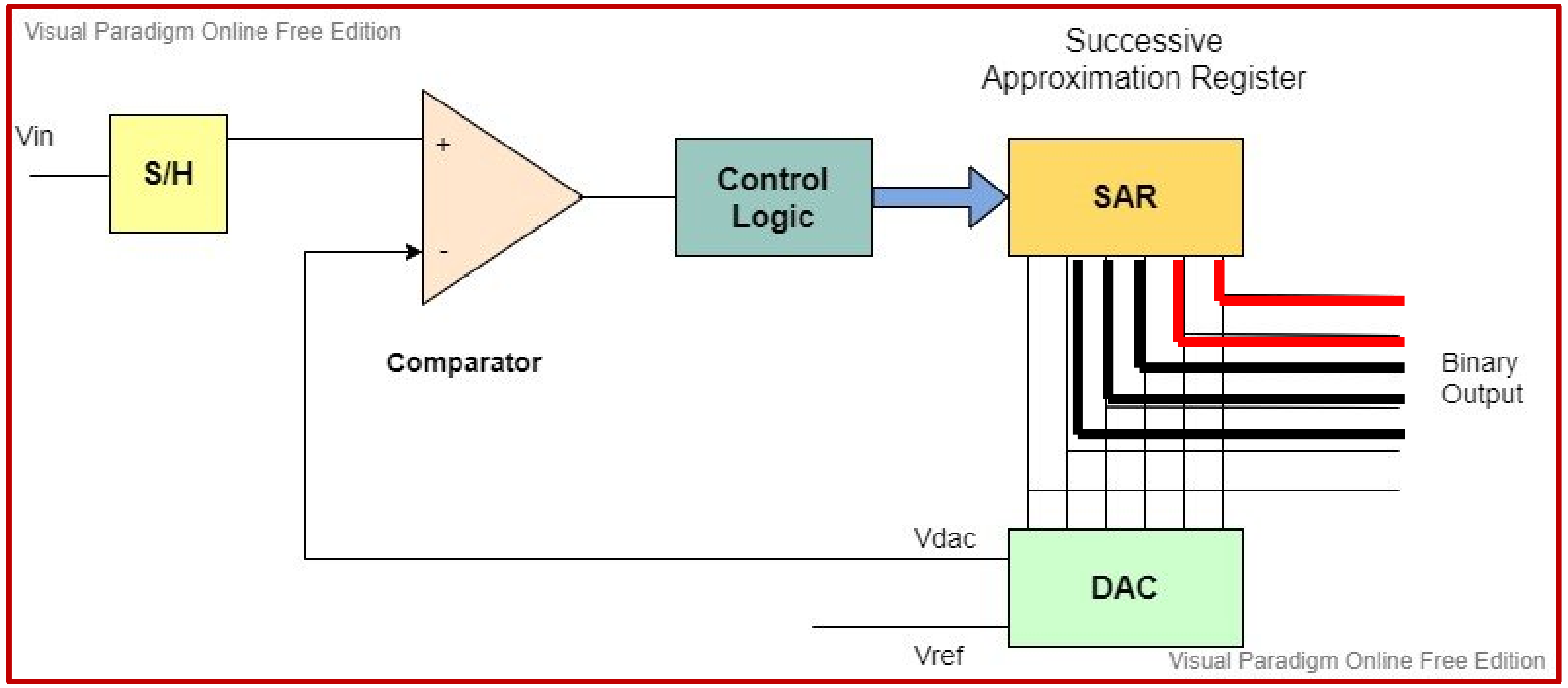
- For an 8-bit ADC, $V_{max} = 10, V_{in} = 7.65$

- Now, $V_{avg_4} = 8.125, V_{avg\_untog} = 7.5$

- For the 3th bit:

$$V_{avg_3} = \frac{7.5 + 8.125}{2} = 7.8125$$

$$(V_{in} = 7.65) < \left(V_{avg_3} = 7.8125\right) \therefore B_3 = 0$$



Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	1	0	0	0			

Analog-to-Digital Converter (ADC): *Behind The Scene*

The Successive Approximation Register

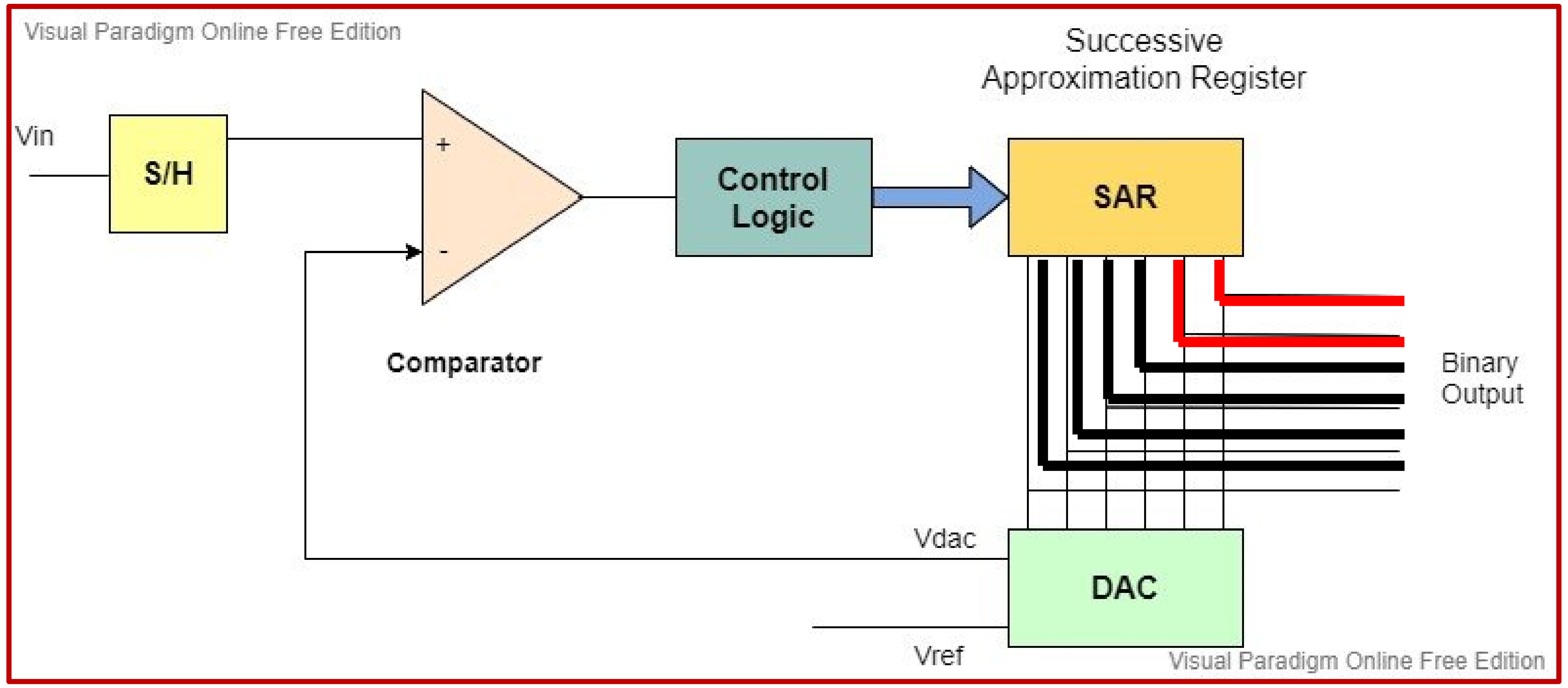
- For an 8-bit ADC, $V_{max} = 10, V_{in} = 7.65$

- Now, $V_{avg_3} = 7.8125, V_{avg\_untog} = 7.5$

- For the 2th bit:

$$V_{avg_2} = \frac{7.5 + 7.8125}{2} = 7.65625$$

$$(V_{in} = 7.65) < \left(V_{avg_2} = 7.65625\right) \therefore B_2 = 0$$



Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	1	0	0	0	0		

GUC German University in Cairo الجامعة الألمانية بالقاهرة

Analog-to-Digital Converter (ADC): *Behind The Scene*

The Successive Approximation Register

- For an 8-bit ADC, $V_{max} = 10, V_{in} = 7.65$

- Now, $V_{avg_2} = 7.65625, V_{avg\_untog} = 7.5$

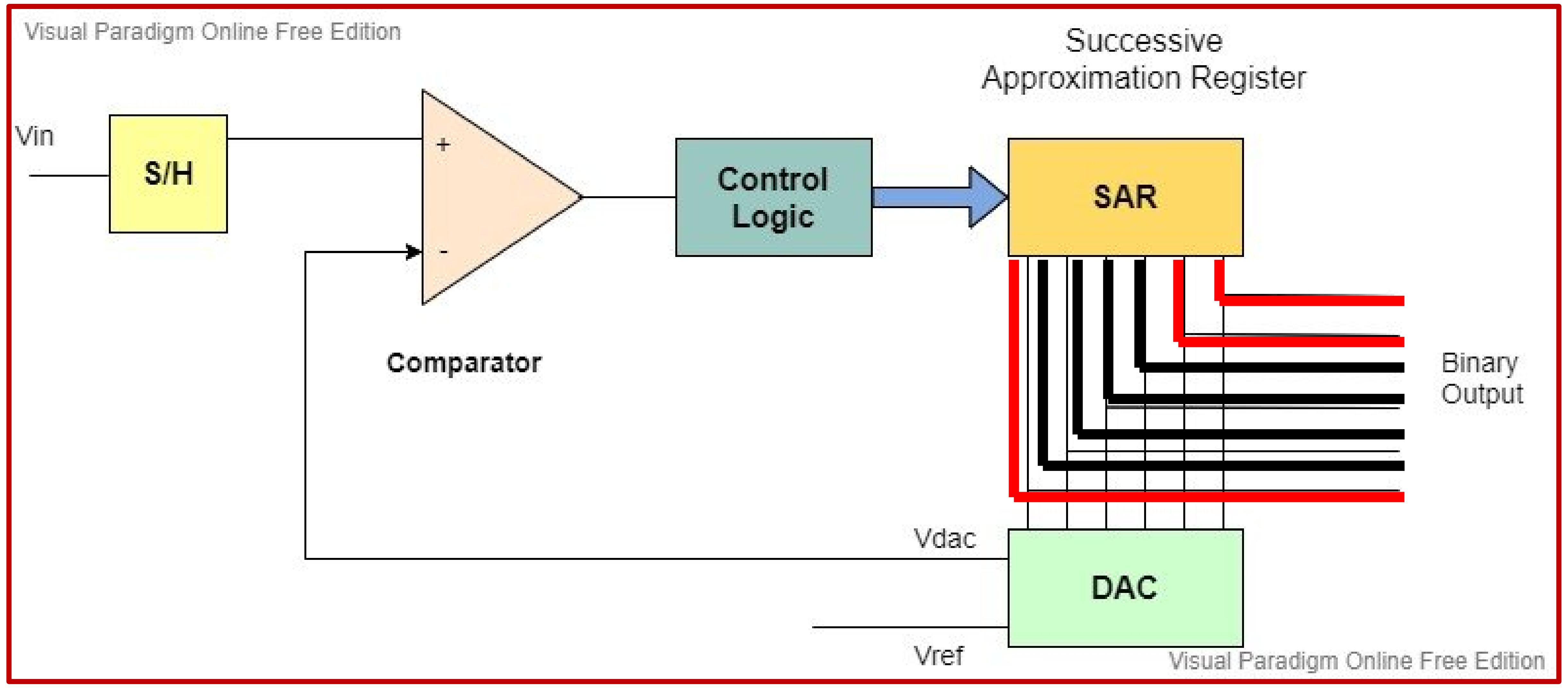
- For the 1th bit:

$$V_{avg_1} = \frac{7.5 + 7.65625}{2} = 7.578125$$

$$(V_{in} = 7.65) > (V_{avg_1} = 7.578125\ ) \therefore B_1 = 1$$

- Since a toggle occurred then

$$V_{avg\_untog} = V_{avg_2} = 7.65625$$



Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	1	0	0	0	0	1	

Analog-to-Digital Converter (ADC): *Behind The Scene*

The Successive Approximation Register

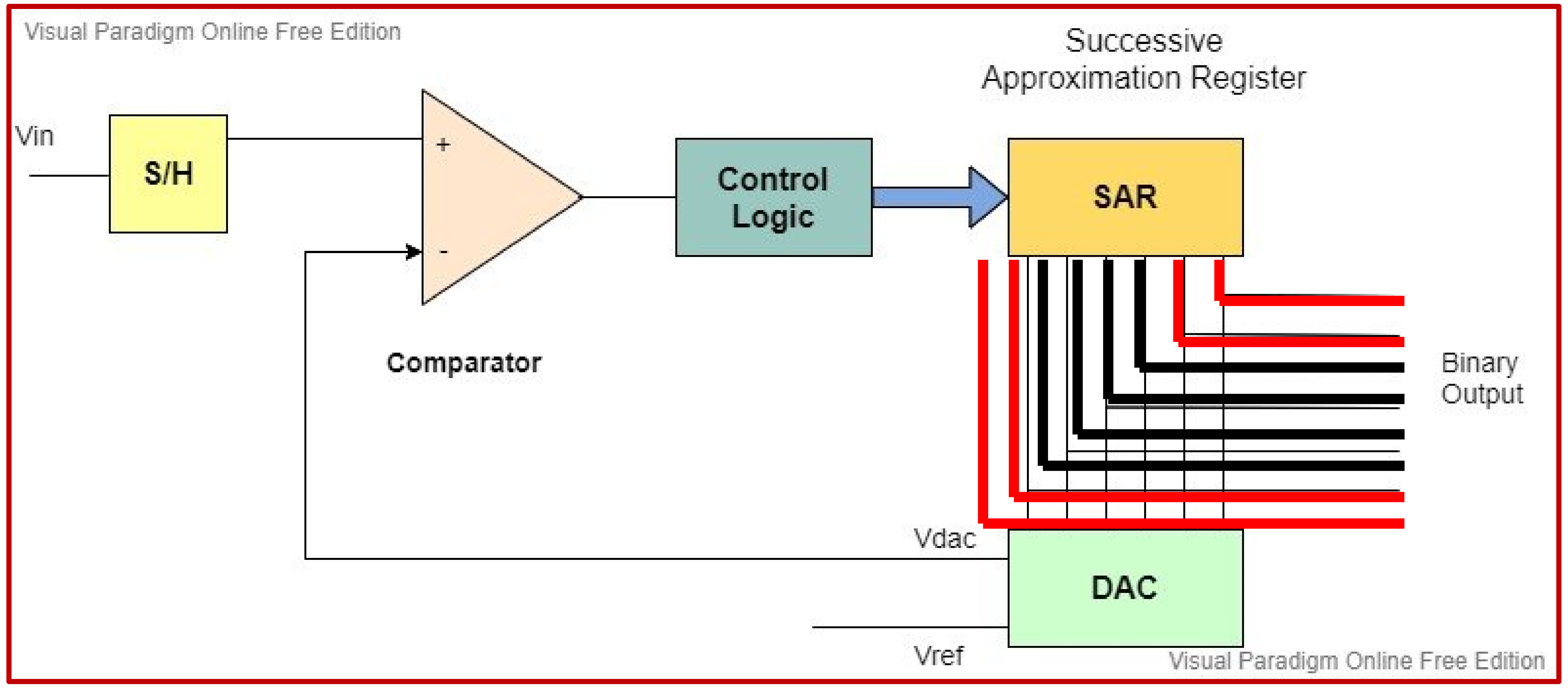
- For an 8-bit ADC, $V_{max} = 10, V_{in} = 7.65$

- Now, $V_{avg_2} = 7.578125, V_{avg\_untog} = 7.65625$

- For the 0th bit:

$$V_{avg_0} = \frac{7.65625 + 7.578125}{2} = 7.6171875$$

$$(V_{in} = 7.65) > \left(V_{avg_0} = 7.6171875\right) \therefore B_0 = 1$$



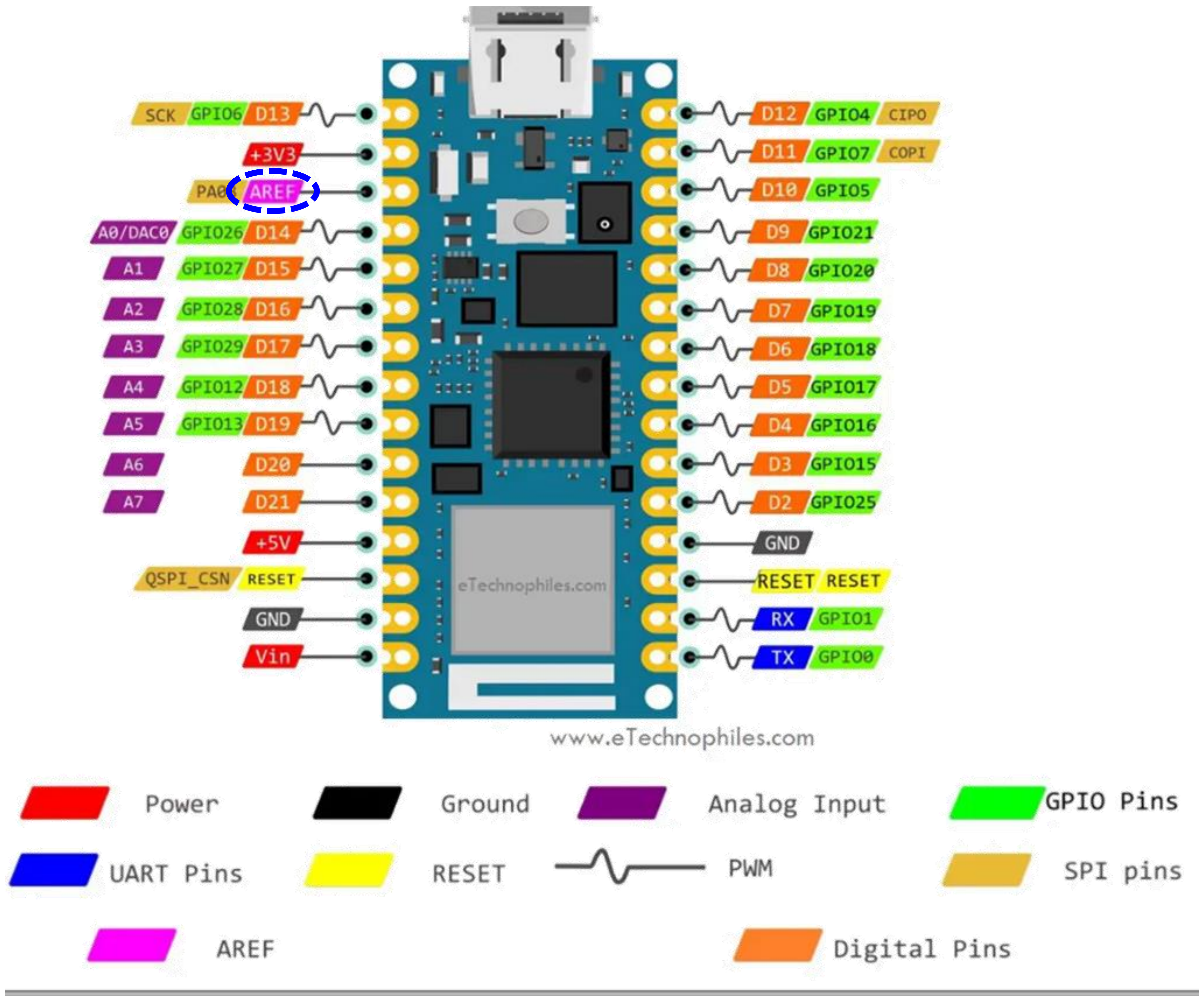
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
1	1	0	0	0	0	1	1

Sensor Characteristics: *Resolution*

- ***Example:***
  - ➢Imagine that a sensor only outputs from 0-3 volts connected to 10-bit ADC embedded in a 5-volt microcontroller. This means what?
    - ADC $Res \ = \frac{5}{2^{10}-1} = 4.88mV$
    - Analog reading of 3 volts will be converted to digital reading $digital_{range} = \frac{3}{0.00488} \approx 614$!!!!!
    - ***Comment:*** This is the maximum reading by the sensor, yet it did not reach the highest resolution

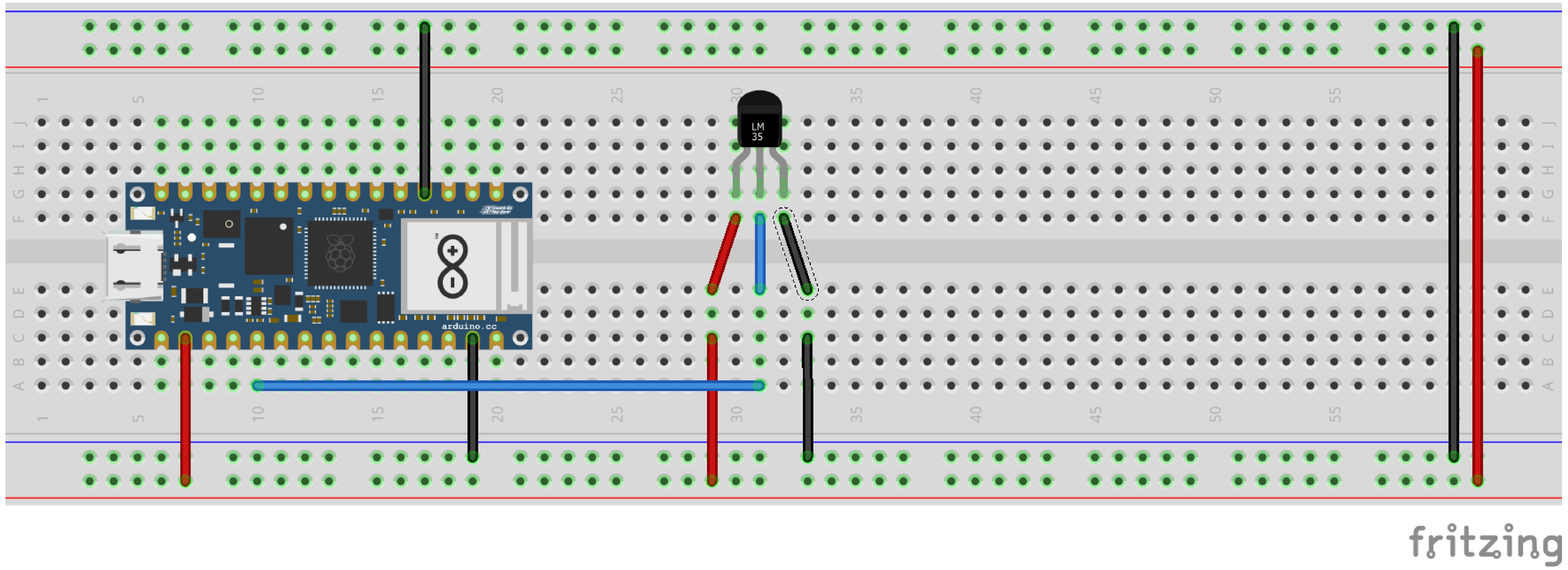
Sensor Characteristics: *Resolution*

- ***Example:***
- Can we improve the resolution?
  - ➢ ***Answer 1:*** Use a more precise sensor (more expensive).
  - ➢ ***Answer 2:*** use **Analog Reference (AREF)** Pin!
    - To increase the resolution of that measurement, the microcontroller can take a reference voltage, *usually supplied by a 'voltage divider'*
    - Feed it a 3v AREF,
    - New ADC $Res = \frac{3}{2^{10}-1} = 2.932mV$
    - That's ~40% more precise



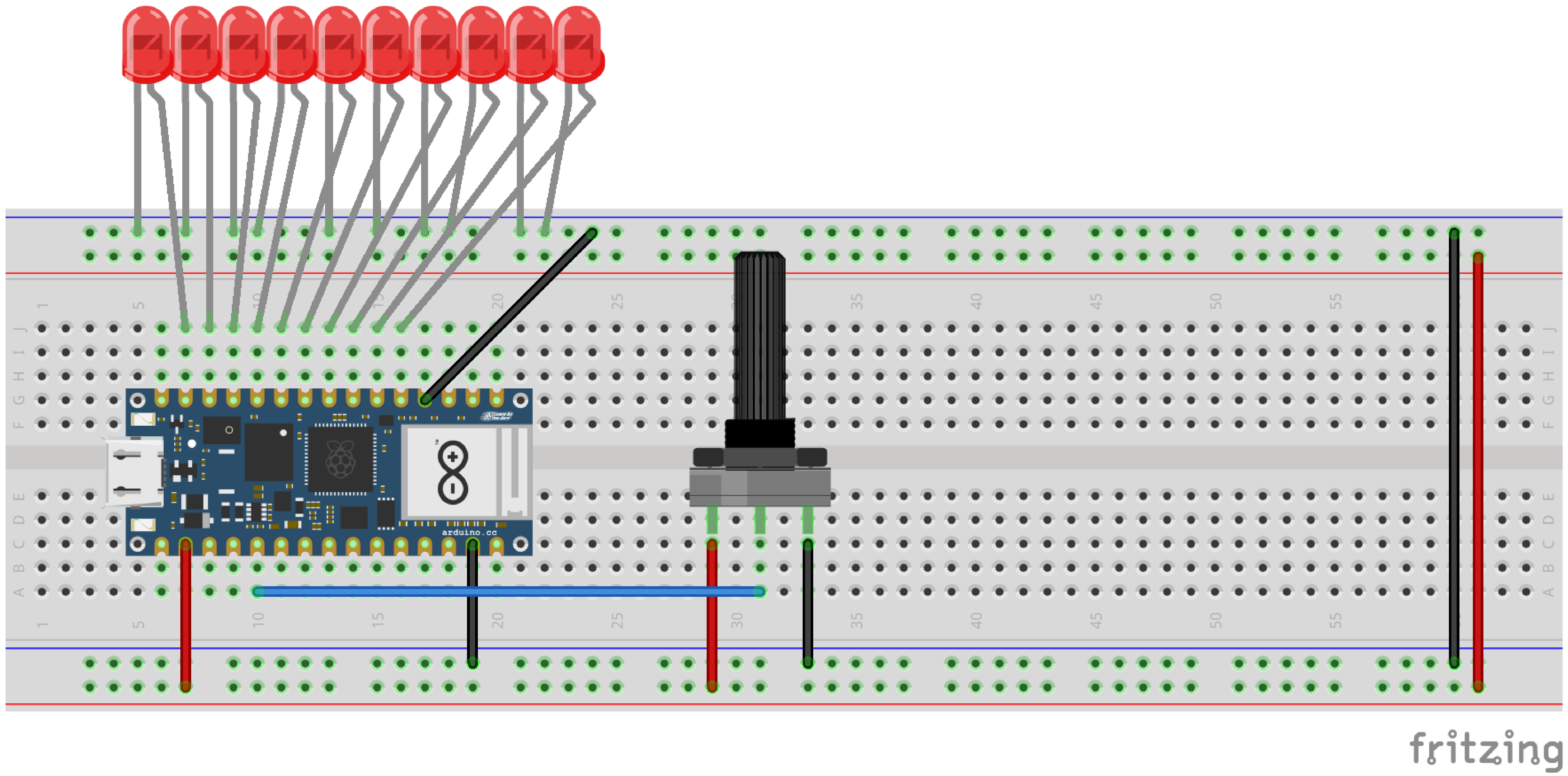
Let's See Some Action ☺

Experiment 1:



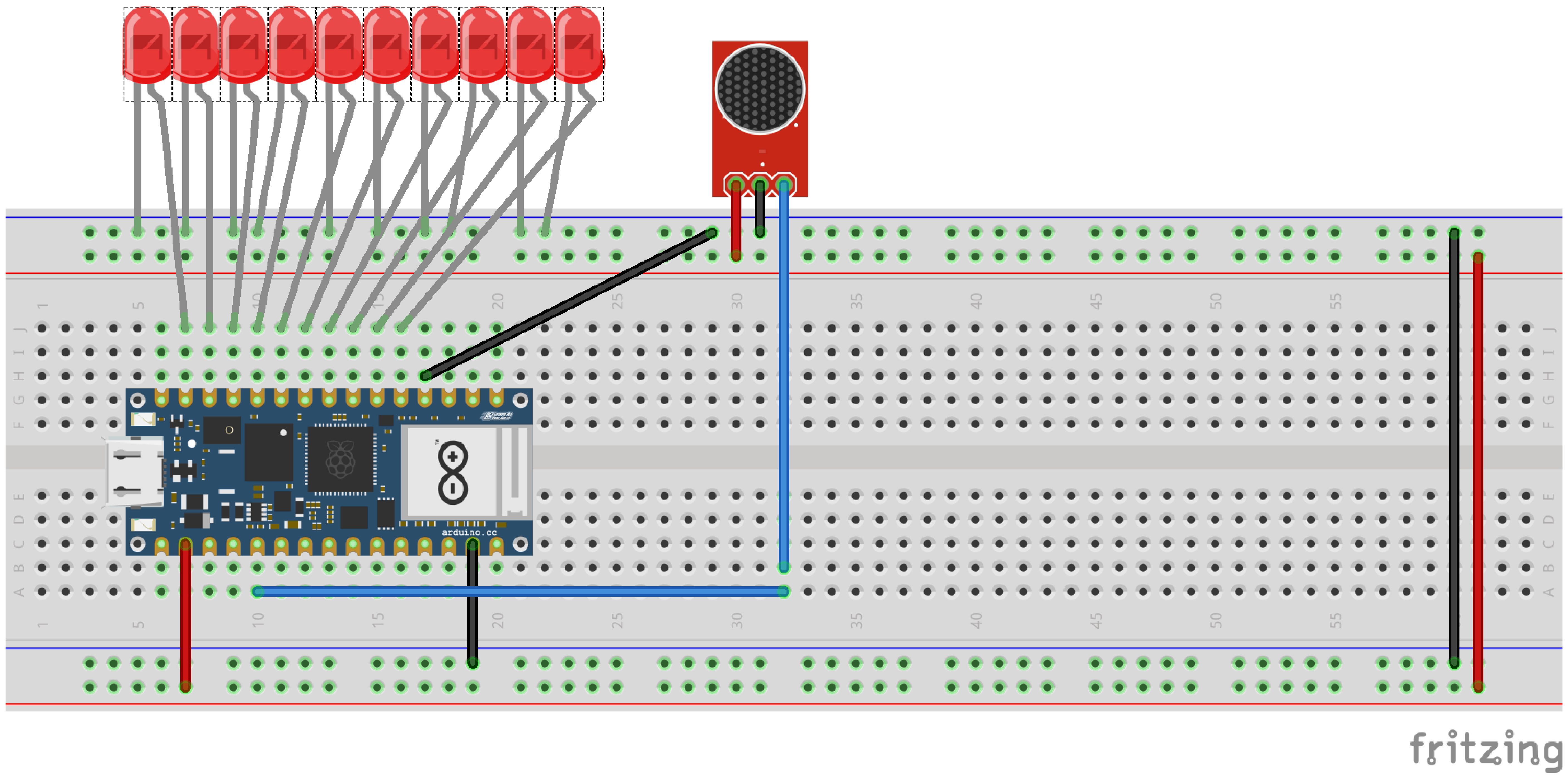
- The experiments uses a Temperature Sensor to measure the room temperature and show the time response curve (time-Reading) to spot the fluctuations due to uncertainties.
- The above figure shows the wiring diagram.
- The code is written in C-coding.

Experiment 2:



- The experiments uses a potentiometer sensor (analog) to test the ADC, showing the digital output on 10 LEDs
- The above figure shows the wiring diagram.
- The code is written in C-coding.

Experiment 3:



- The experiments uses a microphone sensor (analog) to test the ADC, showing the digital output on 10 LEDs
- The above figure shows the wiring diagram.
- The code is written in C-coding.

For Further Inquiries, Please send an email

Catherine.elias@guc.edu.eg, Catherine.elias@ieee.org

Thank you for your attention!

See you next time ☺