ADC
SUCCESSIVE APPROXIMATION METHOD

BY
SUBATHRA S

ADC - SUCCESSIVE APPROXIMATION METHOD

OBJECTIVE

To perform analog to digital conversion by interfacing Analog to Digital Convertor with 8085 Microprocessor using Successive Approximation Method.

APPARATUS REQUIRED

- 8085 Microprocessor Trainer kit
- Successive Approximation ADC (ALS-NIFC-07) Kit
- Power Supply
- RPS (0 – 5v)
- Flat Ribbon Cable

DESCRIPTION

The ADC interface consists of a NOR gate crystal oscillator, a CMOS clock driver which feeds 768Khz as the input clock to the ADC, a regulator (723) to connect the +12v to +5v required by the IC, a stable voltage reference(LM 336) and buffer (which provides +5v reference). A multi turn cermet allows adjustment to the reference voltage.

The channel select, ALE, start conversion and output enable lines are interfaced through port lines (connect a flat cable from the programmable peripheral interface 8255 connector on the trainer to the connector C_1 in the interface).

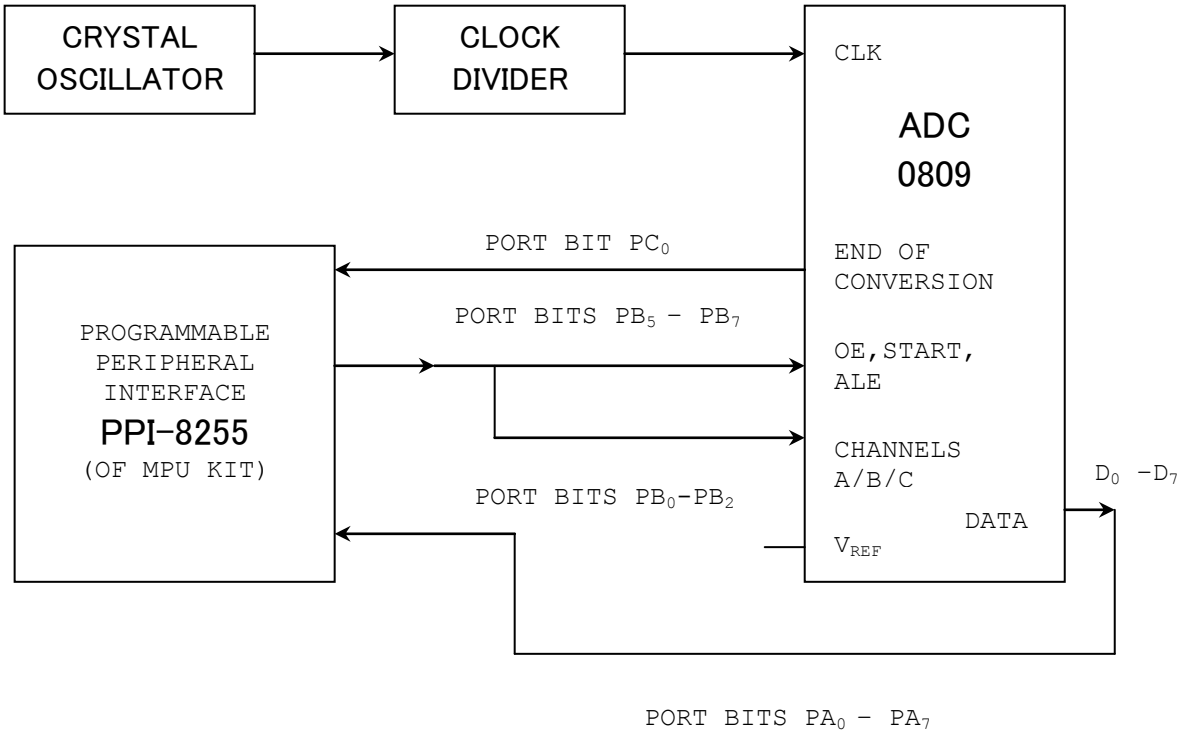
PORT LINES	DESCRIPTION
PA_0 - PA_7	Connect data line D_0 - D_7
PB_0	Channel select data lines A
PB_1	Channel select line B
PB_2	Channel select line C
PB_5	ALE to latch the address
PB_6	Start conversion
PB_7	Input Enable
PC_0	End of conversion

ALGORITHM

- Initialize the 8255 PPI in I/O operation.
- Give start of conversion by making PB6 high transition.
- PB7 is made high during the digital value.
- The digital value is sent to Port A and the digital value is displayed.

BLOCK DIAGRAM

NIFC 07
8 BIT SUCCESSIVE APPROXIMATION ADC INTERFACE



ALL PORT BITS ARE TERMINATED
IN A CONNECTOR P4

ASSEMBLY LANGUAGE PROGRAM

ADDRESS	LABEL	MNEMONICS	OPCODE/OPERAND
C200		**MVI A,91_H**	3E 91
C202		**OUT CWR**	D3 DB
C204	**NEXT**	**MVI A,60_H**	3E 60
C206		**OUT PORTB**	D3 D9
C208		**XRA A**	AF
C209		**OUT PORTB**	D3 D9
C20B		**MVI A,80_H**	3E 80
C20D		**OUT PORTB**	D3 D9
C20F		**MVI A,$E0_H$**	3E E0
C211		**OUT PORTB**	D3 D9
C213	**REPEAT**	**IN PORTC**	DB DA
C215		**RRC**	OF
C216		**JC REAPEAT**	DA 13 C2
C219		**MVI A,60_H**	3E 60
C21B		**OUT PORTB**	D3 D9
C21D		**IN PORTA**	DB D8
C21F		**STA $FFF9_H$**	32 F9 FF
C222		**CALL UPDDT**	CD D3 06
C225		**JMP NEXT**	C3 04 C2

PROGRAM TRACE

<table>
<tr><th>LABEL</th><th>MNEMONICS</th><th>DESCRIPTION</th></tr>
<tr><td></td><td>MVI A,91_H</td><td>Initializing the ports of the PPI 8255 as O/P ports by writing the control word as 91_H.
<table>
<tr><td>DATA</td><td>D_7</td><td>D_6</td><td>D_5</td><td>D_4</td><td>D_3</td><td>D_2</td><td>D_1</td><td>D_0</td></tr>
<tr><td>BITS</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>COMMENT</td><td>I/O mode</td><td colspan="2">Mode0</td><td>PortA I/P</td><td>PortC Upper O/P</td><td>Mode0</td><td>PortB O/P</td><td>PortC Lower I/P</td></tr>
</table>
91_H is moved to accumulator.
REGISTERS
<table>
<tr><td>A</td><td>91</td><td>XX</td><td>F</td></tr>
<tr><td>B</td><td>XX</td><td>XX</td><td>C</td></tr>
<tr><td>D</td><td>XX</td><td>XX</td><td>E</td></tr>
<tr><td>H</td><td>XX</td><td>XX</td><td>L</td></tr>
</table></td></tr>
<tr><td></td><td>OUT CWR</td><td>Control word specifies the I/O function for each ports of 8255.</td></tr>
<tr><td>NEXT</td><td>MVI A,60_H</td><td>60_H is loaded in to the Accumulator.
REGISTERS
<table>
<tr><td>A</td><td>60</td><td>XX</td><td>F</td></tr>
<tr><td>B</td><td>XX</td><td>XX</td><td>C</td></tr>
<tr><td>D</td><td>XX</td><td>XX</td><td>E</td></tr>
<tr><td>H</td><td>XX</td><td>XX</td><td>L</td></tr>
</table>
❖ DEFAULT OE SELECTION
Since the OE, START & ALE were connected to Port B of PPI bits B_7, B_6, B_5 through an inverter. Hence if we give ‘0’ as input, it will be complemented & the output ‘1’ is being generated at the other end.
<table>
<tr><td>DATA</td><td>B_7</td><td>B_6</td><td>B_5</td><td>B_4</td><td>B_3</td><td>B_2</td><td>B_1</td><td>B_0</td></tr>
<tr><td>BITS</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>COMMENT</td><td>OE</td><td>START</td><td>ALE</td><td colspan="3">X</td><td>C</td><td>B</td><td>A</td></tr>
</table></td></tr>
<tr><td></td><td>OUT PORTB</td><td>Output the 60_H configuration through port B.</td></tr>
<tr><td></td><td>XRA A</td><td>Clear the accumulator.
REGISTERS
<table>
<tr><td>A</td><td>00</td><td>XX</td><td>F</td></tr>
<tr><td>B</td><td>XX</td><td>XX</td><td>C</td></tr>
<tr><td>D</td><td>XX</td><td>XX</td><td>E</td></tr>
<tr><td>H</td><td>XX</td><td>XX</td><td>L</td></tr>
</table></td></tr>
<tr><td></td><td>OUT PORTB</td><td>Output the 00_H configuration through Port B.</td></tr>
</table>

<table>
<tr><td></td><td>MVI A,80_H</td><td>80_H is loaded in to Accumulator.
REGISTERS
A | 80 | XX | F
B | XX | XX | C
D | XX | XX | E
H | XX | XX | L

❖ START CONVERSION
In order to start conversion, we set bit START & ALE low and OE high initially. During hardware process, it will be complemented & output is processed.

<table><tr><td>DATA</td><td>B₇</td><td>B₆</td><td>B₅</td><td>B₄</td><td>B₃</td><td>B₂</td><td>B₁</td><td>B₀</td></tr><tr><td>BITS</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>COMMENT</td><td>OE</td><td>START</td><td>ALE</td><td colspan="3">X</td><td>C</td><td>B</td><td>A</td></tr></table></td></tr>
<tr><td></td><td>OUT PORTB</td><td>Output the 80_H configuration through Port B.</td></tr>
<tr><td></td><td>MVI A,E0_H</td><td>E0_H is moved to accumulator.
REGISTERS
A | E0 | XX | F
B | XX | XX | C
D | XX | XX | E
H | XX | XX | L

❖ DISABLE START, ALE, OE
Initially set START, ALE, OE high. Hence in the hardware process, it will be complemented & they are disabled.

<table><tr><td>DATA</td><td>B₇</td><td>B₆</td><td>B₅</td><td>B₄</td><td>B₃</td><td>B₂</td><td>B₁</td><td>B₀</td></tr><tr><td>BITS</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>COMMENT</td><td>OE</td><td>START</td><td>ALE</td><td colspan="3">X</td><td>C</td><td>B</td><td>A</td></tr></table></td></tr>
<tr><td></td><td>OUT PORTB</td><td>Output the E0_H configuration through Port B.</td></tr>
<tr><td>REPEAT</td><td>IN PORTC</td><td>Read data ready status.
Since Port C lower bits are configured as input bits (PC₃ -PC₀)</td></tr>
<tr><td></td><td>RRC</td><td>Rotate D₀ into carry flag.</td></tr>
<tr><td></td><td>JC
REAPEAT</td><td>If D₀ =1,conversion is not yet complete;
go back & check.</td></tr>
<tr><td></td><td>MVI A,60_H</td><td>60_H is loaded in to Accumulator.

REGISTERS
A | 60 | XX | F
B | XX | XX | C
D | XX | XX | E
H | XX | XX | L

❖ DEFAULT OE SELECTION
Since the OE, START, ALE are connected to Port B of PPI bits B_{7,} B_{6,} B₅ through an inverter. Hence if we give ‘0’ as input, it will be complemented & the output ‘1’ is being generated at the other end.

<table><tr><td>DATA</td><td>B₇</td><td>B₆</td><td>B₅</td><td>B₄</td><td>B₃</td><td>B₂</td><td>B₁</td><td>B₀</td></tr><tr><td>BITS</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>COMMENT</td><td>OE</td><td>START</td><td>ALE</td><td colspan="3">X</td><td>C</td><td>B</td><td>A</td></tr></table></td></tr>
<tr><td></td><td>OUT PORTB</td><td>Output the 00_H configuration through Port B.</td></tr>
</table>

	IN PORTA	Read output & save it in to the accumulator.
	STA FFF9$_H$	Store the result at data field using monitor routine location address.
	CALL UPDDT	Display at data location.
	JMP NEXT	Repeat the looping.

EXECUTION

ANALOG INPUT (volts)	**DIGITAL OUTPUT (hexa decimal)**
0.0	00_H
0.5	08_H
1.0	23_H
1.5	$3D_H$
2.0	54_H
2.5	75_H
3.0	93_H
3.5	$A8_H$
4.0	$C2_H$
4.5	$E2_H$
5.0	FF_H

REFERENCE

1. Ramesh S.Gaonkar, Microprocessor Architecture, Programming, and Applications, Fourth Edition, Penram International Publishing (India), 2000.

2. S.Subathra, “Advanced Microprocessor Laboratory”, Record work, Adhiparashakthi Engineering College, Melmaruvathur, October 2002

3. S.Subathra, “Programming in 8085 Microprocessor and its applications – An Innovative Analysis”, Technical Report, Adhiparashakthi Engineering College, Melmaruvathur, March 2003

4. Micro-85 EB, User Manual, Version – 3.0, CAT #M85 EB-002, VI Microsystems Pvt. Ltd., Chennai.