Analog-to-digital conversion Digital-to-analog conversion

Part 2

ECE 2534

Outline

- Motivation
- Basic idea
- ADC on the PIC32
- Digital-to-analog conversion methods
- Analog-to-digital conversion methods

REMINDER:

Need to interface with the real (analog) world

Sensor → ADC → µController → DAC → Actuator

Analog to digital:

$$D = \left\lfloor \left(\frac{V_{In} - V_{RefLow}}{V_{RefHigh} - V_{RefLow}} \right) \times 2^n \right\rfloor$$

Digital to analog:

$$V_{Out} = \left(\frac{D}{2^n} \right) \left(V_{RefHigh} - V_{RefLow} \right) + V_{RefLow}$$

Outline

- Motivation
- Basic idea
- ADC on the PIC32
- Digital-to-analog conversion methods
- Analog-to-digital conversion methods

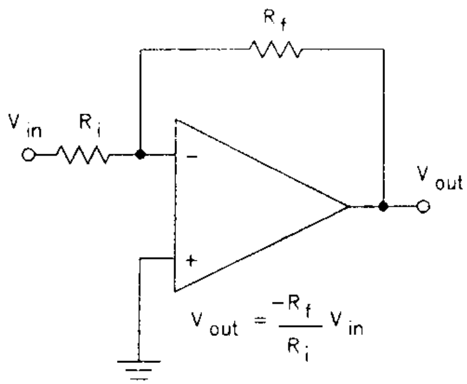
Digital-to-Analog Conversion

- Goal
  - **Convert a digital value to an analog voltage**
  - Usually an unsigned binary number, 8 to 16 bits
  - Common output range: a few volts, such as 0 to Vcc or –12 V to +12 V
- Normally use a linear mapping
- Common concerns
  - Resolution
  - Accuracy
  - Conversion speed

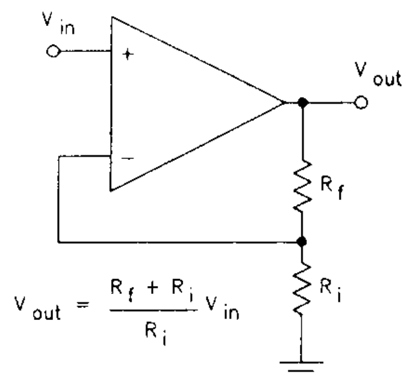
Three methods for D/A conversion

- Binary weighted ladder
- R-2R ladder
- Pulse-width modulation

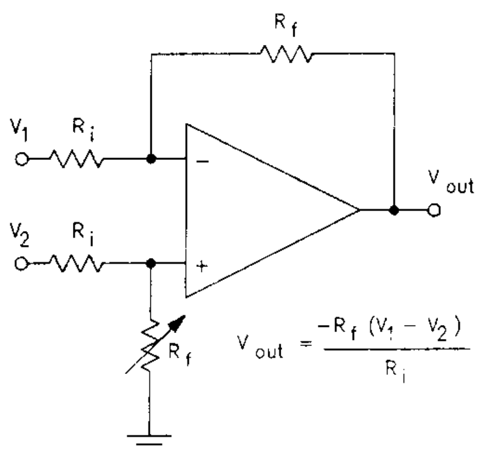
First. . . *some op-amp circuits*



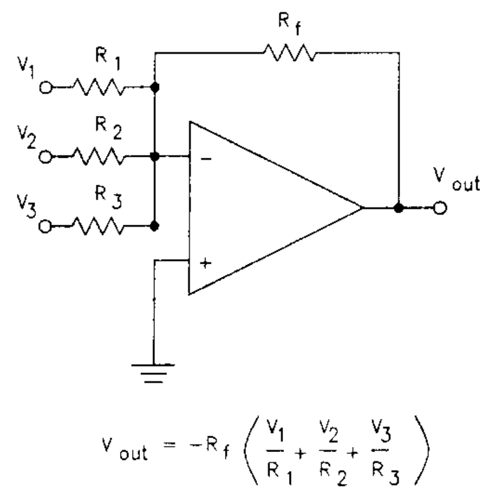
(a) Inverting Amplifier



(b) Noninverting Amplifier



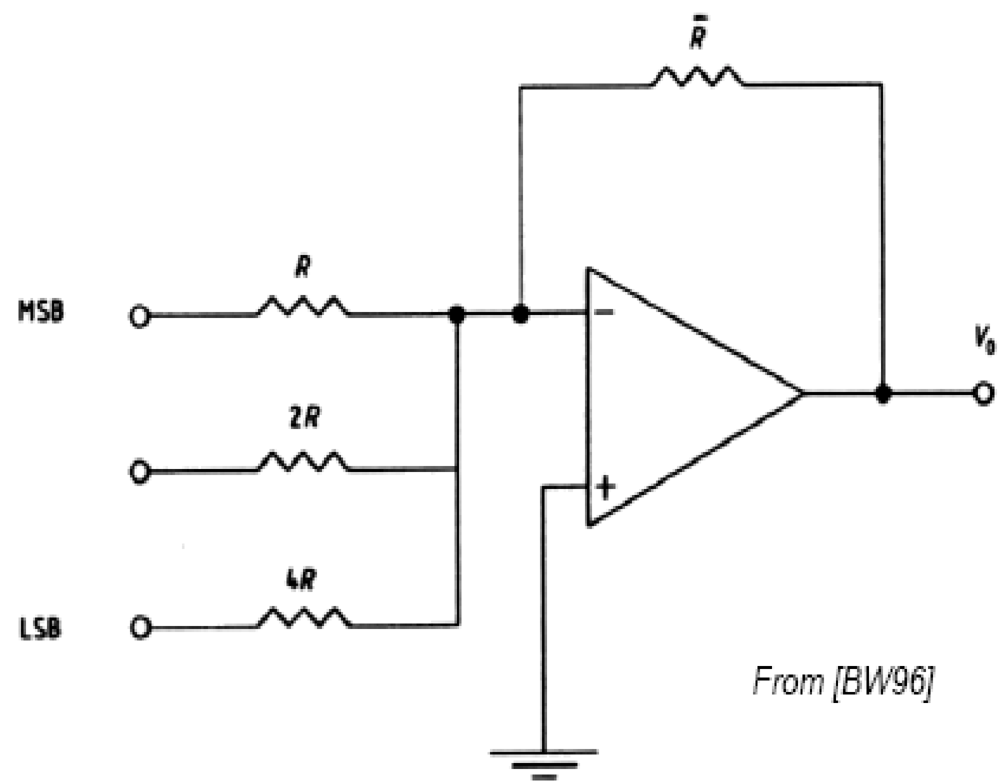
(c) Differential Amplifier



(d) Summing Amplifier

Binary weighted ladder

- Based on the Summing Amplifier
- One bit per input
- Each input resistor is 2X the previous value
- $V_{RefHigh}$ if input bit is 1, otherwise ground
- 3-bit example:



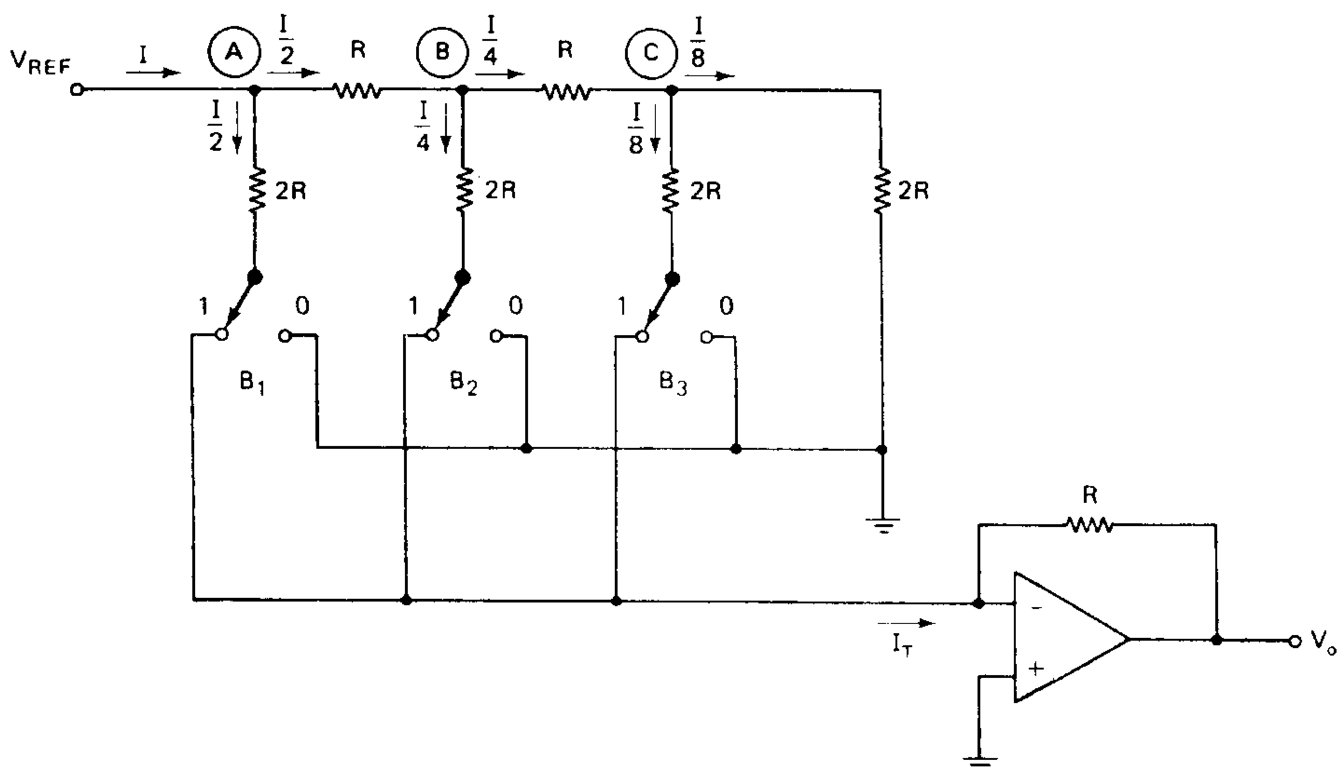
From [BW96]

Binary weighted ladder

- Fast conversion, *but . ..*
- Not practical for large *n* because many different high-accuracy resistors are needed
- The lowest-value resistor *R* affects the MSB and must have the highest accuracy
- “usually limited to 8-bit resolution or less”

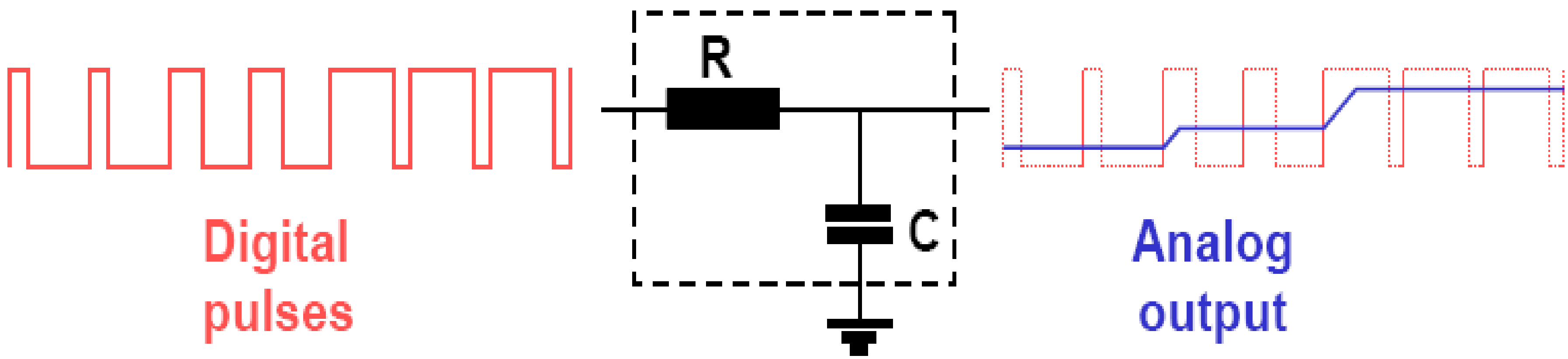
R-2R ladder

- Uses 2*n* resistors, with only two resistance values
- Somewhat slower than the previous case, because the RC effects increase for each additional R-2R stage
- 3-bit example:



PWM technique

- Simplest type of DAC
- Digital system generates a PWM waveform, selecting pulse widths based on the binary values being converted to analog form
- The waveform is input to a low-pass analog filter



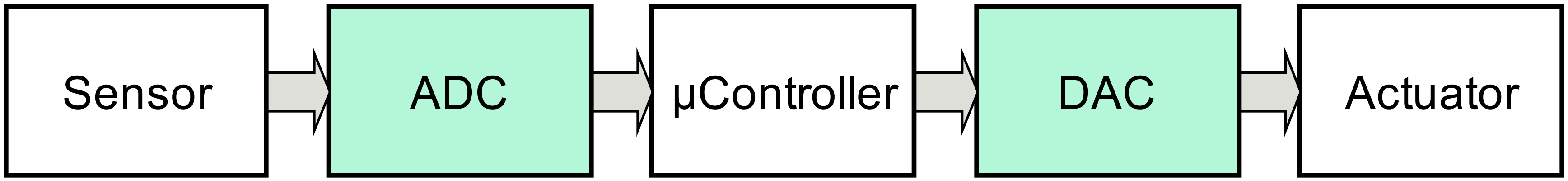
PWM technique

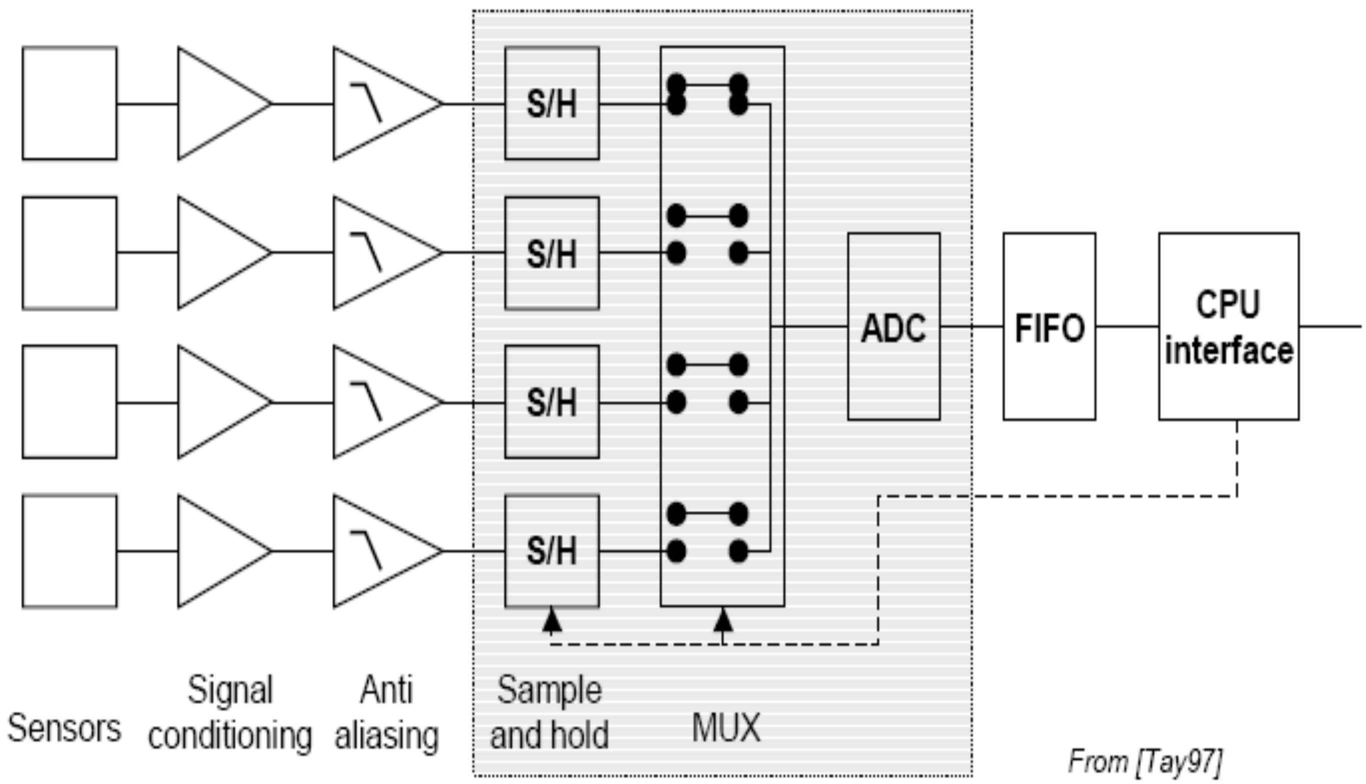
- Often used for electric motor speed control
- Now becoming common in high-fidelity audio

Outline

- Motivation
- Basic idea
- ADC on the PIC32
- Digital-to-analog conversion methods
- Analog-to-digital conversion methods

System overview, again

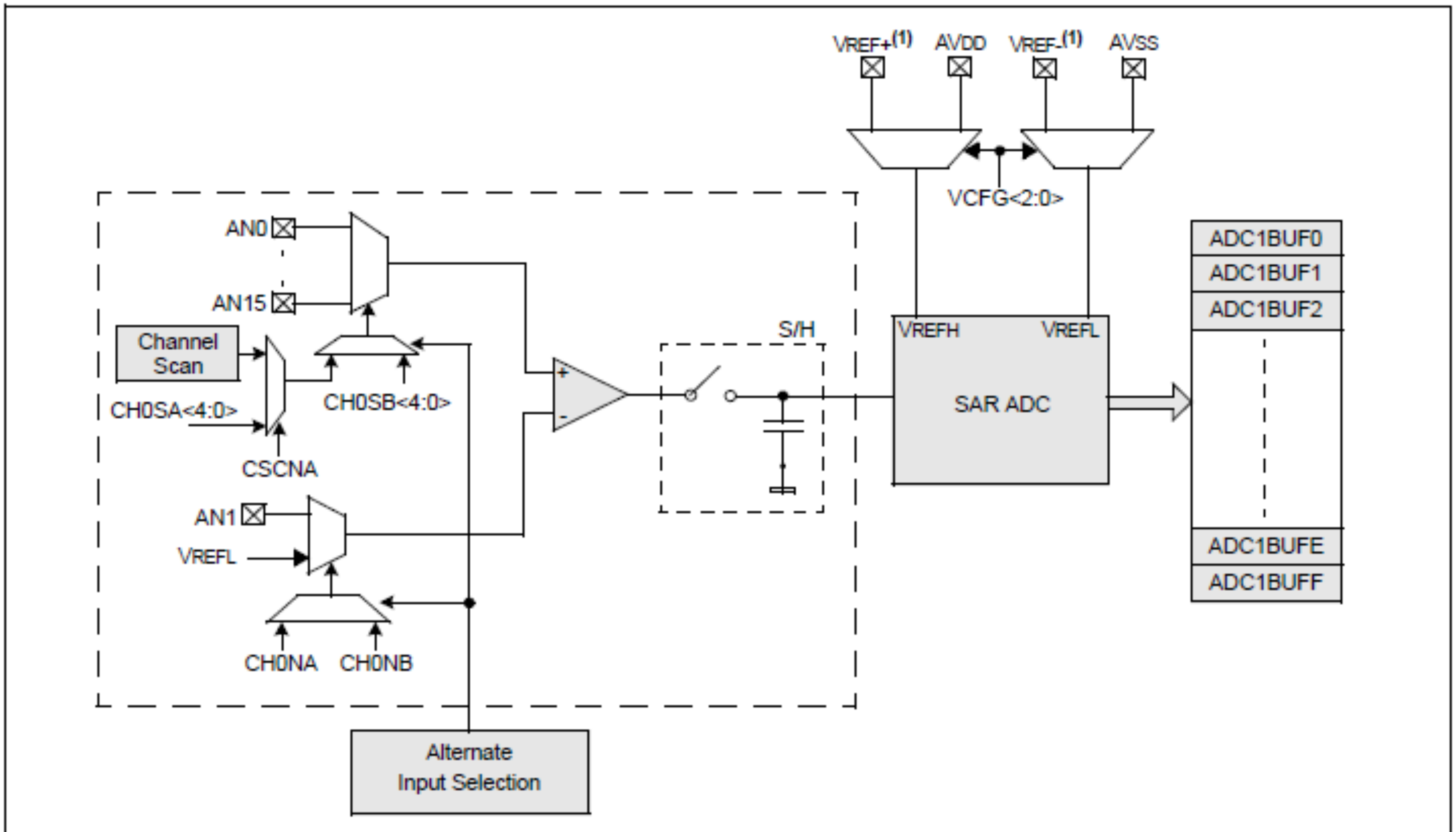




From [Tay97]

Comparision with PIC32

FIGURE 22-1: ADC1 MODULE BLOCK DIAGRAM

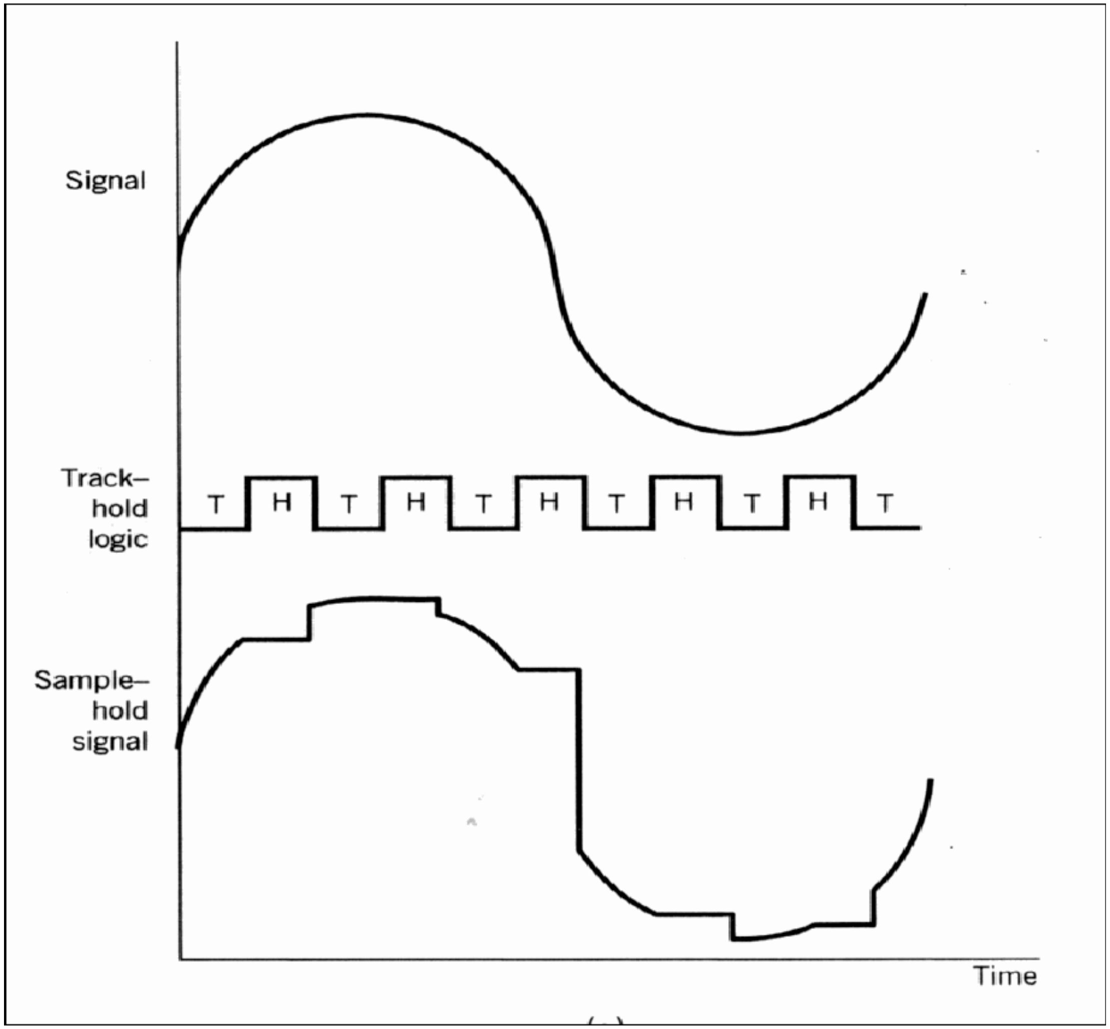


Note 1: VREF+ and VREF- inputs can be multiplexed with other analog inputs.

- *Signal conditioning*
  - Many sensors provide noisy, weak signals (e.g., a few mV or uV)
  - Often need to amplify, filter, linearize
- *Antialiasing*
  - For time-varying signals, often use LPF to make sure the Nyquist criterion is satisfied

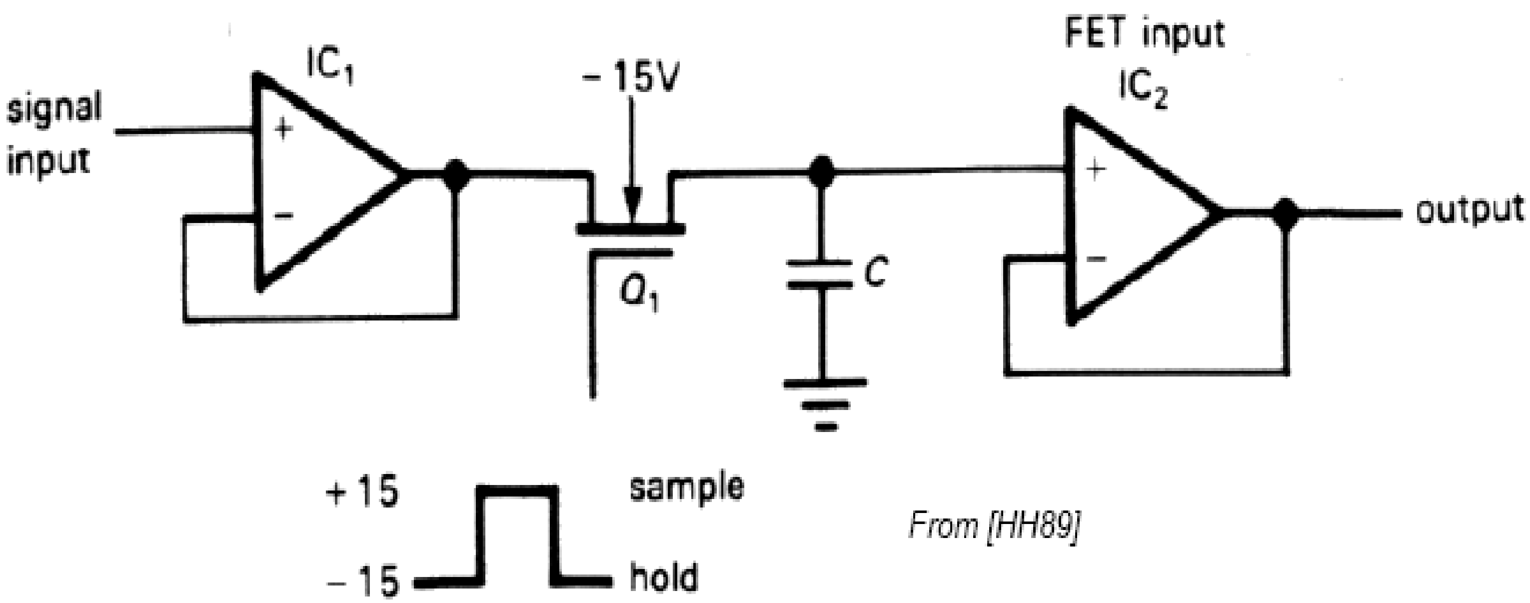
Sample and hold circuits

- *Two basic operating modes*
  - **Sample mode**: The S/H output follows the input ("Acquisition time")
  - **Hold mode**: Try to hold the output constant ("Conversion time")
- *Goal:*
  - Provide a steady voltage level to the ADC during conversion



Example S/H circuit components

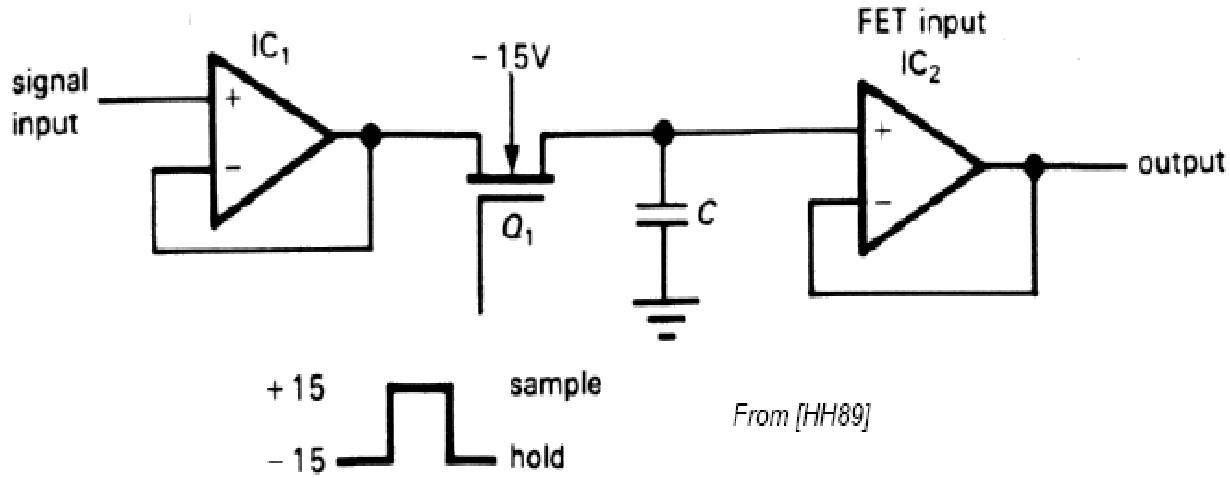
- FET-based switch
- Capacitor to hold voltage steady
- Voltage followers as buffers



From [HH89]

Main S/H circuit components

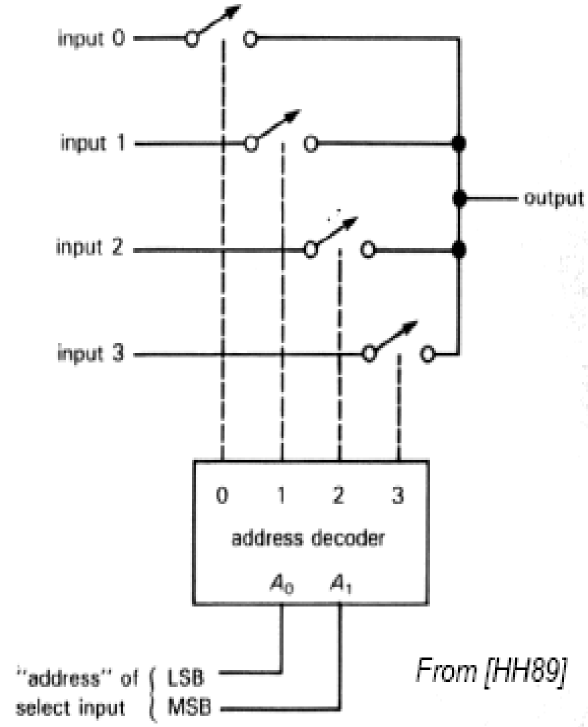
- **Components**
  - FET-based switch
  - Capacitor to hold voltage steady
  - Voltage followers as buffers



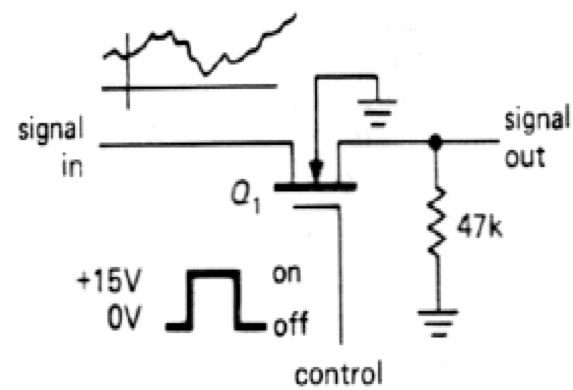
- **Operation**
  - IC1 provides low Zout version of input signal
  - FET passes the signal during 'sample' and disconnects during 'hold'
  - C preserves the value during 'hold'
  - IC2 is a high Zin op-amp to minimize capacitor discharge during 'hold'

Multiplexers

- **MUX**: a device that outputs one signal that is selected from several input signals
- **FET-based analog switches**
  - N-channel enhancement-mode MOS-FET
  - When gate is grounded or negative, the FET is nonconducting
  - Drain-source resistance in the order of 10,000 MΩ
  - Bringing the gate to +15V puts the drain-source channel into conduction
  - Drain-source resistance in the order of 100Ω



From [HH89]



From [HH89]

Given a stable analog voltage, how to perform A/D conversion?

- Goal
  - **Convert an analog voltage level to a digital value**
  - Usually 8 to 16 bits
  - Common input range: a few volts, such as 0 to V_{DD} or –12 V to +12 V
- Normally use a linear mapping
- Common concerns
  - Resolution
  - Accuracy
  - Conversion speed

Analog-to-Digital Conversion

- Typical conversion

$$D = \left(\frac{V_{In} - V_{RefLow}}{V_{RefHigh} - V_{RefLow}} \right) \times 2^n$$

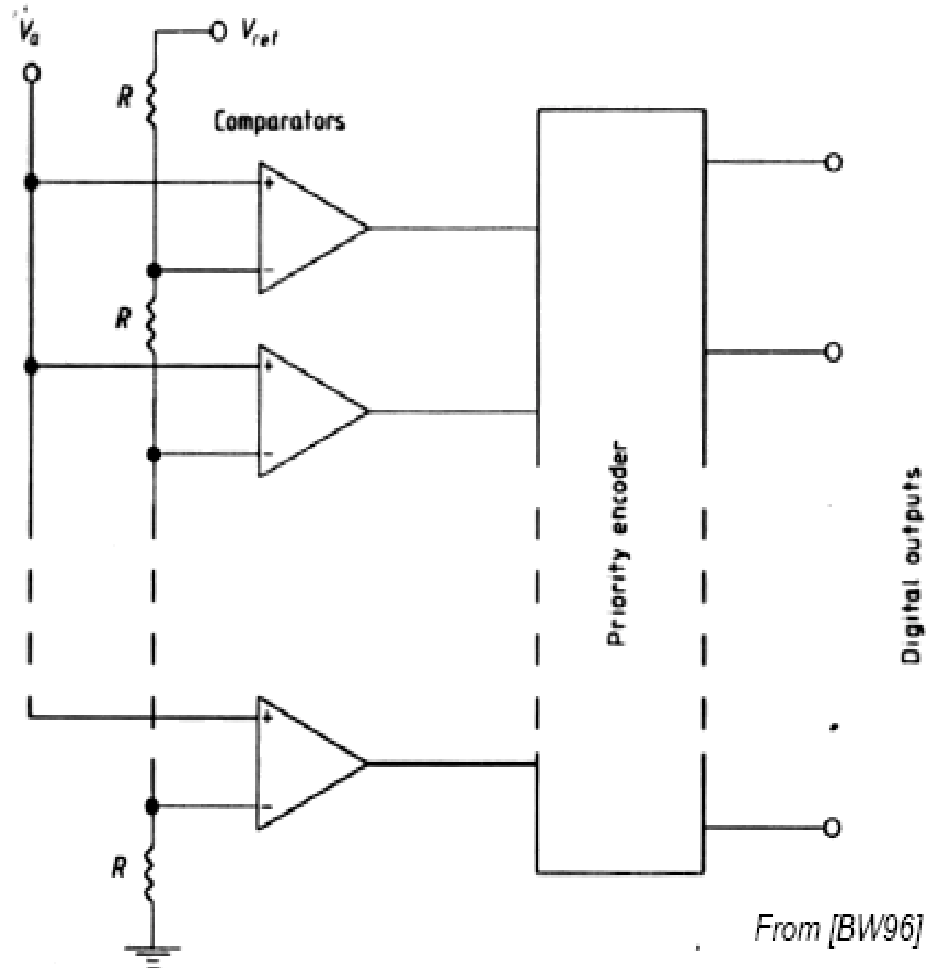
  - V_{In} = analog input voltage
  - D = binary output value
  - n = number of bits
  - $V_{RefHigh}$ = upper reference voltage
  - V_{RefLow} = lower reference voltage

Methods for A/D conversion

- Flash converter
- Counting converter
- Successive approximation

Flash (parallel) conversion

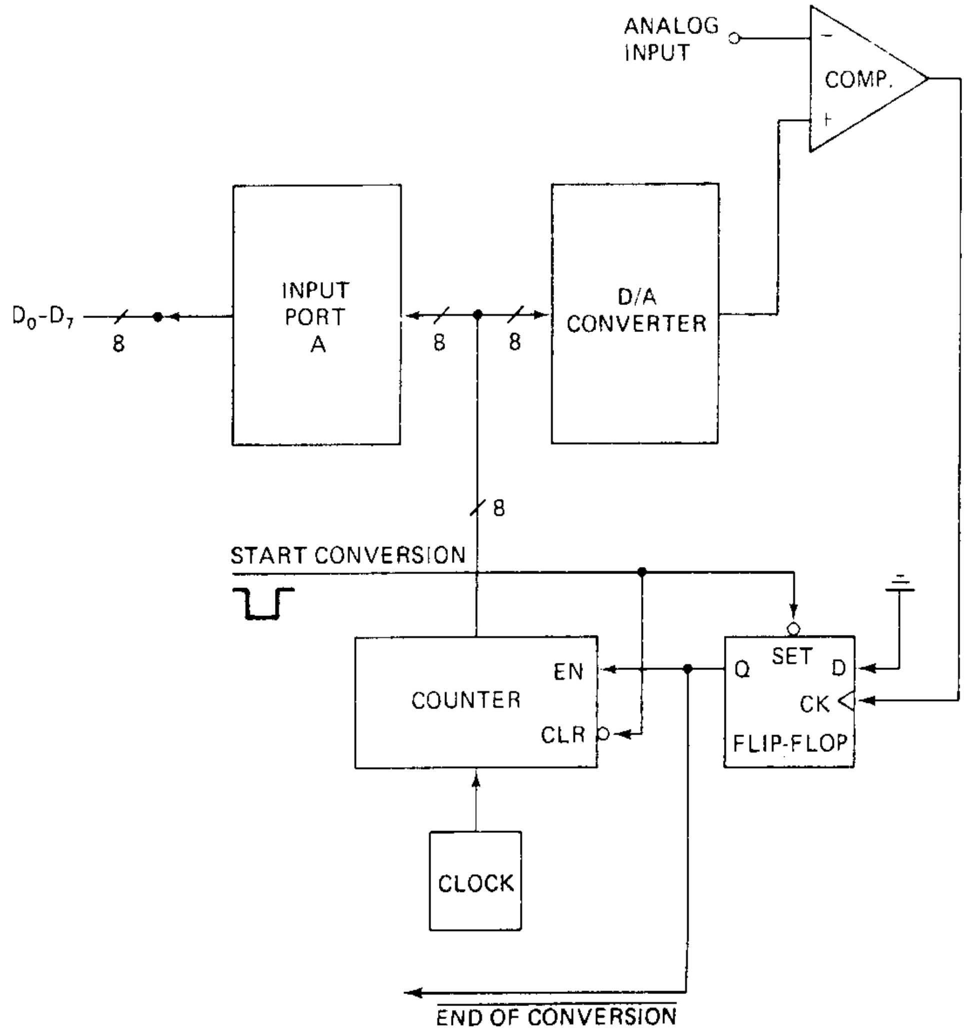
- Use 2^n comparators
- Priority encoder
- Fast & *expensive*



From [BW96]

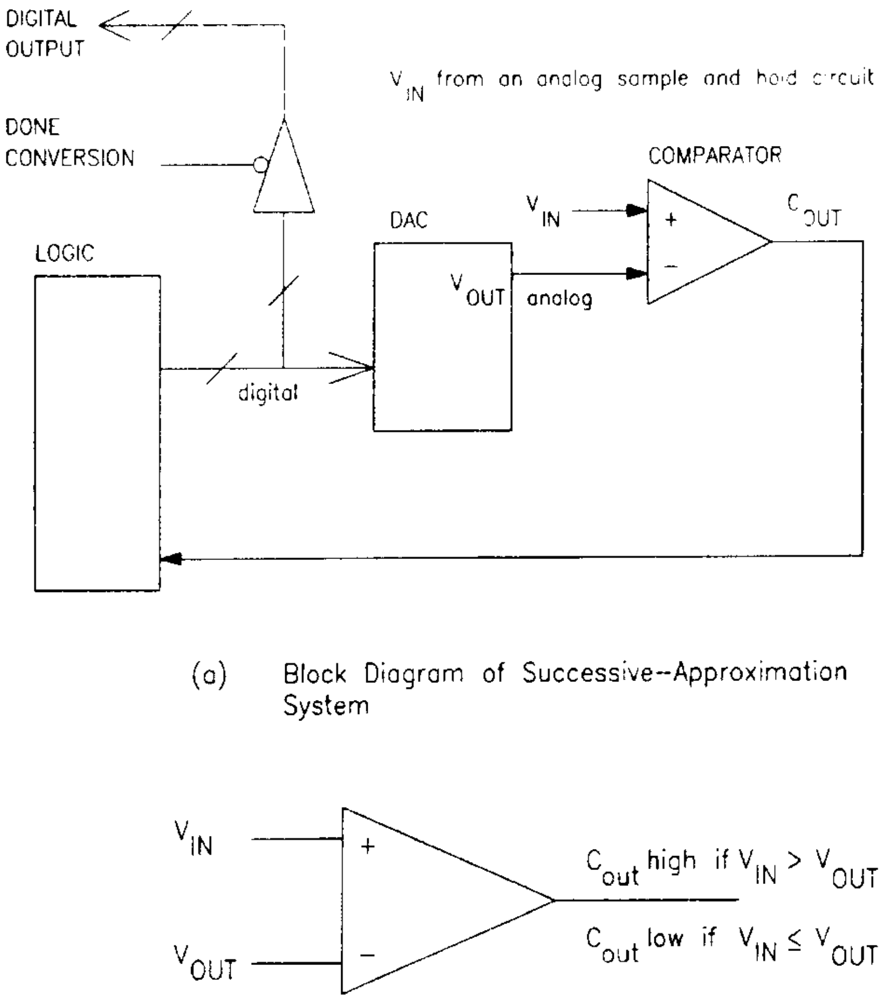
Counting conversion

- Use a DAC
- Count up until DAC output matches the input voltage
- *Slow*



Successive approximation

- Use a DAC
- Do a *smarter* search for the best digital value

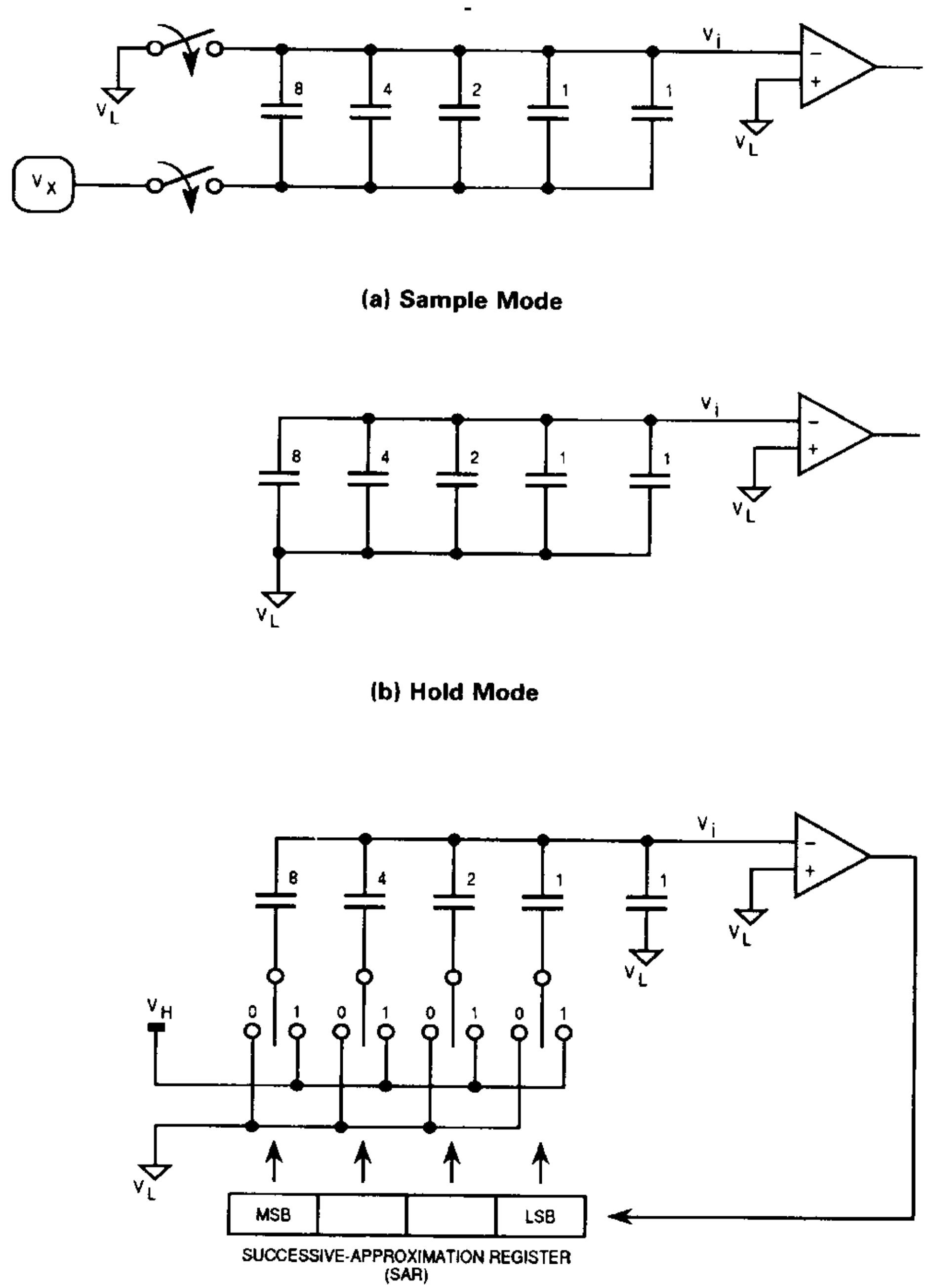


(a) Block Diagram of Successive--Approximation System

(b) Operation of Comparator

Capacitive ladder

- A form of successive approximation
- First try MSB, then next bit, then next, . . .



(a) Sample Mode

(b) Hold Mode

(c) Approximation Mode

Outline

- Motivation
- Basic idea
- ADC on the PIC32
- Digital-to-analog conversion methods
  - Binary weighted ladder
  - R-2R ladder
  - Pulse-width modulation
- Analog-to-digital conversion methods
  - Flash converter
  - Counting converter
  - Successive approximation