EEL 4744C: Microprocessor Applications

Lecture 10

Part 1
Analog Input & Output

Digitizing Signals

• To process continuous signals as functions of time

• Advantages
  – free from noise (e.g. error resilient)
  – can be stored, retrieved, and manipulated by computer (e.g. speech recognition)

• Disadvantages
  – loss of information
  – higher bandwidth requirement for transmission

Data Acquisition System

Transducer

• Transducers convert analog input to electrical signals (e.g. voltages or currents)

Signal Conditioning

• Signal conditioning: electrical isolation and buffering (e.g. protect from static discharges), amplification (e.g. produce necessary voltage for ADC), and bandwidth limiting (e.g. low-pass filter to limit range of frequencies for digitization)
**Analog Multiplexer**

- Analog MUX: select from several analog inputs

**Sample-and-Hold**

- Sample-and-hold circuit holds signal steady while ADC converts it
  - High-quality capacitor and high-speed semiconductor switch
  - Close switch for very short period, let capacitor charge, then switch opened and voltage held for ADC during its conversion time
  - May be included within ADC device

**A/D Converter**

- ADC converts sampled signal to digital values; has a word-size just like other digital devices

**Sampling Theorem & Aliasing**

- Shannon’s Theorem: sampling frequency must be at least twice the signal frequency
  - If not followed, then undersampled, which leads to aliasing and invalid signal reconstruction
  - Signal conditioning circuit may include antialiasing filter, helps here by attenuating frequencies above \( \frac{1}{2} \) the sample freq.
  - Nyquist frequency: the maximum frequency one can sample without aliasing (Nyquist frequency = \( \frac{f_{sample}}{2} \))

**A/D Conversion**

- Normally interfaced to rest of system via parallel I/O interface circuit or parallel input port
  - START_CONVERT control signal asserted by CPU to begin the conversion
  - END_CONVERT informs CPU when conversion complete (use via polling or interrupt)

**A/D Converter Types**

- Successive approximation A/D
  - Starting at MSB, each bit in reg. tested in succession with DAC output compared vs. input
  - With each bit tried: DAC output lower than input signal \( \Rightarrow \) bit left set; higher \( \Rightarrow \) bit is reset
  - However, N bit-times needed to set and test all the bits in succession
**Tracking A/D Converter**

- Has an up-down counter controlled by comparator
- If input signal higher or lower than DAC output, counter counts up or down, respectively
- Converter may quickly converge to correct digital value when signal not changing quickly

---

**Integrating A/D Converter (Dual slope ADC)**

- Converter integrates input signal for period $T_1$; afterwards, switch changed to minus ref. voltage and integrator discharges to zero at constant rate; time to discharge is period $T_2$ and gives a digital value

---

**Parallel A/D Converter**

- An array of $2^N - 1$ parallel comparators that quickly produces output code in prop. time of comparators plus encoder/decoder logic; fast but costly

---

**Two Stage Parallel A/D Converter**

- Has nearly the speed of flash ADC but less costly

---

**ADC Specifications**

- Conversion time: time required to complete conversion of input signal
  - Implies limit on signal freq. for sampling w/o aliasing ($f_{max} = \frac{1}{2} \cdot f_{sam} = 1/(2 \cdot x \cdot C \cdot T)$)
  - e.g. For ADC with $C \cdot T = 100ms \Rightarrow f_{sam} = 10kHz \Rightarrow f_{max} = 5kHz$.
    (max. freq. that can be converted w/o aliasing)

---

**ADC Specifications**

- Resolution: several forms of term in use, such as # of bits in ADC, smallest input signal for which ADC will produce a digital code
  - Full-scale signal $\pm 2^N$
  - e.g. 8-bit ADC on 5V full-scale signal $\Rightarrow$ res. is $5V/256 = 19.5mV$ (0.4% of full scale value)
### ADC Specifications

- **Accuracy**: ratio of smallest signal to measured signal; in %, describes how close measurement is to actual value  
  - e.g. 8-bit ADC on 5V full-scale signal, accuracy w/ 50mV signal is 19.5/50 = 39%

- **Linearity**: deviation in output codes vs. line from zero to full-scale; best is ± ½ of LSB

![Linearity Diagram](image)

### ADC Errors

- Quantization error is fundamental in A/D conversion due to resolution of ADC  
  - Can be no less than ± ½ LSB

- Other sources of errors include:  
  - Noise (desire peak-to-peak noise to be < ± ½ LSB; choose appropriate ADC resolution or reduce the signal noise)
  - Aliasing (include low-pass filter to attenuate freqs. above Nyquist freq)
  - Aperture time (if signal varies during this period, sample & hold circuit achieve short aperture time)

### ADC Selection

- Choose # of bits or resolution, speed or conversion time, type of digital code produced, etc

- Two ways to choose ADC resolution:  
  - Find dynamic range of input signal to choose # of bits  
    - \( DR = \frac{V_{\text{max}}}{V_{\text{noise}}} \) where \( V_{\text{max}} \) is max. input signal, \( V_{\text{noise}} \) is peak-to-peak noise  
    - For noise within ± ½ LSB, \( N \geq \log_2 DR \)
  - Choose based on resolution required in signal  
    - \( N \geq \log_2 \frac{V_{\text{max}}}{V_{\text{min}}} \), where \( V_{\text{min}} \) is required resolution

- Example: consider ADC for ±5V peak-to-peak range, 5mV peak-to-peak noise, and \( f_{\text{max}} = 3\text{kHz} \):  
  - \( DR = 10V / 5mV = 2000 \)  
  - \( N \geq \log_2 2000 \geq 10.9 \Rightarrow N = 11 \) or more  
  - Maximum conversion time is \( 1 / (2\times3\text{kHz}) = 167\text{ms} \)
D/A Conversion

- DAC diagram
  ![DAC Diagram](image)
  
  - Latch may be part of DAC or must be separately interfaced
  - Signal cond. block may be used to filter and smooth the quantized output, and perhaps also isolation, voltage amplification, etc.
  - Example of quantized DAC output waveform
    ![Example Waveform](image)

D/A Converter Types

- Basic DAC circuit is binary-weighted register DAC
  - weighted current supplied to summing junction of amplifier
    ![Basic DAC Circuit](image)

D/A Converter Types

- R-2R ladder DAC circuit using single-pole, double-throw switches between ground and reference
  - binary-weighted current supplied to summing junction
    ![R-2R Ladder DAC](image)

DAC Specifications

- Resolution and linearity (same as before but w.r.t. output voltage)
- Settling time: time for output voltage to settle within specified error band (e.g. ± ½ LSB)
  ![Settling Time](image)

DAC Specifications

- Glitches: high-speed DACs may have problems with glitches and settling time
  - Glitch caused by asymmetrical switching in D/A switches (e.g. 1→0 switch faster than 0→1 switch ⇒ glitch)
    ![Glitch Example](image)
  - Can be eliminated w/ sample-and-hold on DAC output, strobed to sample data after glitch and settling time
    ![Sample-and-Hold](image)
EEL 4744C: Microprocessor Applications

Lecture 10

Part 2

M68HC12 Analog Input

Read Assignment

- SHE (old version): Chapter 12
- SHE (new version): Chapter 17
- Freescale HC12 Data Sheet: Chapter 17

Introduction

- 8-channel, multiplexed, 8-bit, successive approx. ADC with sample-and-hold
- Linear to ±1 LSB accuracy in full temperature range w/ no missing codes
- Both conversion time and S&H aperture time are programmable
- Uses VRH and VRL to optimize resolution over input signal range
- VRH usually set to input signal max. (but must be ≤ 6V); VRL to min. (but must be ≥ 0V)

HC12 ADC Block Diagram

- A/D derives clock from P-clock; prescalar bits PRS4-PRS0 for +2 (fastest) to +16 (slowest)
  - See Table 12-1
- Limits: max. and min. conversion freqs. permitted are 2MHz and 500kHz, respectively
- Final sample time stage selectable from 2-16 clock periods (2,4,8,16) via SMP1:SMP0 bits
  - See Table 12-2

HC12 ADC Block Diagram

- Signal range must be VRH – VRL > 2.5V
- Resolution is (VRH – VRL)/256
  - e.g. (5V-0V)/256 = 19.5mV
- ADPU bit to enable subsystem (1=enabled); delay of ~100ms needed afterwards before use
- AWAI bit to have ATD stop/continue when HC12 in wait mode (1=stop)
**Sampling & Conversion Timing**

ATDCTL4 register controls the A/D Timing

![Diagram showing ATDCTL4 register controls the A/D Timing]

**A/D Input MUX and Scanning**

- 8 input channels, selected by bits in ATDCTL5 register (The ADC is started by writing to ATDCTL5)
- ADC always completes sequence of either 4 or 8 conversions (chosen by S8CM bit)
- SCAN bit controls whether ADC converts only 1 sequence versus continuously
  - SCAN=0: After the conversions are done, the A/D waits for the program to write to the ATDCTL5 again
  - SCAN=1: A/D starts another conversion cycle immediately

**A/D Input Synchronization**

- ADC can generate interrupts when conversion sequence complete, or user may poll flag
- SCF (sequence complete flag) bit set when the 4/8 conversion sequence is done
- Also, 8 conversion complete flags (CCF7-CCF0) associated with the A/D result registers
  - Set when current conversion writes into associated result register
- These flags contained in 16-bit ATDSTAT status register

**A/D Interrupts**

- Can generate interrupt when current 4/8 conversion sequence is completed
- ASCIE bit used to enable, ASCIF is the flag that generates the interrupt
  - Cleared by reading any result register when AFFC=1
  - A/D seq. complete vector is in the vector table
A/D Programming Example

- A/D programming example: converts the data on Ch 4-7
  - P-clock is 8MHz
  - Prescaler set for P-clock * 4
  - ATD clock is 2MHz (its maximum)
  - 2-clock final sample time used \( \Rightarrow 16 + 2 = 18 \) ATD clock cycles in total conversion time
  \[ \Rightarrow f_{\text{conv}} = \frac{1}{(18 \times 500\text{ns})} = 111.11 \text{ kHz} \]
  \[ \Rightarrow f_{\text{max}} = 55.5 \text{ kHz} \] (i.e. Nyquist frequency)

A/D Programming Example

: AD control registers
  ATDCTL2: EQU $62
  ATDCTL4: EQU $64
  ATDCTL5: EQU $65
  ADR1H: EQU $72
  ADR3H: EQU $74
  ADRSH: EQU $76
  ATDSTAT: EQU $66
  SCF: EQU $10000000 ; Scan complete flag
  AFFC: EQU $01000000 ; Fast clear
  AWAI: EQU $00100000 ; A/D wait mode
  ASCIE: EQU $00000010 ; SCF interrupt enable
  ADPU: EQU $10000000 ; A/D power up bit
  SMP: EQU $01010000 ; SMP0 and SMP1 bits
  PRS0: EQU $00000001 ;

: A/D Mode: S8CM=0 4 conversion sequence
  : MULT=1 4 conversions on channels 4-7
  : CD,CC,CA0,CA1s Analog channel 4-7
  : ADMODE: EQU $00001000

A/D Programming Example

; Power up the A/D

  bset ATDCTL2,ADPU
  ; Generate a "short" delay > 100 microseconds
  delay:   nop       ; 800 clock cycles
  dbne a,delay

; Now set up the A/D

  ; Normal flag clearing, run in WAIT mode, no interrupts
  bclr ATDCTL2,AFFC|AWAI|ASCIE
  ; Select 2 clock sample time and divide by 4 prescaler:
  bclr ATDCTL4,SMP
  bset ATDCTL4,PRS0
  ; 2 MHz conversion freq
  ; Start the conversion by writing the scan select information to ATDCTL5
  loop:   ldab ADR0H ; Channel 4 is here
          staa ATDCTL5

; And wait until conversion done
  spin:   brclr ATDSTAT,SCF,spin
          clra
          ldab ADR1H ; Get the channel 4 value
          clra
          ldab ADR3H ; Channel 4 is here

Dr. Tao Li