Analog to Digital Converter

Denis BISSIERES
Ian CAMPBELL
Yohan LESPERAT

Mechatronics - Fall 04
Contents

• What is ADC?
• Types of ADCs
• HC11 & ADC
What is ADC?

- Definition
- Examples of use
- Conversion process
- Accuracy
Definition

- Most signals we want to process are analog
- i.e.: they are continuous and can take an infinity of values
Definition

- Digital systems require discrete digital data
- ADC converts an analog information into a digital information
Examples of use

- Voltmeter
  \[ \Delta V \rightarrow 7.77 \text{ V} \]

- Cell phone (microphone)
  Voice \rightarrow \text{Wave}
Conversion process

3 steps:
• Sampling
• Quantification
• Coding

These operations are all performed in a same element: the A to D Converter
Conversion process: Sampling

- Digital system works with discrete states
- The signal is only defined at determined times
- The sampling times are proportional to the sampling period ($T_s$)
The signal can only take determined values belonging to a range of conversion ($\Delta V_r$)

- Based on number of bit combinations that the converter can output
- Number of possible states: $N = 2^n$ where $n$ is number of bits
- Resolution: $Q = \frac{\Delta V_r}{N}$

**Conversion process: Quantification**

![Diagram of conversion process](diagram.png)
Conversion process: Coding

- Assigning a unique digital word to each sample
- Matching the digital word to the input signal

\[ x_q(t) \]

\[ T_s \]

\[ \Delta V_r \]
The accuracy of an ADC can be improved by increasing:

- The sampling rate ($T_s$)
- The resolution ($Q$)
Accuracy

Higher Sampling rate

Higher Resolution

What is ADC?

Types of ADCs

HC11 & ADC
Nyquist-Shannon theorem: Minimum sampling rate should be at least twice the highest data frequency of the analog signal
\[ f_s > 2 \cdot f_{\text{max}} \]
What is ADC?

Types of ADCs

HC11 & ADC

Sampling rate

- Analog signals are composed of an infinity of harmonics
- Need to limit the frequency band to its useful part
- Use of an analog filter

In practice: \( f_s \approx (3\ldots5)f_{\text{filter}} \)
Example

- 8 bits converter: \( n=8 \)
- Range of conversion: \( \Delta V_r=5V \)
- Sampling time: \( T_s=1\text{ms} \)

- Number of possible states: \( N=2^8=256 \)
- Resolution: \( Q=\Delta V_r/N=19.5 \text{ mV} \)
- Analog Filter: \( f_{\text{filter}} \approx f_s/5 = 200 \text{ Hz} \)
Types of ADCs

- Flash ADC
- Sigma-delta ADC
- Dual slope converter
- Successive approximation converter
What is ADC?

- **Flash ADC**

  - “parallel A/D”
  - Uses a series of comparators
  - Each comparator compares $V_{in}$ to a different reference voltage, starting with $V_{ref} = 1/2$ lsb
Flash ADC

Comparator is one use of an Op-Amp

<table>
<thead>
<tr>
<th>If</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V_{IN} &gt; V_{REF}$</td>
<td>High</td>
</tr>
<tr>
<td>$V_{IN} &lt; V_{REF}$</td>
<td>Low</td>
</tr>
</tbody>
</table>
Flash ADC

**Advantages**
- Very fast

**Disadvantages**
- Needs many parts (255 comparators for 8-bit ADC)
- Lower resolution
- Expensive
- Large power consumption
Sigma-Delta ADC

- Oversampled input signal goes in the integrator
- Output of integration is compared to GND
- Iterates to produce a serial bitstream
- Output is serial bit stream with # of 1’s proportional to $V_{in}$
Sigma-Delta ADC

Advantages

• High resolution
• No precision external components needed

Disadvantages

• Slow due to oversampling
The sampled signal charges a capacitor for a fixed amount of time.

By integrating over time, noise integrates out of the conversion.

Then the ADC discharges the capacitor at a fixed rate while a counter counts the ADC's output bits. A longer discharge time results in a higher count.
Dual Slope converter

**Advantages**
- Input signal is averaged
- Greater noise immunity than other ADC types
- High accuracy

**Disadvantages**
- Slow
- High precision external components required to achieve accuracy
Successive Approximation

Is $V_{in} > \frac{1}{2}$ ADC range?

- Sets MSB
- Converts MSB to analog using DAC
- Compares guess to input
- Set bit
- Test next bit
Successive Approximation

**Advantages**
- Capable of high speed
- Medium accuracy compared to other ADC types
- Good tradeoff between speed and cost

**Disadvantages**
- Higher resolution successive approximation ADCs will be slower
- Speed limited \(\sim 5\text{Msps}\)

What is ADC?
Types of ADCs
HC11 & ADC
ADC Types Comparison

<table>
<thead>
<tr>
<th>Type</th>
<th>Speed (relative)</th>
<th>Cost (relative)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dual Slope</td>
<td>Slow</td>
<td>Med</td>
</tr>
<tr>
<td>Flash</td>
<td>Very Fast</td>
<td>High</td>
</tr>
<tr>
<td>Successive Approx</td>
<td>Medium – Fast</td>
<td>Low</td>
</tr>
<tr>
<td>Sigma-Delta</td>
<td>Slow</td>
<td>Low</td>
</tr>
</tbody>
</table>
HC11 & ADC

• Structure of the acquisition
• ADCTL Register
• Option Register
• Data conversion
• Acquisition
Port E (analog input)
8 channels

What is ADC?

Types of ADCs

HC11 & ADC

8-bits CAPACITIVE DAC WITH SAMPLE AND HOLD

SUCCESSIVE APPROXIMATION REGISTER AND CONTROL

VRH

VRL

RESULT REGISTER INTERFACE

ADR1

ADR2

ADR3

ADR4

ADCTL A/D CONTROL

INTERNAL DATA BUS

P 64 M68HC11 Family Data Sheet
Structure of the acquisition

- 8 channel/bit input
- VRL = 0 volts
- VRH = 5 volts
- Digital input on P

What is ADC?
Types of ADCs
HC11 & ADC
ADCTL Register

<table>
<thead>
<tr>
<th>ADCTL ($1030)</th>
<th>CCF</th>
<th>0</th>
<th>SCAN</th>
<th>MULT</th>
<th>CD</th>
<th>CC</th>
<th>CB</th>
<th>CA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reset to:</td>
<td>0</td>
<td>0</td>
<td>u</td>
<td>u</td>
<td>u</td>
<td>u</td>
<td>u</td>
<td>u</td>
</tr>
</tbody>
</table>

**MULT** - Single or multiple channel
- 0: Sample a single channel (four times)
- 1: Sample four channels

**CD,CC,CB,CA** - Channel selection
- If MULT is 0, then CC-CA bits specify the channel
- If MULT is 1, then CC specifies the group:
  - 0: Sample AN0-AN3, 1: Sample AN4-AN7
- CD is reserved for factory test use

**CCF** - Conversion Complete Flag
- Set when all four conversions are complete
- Cleared by writing to ADCTL - starts the next conversion

**SCAN** - Continuous scan mode
- 0: Take one set of four conversions and stop
- 1: Continually perform new conversions
## ADCTL Register

### ADR# Behavior

<table>
<thead>
<tr>
<th>ADR# Behavior</th>
<th>Single Channel (MULT = 0)</th>
<th>Multiple Channel (MULT = 1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single Conversion (SCAN = 0)</td>
<td>One channel converted 4 times consecutively. The results are stored in ADR1-ADR4</td>
<td>4 channels converted once. The results are stored in ADR1-ADR4</td>
</tr>
<tr>
<td>Continuous Conversion (SCAN = 1)</td>
<td>One channel is continuously converted. ADR1-ADR4 overwritten</td>
<td>4 channels are continuously converted. ADR1-ADR4 overwritten</td>
</tr>
</tbody>
</table>
Single Channel

What is ADC?

Types of ADCs

HC11 & ADC

A/D Converter

Result Register Interface

ADR1 – Result1
ADR2 – Result2
ADR3 – Result3
ADR4 – Result4
Multiple Channels

What is ADC?

Types of ADCs

HC11 & ADC

A/D Converter

Result Register Interface

ADR1 – Result1
ADR2 – Result2
ADR3 – Result3
ADR4 – Result4
**ADCTL Register**

Conversion Sequence

<table>
<thead>
<tr>
<th>E Clock cycles:</th>
<th>Sample (12)</th>
<th>Bit 7 (4)</th>
<th>6 (2) _ (2) 0 (2)</th>
<th>End (2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADCTL write (1)</td>
<td>1st, ADR1</td>
<td>2nd, ADR2</td>
<td>3rd, ADR3</td>
<td>4th, ADR4</td>
</tr>
<tr>
<td>0</td>
<td>32</td>
<td>64</td>
<td>96</td>
<td>128 total</td>
</tr>
</tbody>
</table>

What is ADC?

Types of ADCs

HC11 & ADC
ADCTL Register

A/D Result Registers (ADR1 – ADR4):

ADR1 = $1031
ADR2 = $1032
ADR3 = $1033
ADR4 = $1034
# ADCTL Register

## A/D Channel Assignment

<table>
<thead>
<tr>
<th>CD</th>
<th>CC</th>
<th>CB</th>
<th>CA</th>
<th>Channel Signal</th>
<th>1/2</th>
<th>Mult =1, ADF</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>PE0</td>
<td>ADR1</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>PE1</td>
<td>ADR2</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>PE2</td>
<td>ADR3</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>PE3</td>
<td>ADR4</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>PE4</td>
<td>ADR1</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>PE5</td>
<td>ADR2</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>PE6</td>
<td>ADR3</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>PE7</td>
<td>ADR4</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Reserved</td>
<td>ADR1</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>Reserved</td>
<td>ADR2</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>Reserved</td>
<td>ADR3</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>Reserved</td>
<td>ADR4</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>VH</td>
<td>ADR1</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>VL</td>
<td>ADR2</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1/2 VH</td>
<td>ADR3</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>Reserved</td>
<td>ADR4</td>
<td></td>
</tr>
</tbody>
</table>
ADCTL Register

ADR# Behavior

Single Channel  
(MULT = 0)  
The Channel is selected by CA, CB, CC

Multiple Channel  
(MULT = A)  
The group of Channels is selected by CC only
### Option Register

#### Options Register ($1039$)

<table>
<thead>
<tr>
<th>ADPU</th>
<th>CSEL</th>
<th>IRQE</th>
<th>DLY</th>
<th>CME</th>
<th>CR1</th>
<th>CR0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bit:</td>
<td>7</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>2</td>
</tr>
</tbody>
</table>

- **ADPU** = A/D power up
- **CSEL** = Clock Select
- **IRQE** = Config. IRQ
- **DLY** = Enable start-up delay
- **CME** = Clock Monitor
- **Bit 2** = not implemented
- **CR1** = COP Timer Rate
- **CR2** = COP Timer Rate

What is ADC?

Types of ADCs

HC11 & ADC
Option Register

ADPU - A/D Charge Pump

0: Turn off the A/D
1: Turn on the A/D (by enabling the charge pump)

Note: Wait at least 100 microseconds before using the A/D
(This is 200 cycles at a 2MHz E-clock)

CSEL - A/D Clock select

0: Use the E-clock for the A/D
1: Use a special internal A/D clock that runs at around 2MHz

Note: If the E-clock is 750KHz or higher, CSEL should be 0.
Otherwise CSEL should be 1.

DLY – Delay

0 = No delay is used and MCU resumes within approx. 4 cycles.
1 = 4000 E clock cycle delay imposed to allow crystal stabilization
## Data conversion

<table>
<thead>
<tr>
<th>% (1)</th>
<th>Bit 7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>Bit 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>50%</td>
<td>25%</td>
<td>12.5%</td>
<td>6.25%</td>
<td>3.12%</td>
<td>1.56%</td>
<td>0.78%</td>
<td>0.39%</td>
<td></td>
</tr>
</tbody>
</table>

| Volts (2) | 2.500 | 1.250 | 0.625 | 0.3125 | 0.1562 | 0.0781 | 0.0391 | 0.0195 |

| Volts (3) | 1.65 | 0.825* | 0.4125 | 0.2063 | 0.1031 | 0.0516 | 0.0258 | 0.0129 |

\[
x \times 2^{-1} \times 2^{-2} \times 2^{-3} \times 2^{-4} \times 2^{-5} \times 2^{-6} \times 2^{-7} \times 2^{-8}
\]

(1) %of VRH-VRL, (2) VRH=5 VRL=0, (3) VRH=3.3 VRL=0
What is ADC?

Types of ADCs

HC11 & ADC

Data conversion

MAX:
• .1111 1111 = .FF_{16} = 0.99609375_{10} = 99.6093 %

Resolution:
• .0000 0001 = .01_{16} = 0.00390625_{10} = 0.3906 %

MIN:
• .0000 0000 = 0_{16} = 0_{10} = 0 %
Data conversion

Some additional notes:

- $0V \leq$ analog input $\leq 5V$
- Charge pump allows VRH max 6-7V
- VRL and VRH convert to $00$ and $FF$
- Digital input of Port E pins not recommended during A/D sample time
**Acquisition**

<table>
<thead>
<tr>
<th>OPTION ($1039)</th>
<th>ADPU</th>
<th>CSEL</th>
<th>IREQ</th>
<th>DLY</th>
<th>CME</th>
<th>0</th>
<th>CR1</th>
<th>CR2</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADCTL ($1030)</td>
<td>CCF</td>
<td>0</td>
<td>SCAN</td>
<td>MULT</td>
<td>CD</td>
<td>CC</td>
<td>CB</td>
<td>CA</td>
</tr>
</tbody>
</table>

- **OPTION** EQU $1039
- **ADCTL** EQU $1030
- **ADR1** EQU $1031
- **ORG** $1040

```
LDAA #$80
STAA OPTION
LDY #$411A
NOP
NOP
NOP
LOOP
DEY
BNE LOOP
LDAA #$00
STAA ADCTL
LDX #ADCTL
WAIT
BRCLR 0,X #$80 WAIT
LDAA ADR1
PSHA
JSR $4000
SWI
END
```

- Delay for charge pump to stabilize 100µs
- ADPU=1, CSEL=0
- SCAN=0, MULT=0, CHAN GRP=00
- Wait until CCF or bit 7="1"
- Read and store result
Acquisition

Subroutine output to the screen the decimal equivalence of the stack:

```
ORG $4000
PULY
PULB
CLRA
LDX #$000A
IDIV
STAB $0000
XGDX
LDX #$000A
IDIV
STAB $0001
XGDX
TBA
```

Subroutines Outrhlf, convert to ASCII Number and output to screen
first the hundreds number, the tens and then units number

```
JSR $FFB5
LDAA $0001
JSR $FFB5
LDAA $0000
JSR $FFB5
PSHY
RTS
```

(Note: Remember that ACCA is the high byte of ACCD and ACCB is the low byte of ACCD. Return address used for JSR and RTS is stored in INDEX Y in the subroutine)

A number between 000 and 255 will be print on the screen. If it is 255 -> 100 %
000 -> 0 %
Questions ?