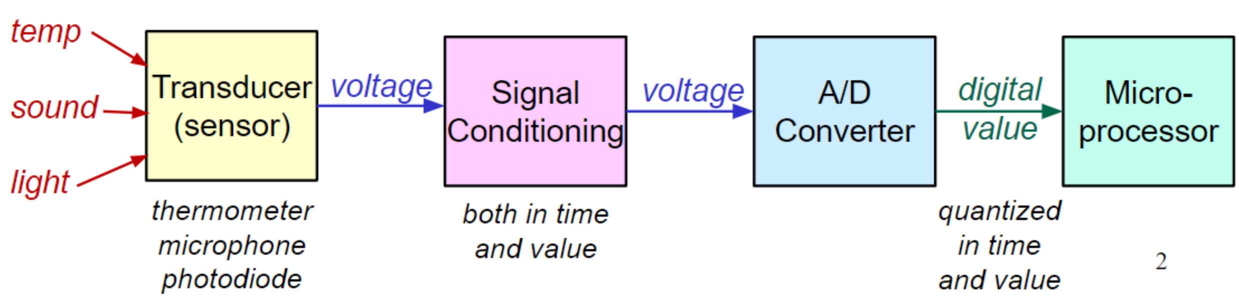
Analog to Digital Conversion

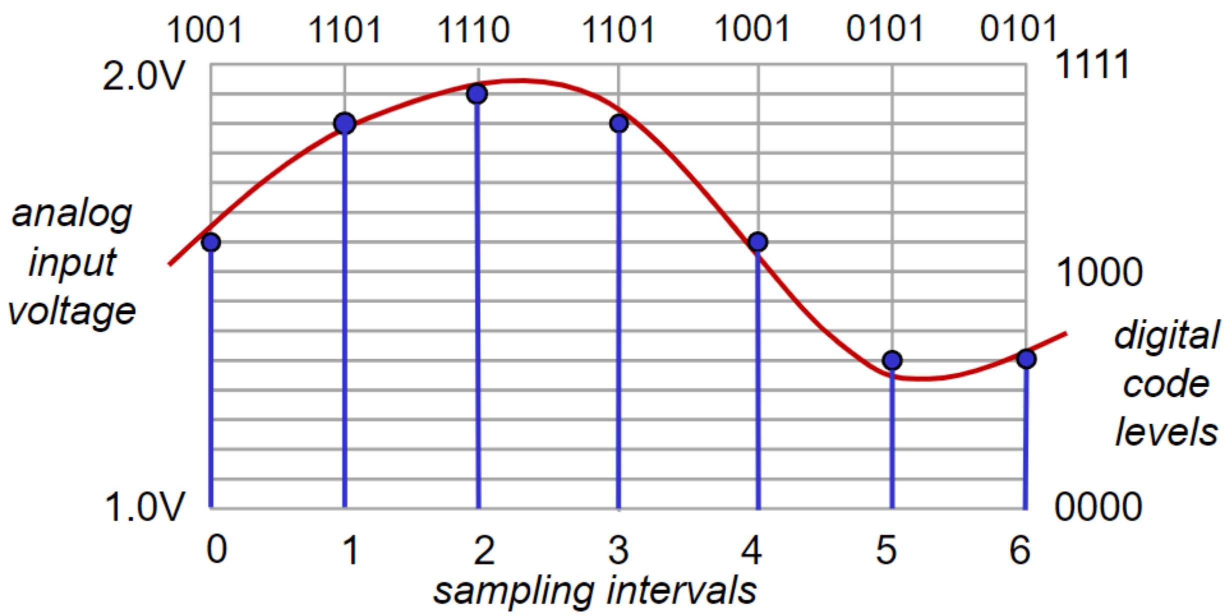
Real World of Analog

- Microprocessor deals exclusively with digital data
  - Finite precision representations of external real world and internal computational data
- Microcontroller in embedded application takes inputs from real-world sensors
  - Some of these are already digital (e.g. switches, keyboard, mouse)
  - Many are analog (e.g. pressure, temperature, light intensity, microphone, airflow, engine speed, oxygen level)
- Analog-to-Digital converter (A/D) transforms analog signal into digital representation used by microprocessor



Analog to Digital Conversion

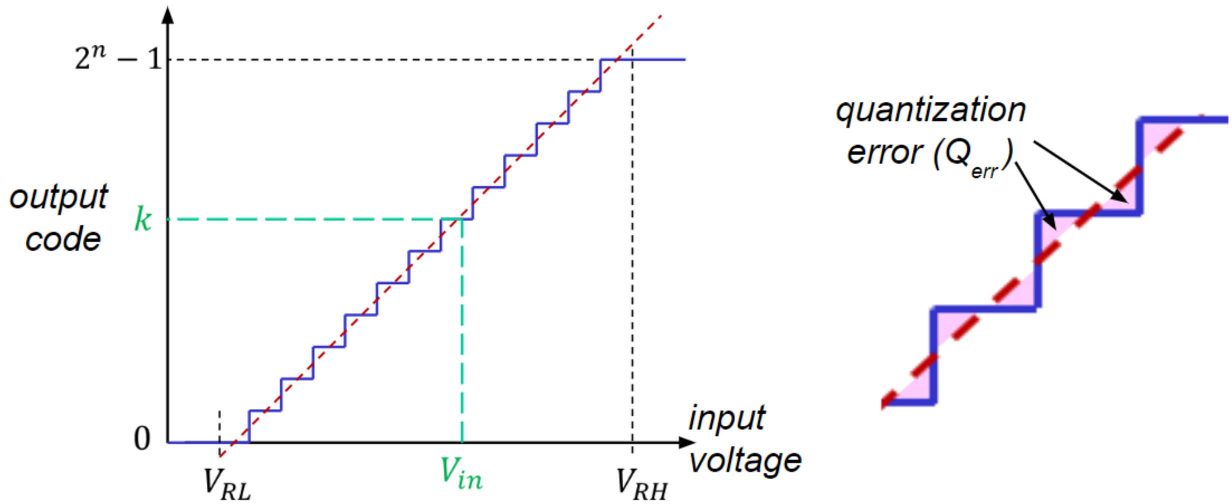
- A/D converter samples analog signal at regular intervals and generates digital code which is its best (closest) approximation to analog value at that instant



- Analog signal: continuous in time and value
- Digital signal: quantized in time and value

A/D Transfer Function

- n-bit A/D converter has 2^n possible output codes
- Input voltage range typically defined by two reference voltages V_{RL} and V_{RH}



$$V_{in} = V_{RL} + \frac{(V_{RH} - V_{RL}).k}{2^n - 1} \pm Q_{err}$$

4

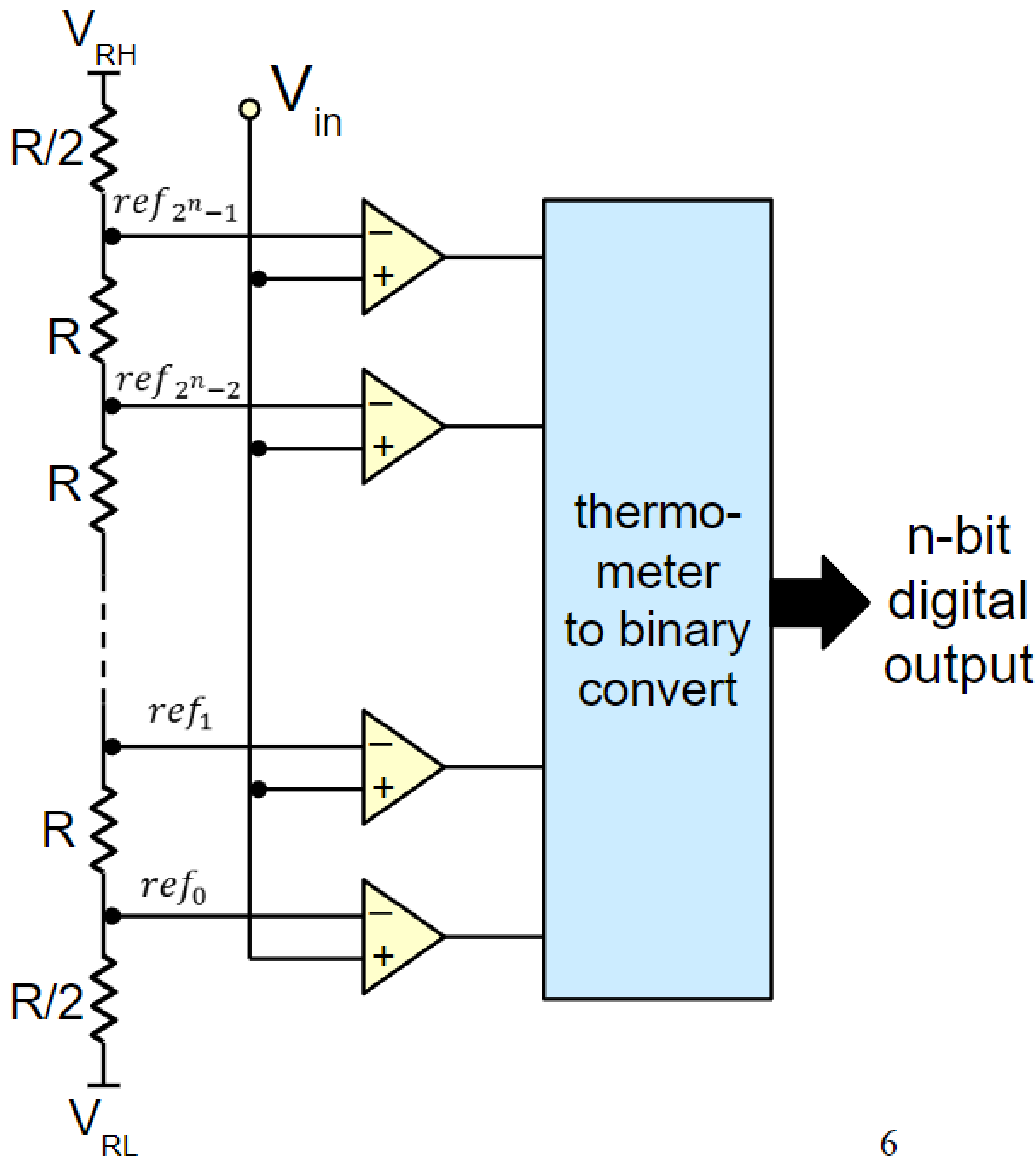
A/D Characteristics

Resolution

- Often quoted in terms of # bits (e.g. 12-bit converter)
- Analog resolution is $(V_{RH} - V_{RL})/2^n$

Flash (Parallel) A/D Converter

- Resistor ladder generates 2^n reference voltages
- 2^n comparators simultaneously compare input with each reference
- Comparator output k is high if Vin > ref$_k$
- Conversion logic generates code indicating greatest value of k for which comparator output is high
- Very high speed
- Expensive in area & power
- Limited to ~8-bits

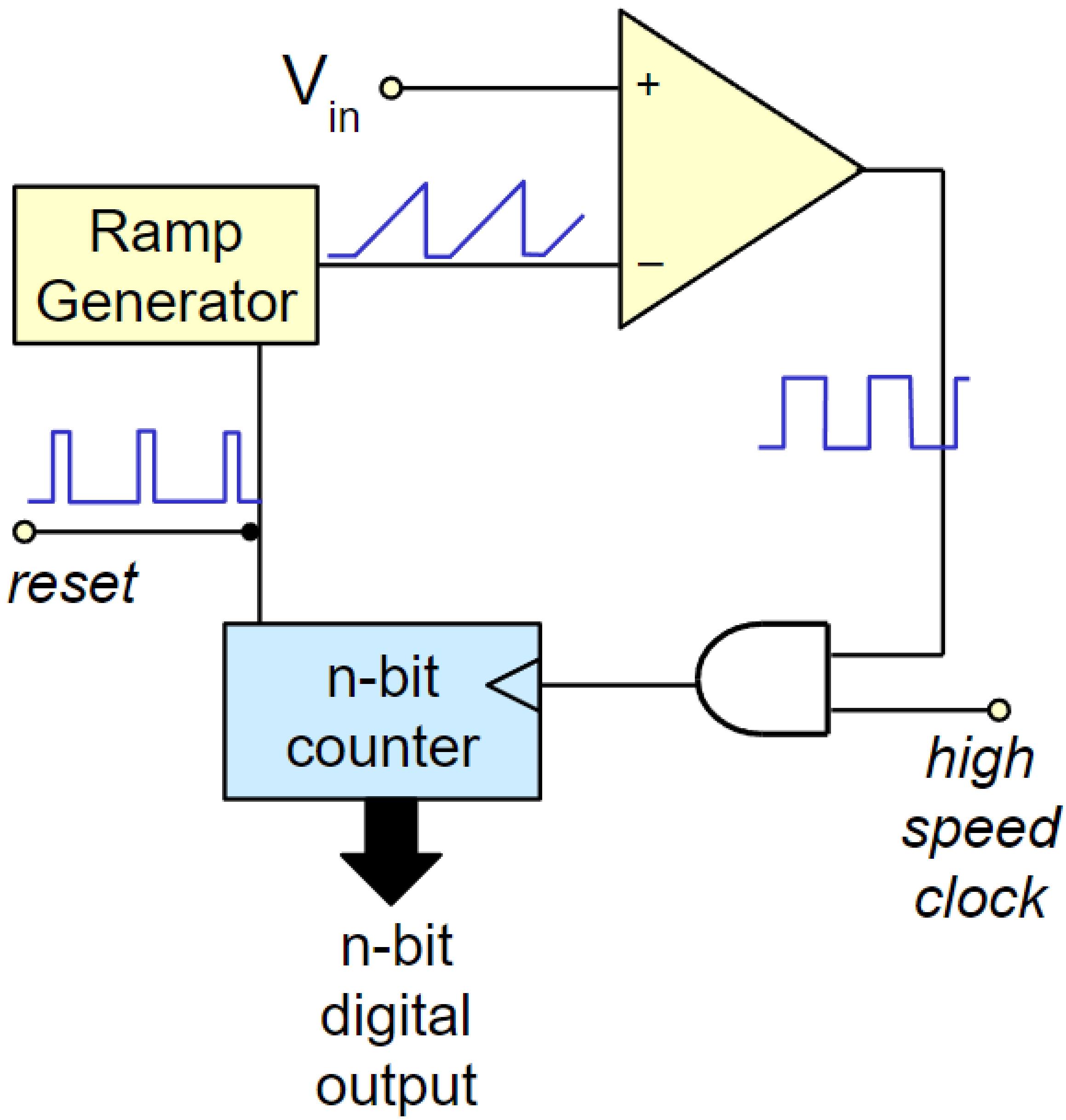


Single Slope A/D Converter

- Compares input to linear ramp to generate pulse width proportional to V$_{in}$
- Pulse used to gate clock to high speed digital counter
- Simple hardware - popular in low speed applications

- High resolution possible
- Performance limited by: $f_{ck} = f_{samp} \times 2^n$

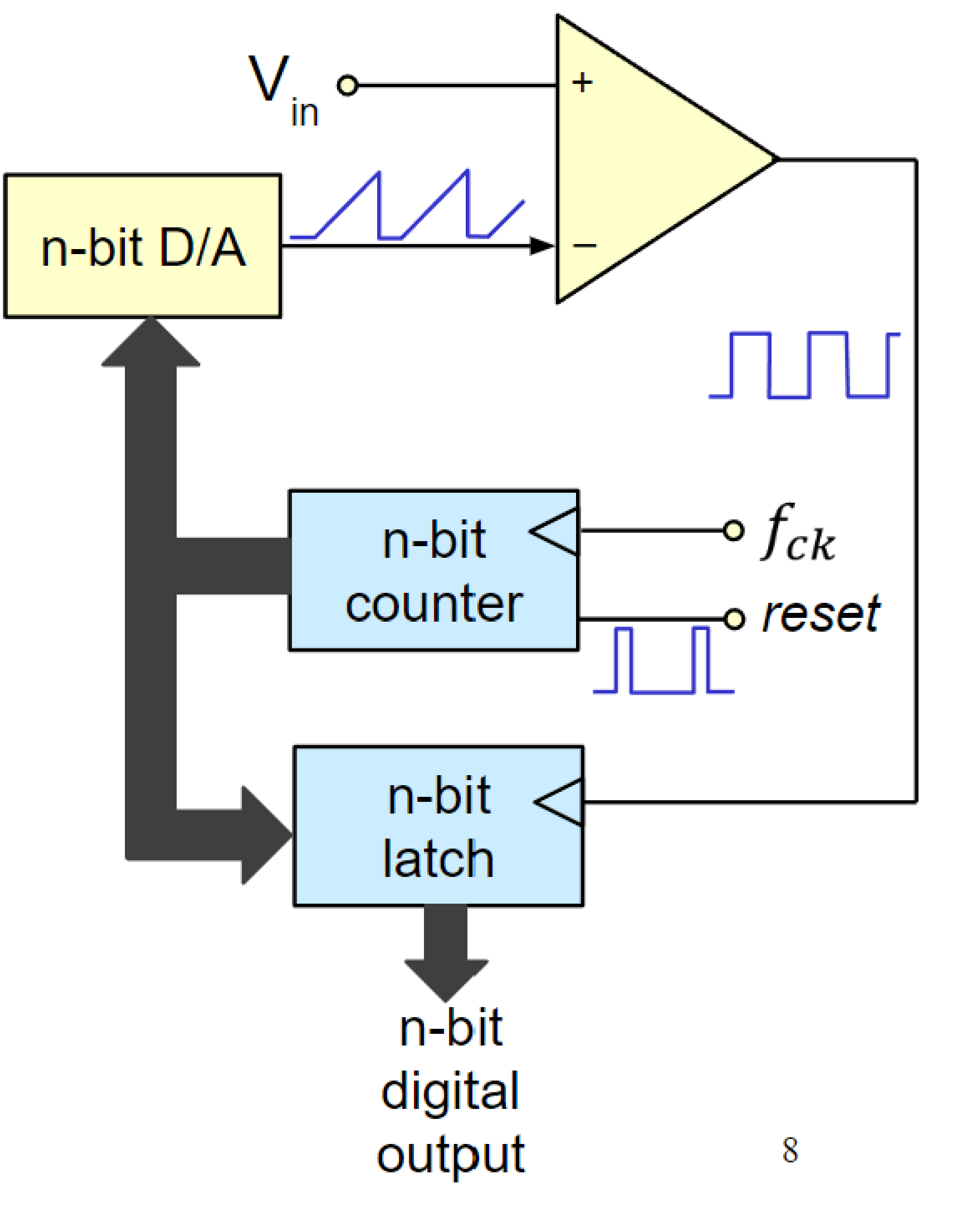
e.g. for $f_{samp} = 1$ MHz, 12-bit converter requires $f_{ck} = 4$ GHz



Counter Ramp A/D Converter

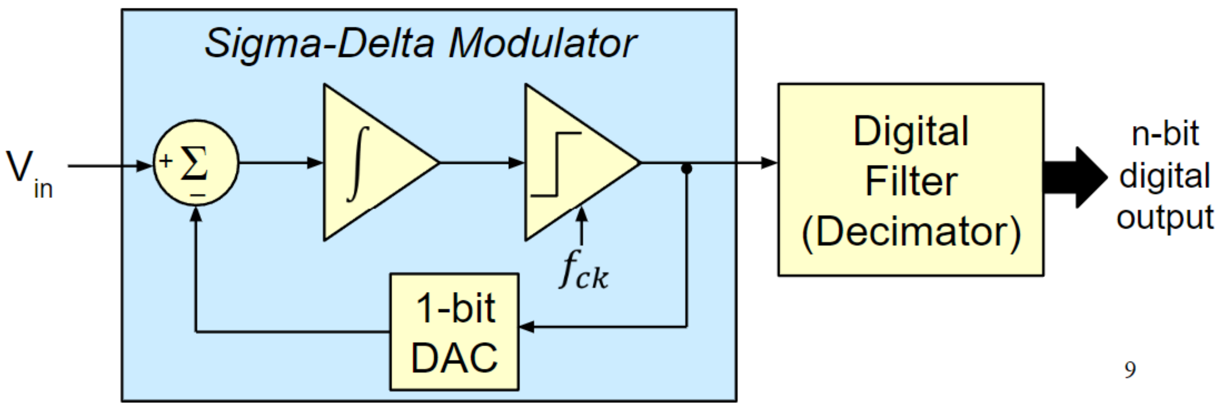
- Variant on single-slope converter
- Ramp is generated by counter driving D/A converter

- When D/A output ramp crosses V_{in}, counter value is captured in n-bit latch
- Does not require precision analog ramp generation
- Precision limited by linearity of D/A



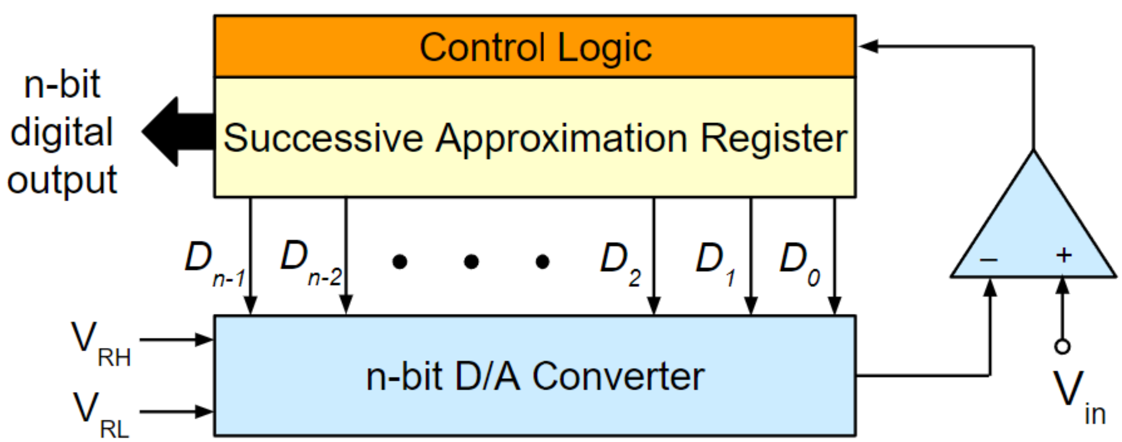
Sigma Delta (Oversampling) A/D Converter

- Sigma-delta modulator consists of summer, integrator, clocked comparator, and 1-bit DAC
- Modulator runs at many times (e.g. 16x - 1000x) required sampling frequency to produce very high speed 1-bit waveform
- Digital filter converts this to much slower n-bit digital output
- Since 1-bit DAC is perfectly linear, can produce very high resolution (up to 24-bit)
- Sampling frequency is limited by need to over-sample



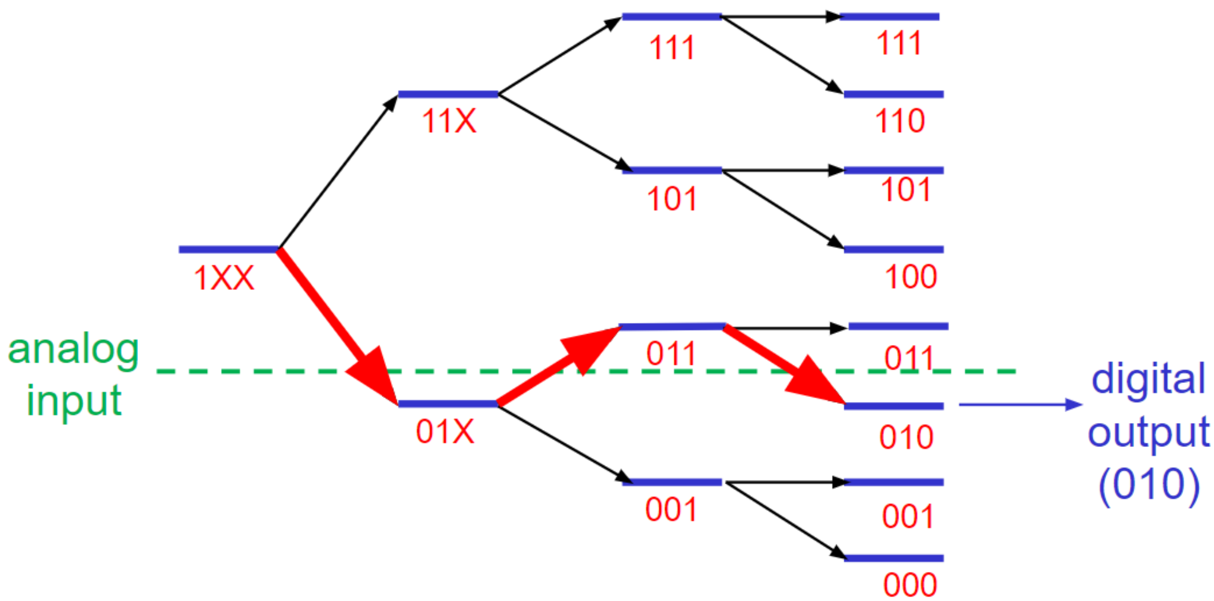
Successive Approximation A/D Converter

- Guesses and then corrects digital code in SAR one bit at time



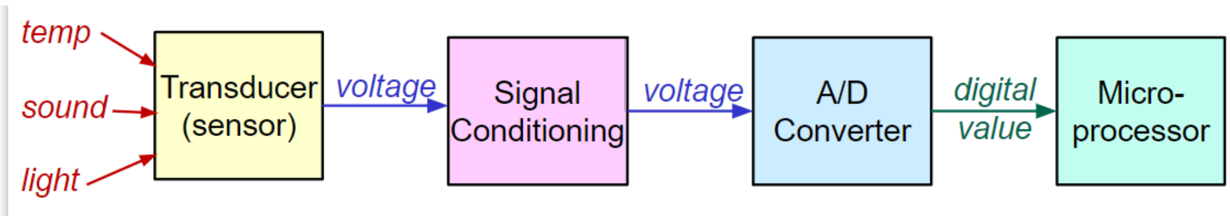
- Initially set all bits in SAR to ‘0’
- Then starting with MSB, for each bit:
    - Set bit to ‘1’ and convert output of SAR to analog value with D/A
    - Compare output of D/A to input voltage
    - If D/A is larger, set this bit back to ‘0’ and go on to next (lesser sig.) bit
    - If input is larger, retain ‘1’ for this bit and go to next bit

Successive Approximation Process



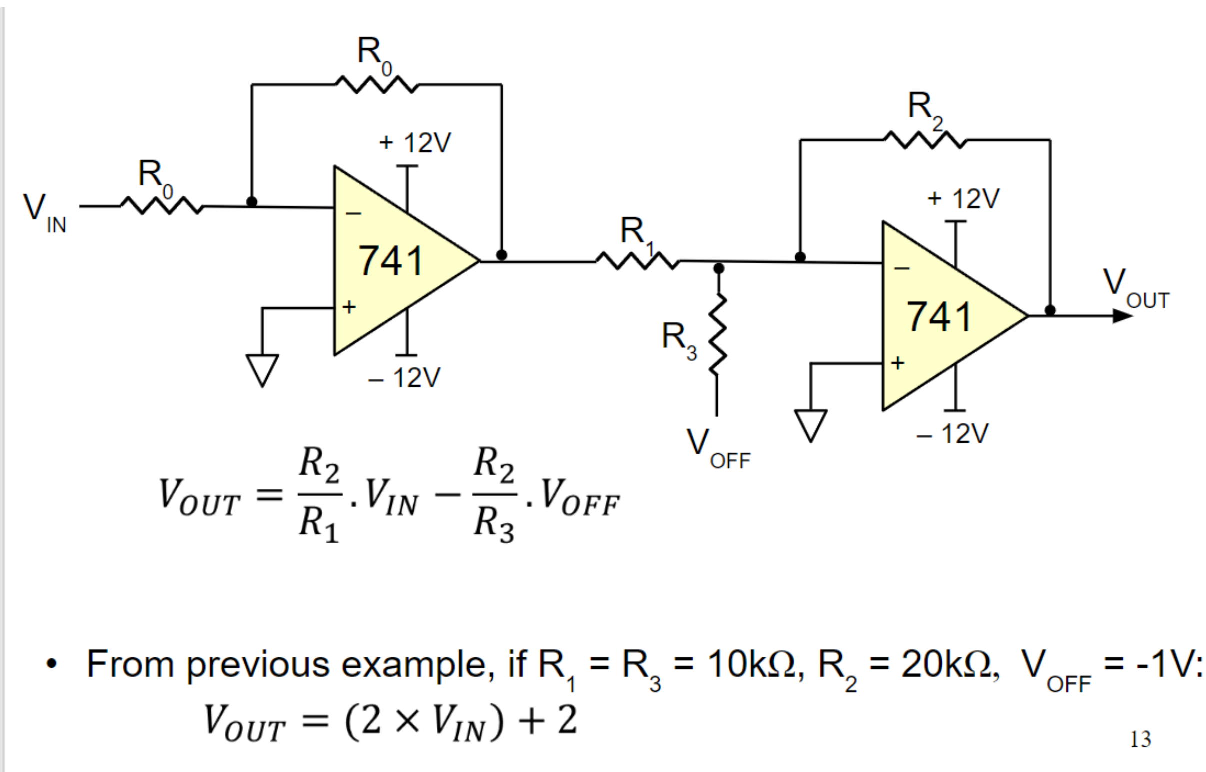
- SAR gives good tradeoff between speed and precision
- One of most popular A/D techniques in embedded systems
- Used in HCS12

Signal Conditioning



- Signal Conditioning is process of matching transducer output to input characteristics of A/D
  - Need to match in voltage and time (frequency)

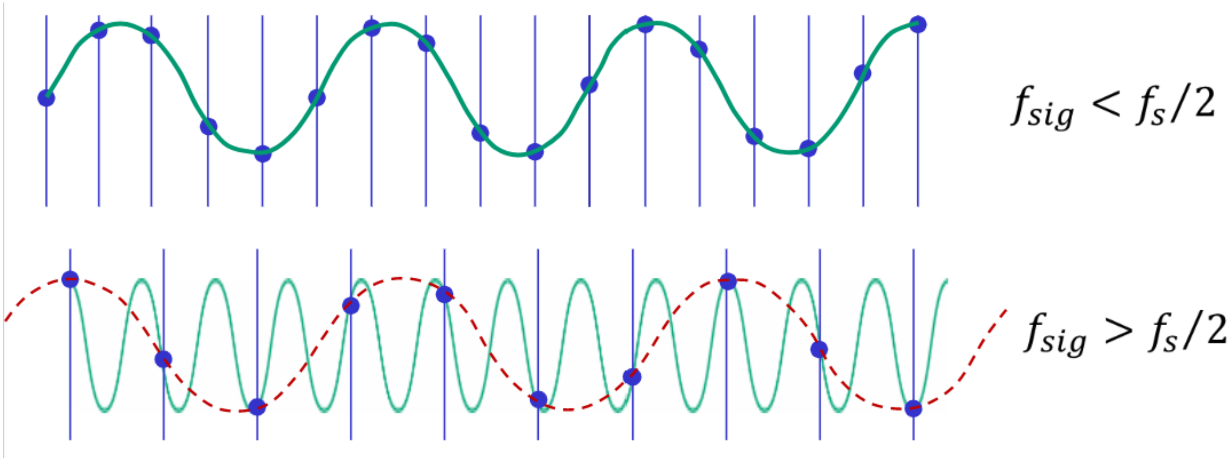
Shift & Scale Circuit



Nyquist Frequency

- If f_s is sampling frequency, $f_s/2$ is known as Nyquist frequency

Aliasing



- Even if desired signal does not contain components > Nyquist, there may be high frequency noise components which must be removed
- Signal conditioning circuits frequently include sharp low-pass filter to take out any signal components > Nyquist

A/D Conversion on Arduino Due

- Basic Arduino has 6 A/D channels, 10-bit about 7700hz
- Due has
    - 12 bit accuracy (1 part in 4096)
    - 1 MHz sample rate
    - 16 input channels
    - Basic Arduino can be run faster than rated A/D conversion less accurately

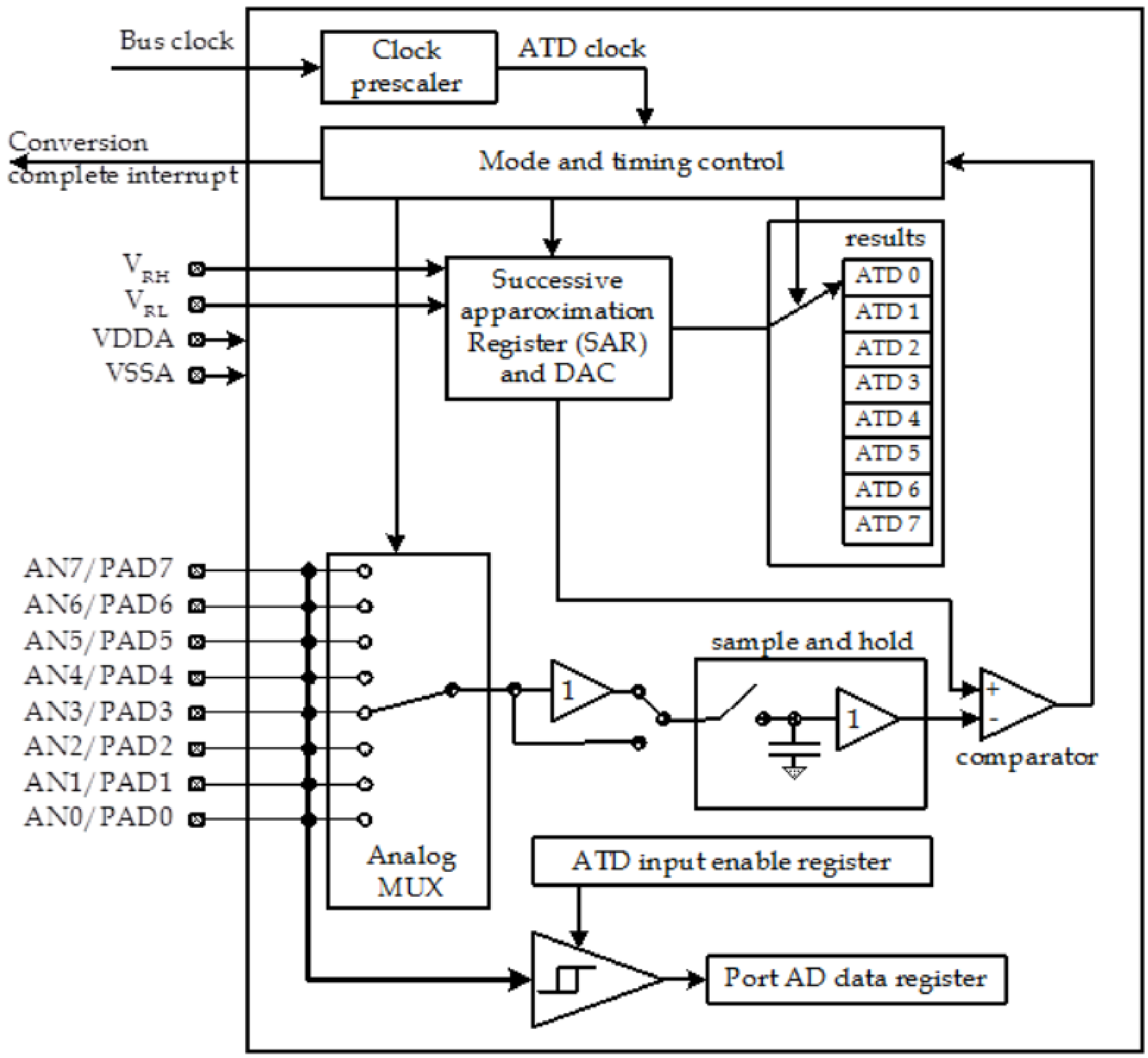
END OF KRUGER’S SLIDES

A/D Conversion on HCS12

- HCS12 may have one or two 8-channel 10-bit A/D’s
- Each uses successive approximation method
- A/D’s runs off ATD clock that can be set 500kHz ~ 2 MHz

    - At 2 MHz, ADC can perform 8-bit conversion in 6 μs or 10-bit conversion in 7 μs
- A/D conversion may be internally triggered (by writing to control register) or externally triggered (via pins AN7 or AN15)
- May be single conversion or sequence of conversions
- Result(s) can be 8-bit or 10-bit, signed or unsigned:
    - (-128 to +127) or (-512 to +511) signed
    - (0 to 255) or (0 to 1023) unsigned
- Result(s) stored in 16-bit register(s)
    - Either left or right justified

ATD Block Diagram



A/D Pins & Registers

- **Signal Pins:**
    - AD0 module uses pins AN0 ~ AN7
    - AD1 module uses pins AN8 ~ AN15
    - AN7 (AN15) pin can optionally be used to trigger AD0 (AD1) module
    - V_{RH} and V_{RL} are high and low reference voltage inputs
    - V_{DDA} and V_{SSA} are power supply and ground pins
- **Each A/D has following registers:**

- Six control registers ATDxCTL0 ~ ATDXCTL5
  - (0 and 1 for factory testing only)
- Two status registers ATDxSTAT0 ~ ATDxSTAT1
- One input enable register ADTxDIEN
- One port data register PTADx
- Eight 16-bit result registers ATDxDR0 ~ ATDxDR7

where x=0 or 1 (will only describe AD0 registers in following slides)

ATD Control Register 2 (ATD0CTL2)