RELAY INTERFACE

BY
SUBATHRA S

RELAY INTERFACE

OBJECTIVE

To write an assembly language program to interface the relay with 8085 Microprocessor Trainer Kit.

APPARATUS REQUIRED

- 8085 Microprocessor Trainer Kit
- Relay Interface kit
- Flat Ribbon Cable
- Power Supply

DESCRIPTION

TO WORK WITH TWO CHANNEL RELAY

a. To work on this , only one FRC from P3 on kit to JP1 on interface is sufficient,

b. Connect +5 and Gnd. Supply with the help of four-way power mate.

c. Execute the program to switch ON and OFF the relay.

GO <STARTING ADDRESS> <EXEC>

In program if you make PC_0 high and PC_1 low of 8255, then (Terminal blocks) TB1's both terminals get shorted and TB2's both terminals are open By doing this , the channel which you made high remains shorted till we press RESET or till you should make that bit of the channel low(0).

To understand better take 2 LED's each in series with 330ohms resistors. Connect the pin 1 of both TB's(TB1 & TB2) V_{CC} and connect anodes of the LEDs to pin 2 of both the terminal block.

Now execute the program named two channel relay.

GO <STARTING ADDRESS> <EXEC>

Go<41AD> <. >.

When you execute the program the LEDs will be switched, correspondingly in the data field of kit. Observe that for 'ON' LED 1 will be displayed and for 'OFF' LED 0 will be displayed.

With this interface a EPROM (27128A of 16K) is provided, which contains all the programs fused, insert this EPROM in the extended socket U9 of the kit and execute the programs.

ALGORITHM

1.Initialize 8255 PPI Control word register.

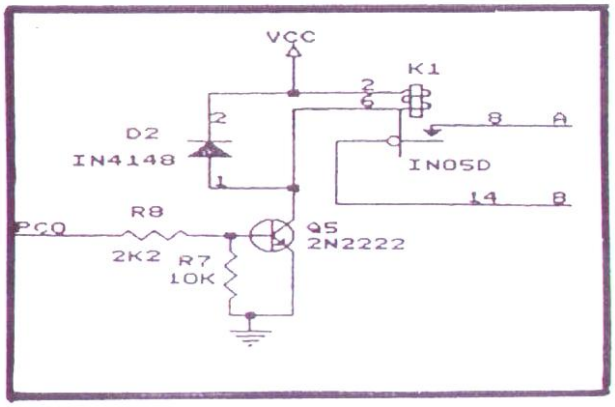
2.Send 01H through Port C to activate the relay and call delay subroutine program.

3.Send 00H through Port C to deactivate the relay.

4.Repeat steps 2 & 3.

CIRCUIT DIAGRAM

RELAY INTERFACING



ASSEMBLY LANGUAGE PROGRAM

ADDRESS	LABEL	MNEMONICS	OPCODE/OPERAND
C000		LXI SP,C300$_H$	31 00 C3
C003		MVI A,80$_H$	3E 80
C005		OUT CWR	D3 DB
C007		MVI A,01$_H$	3E 01
C009		OUT PORTC	D3 DA
C00B	LOOP1	CALL DELAY	CD 14 C0
C00E		CMA	2F
C00F		OUT PORTC	D3 DA
C011		JMP LOOP1	C3 0B C0
C014	DELAY	PUSH PSW	F5
C015		MVI D,0A$_H$	16 0A
C017	START	LXI B,208E$_H$	01 8E 20
C01A	LOOP2	DCX B	0B
C01B		MOV A,C	79
C01C		ORA B	B0
C01D		JNZ LOOP2	C2 1A C0
C020		DCR D	15
C021		JNZ START	C2 17 C0
C024		POP PSW	F1
C025		RET	C9

PROGRAM TRACE

<table>
<tr><th>LABEL</th><th>MNEMONICS</th><th>DESCRIPTION</th></tr>
<tr><td></td><td>LXI SP,C300$_H$</td><td>Initialize the Stack pointer at C300$_H$ .i.e. loads the 16-bit data in the register pair designated.
REGISTERS
A XX XX F
B XX XX C
D XX XX E
H XX XX L

STACK MEMORY
C300 XX ← Stack pointer
C2FF XX
C2FE XX
C2FD XX
C2FC XX</td></tr>
<tr><td></td><td>MVI A,80$_H$</td><td>Initializing the ports of the PPI 8255 as O/P ports by writing the control word as 80$_H$.

<table>
<tr><td>DATA BITS</td><td>D$_7$
1</td><td>D$_6$
0</td><td>D$_5$
0</td><td>D$_4$
0</td><td>D$_3$
0</td><td>D$_2$
0</td><td>D$_1$
0</td><td>D$_0$
0</td></tr>
<tr><td>COMMENT</td><td>I/O mode</td><td colspan="2">Mode0</td><td>PortA O/P</td><td>PortC Upper O/P</td><td>Mode0</td><td>PortB O/P</td><td>PortC Lower O/P</td></tr>
</table>

80$_H$ is moved to accumulator.
REGISTERS
A 80 XX F
B XX XX C
D XX XX E
H XX XX L</td></tr>
<tr><td></td><td>OUT CWR</td><td>Control word specifies the I/O function for each port of 8255.</td></tr>
<tr><td></td><td>MVI A,01$_H$</td><td>01$_H$ is moved to accumulator.
REGISTERS
A 01 01 F
B XX XX C
D XX XX E
H XX XX L</td></tr>
<tr><td></td><td>OUT PORTC</td><td>RELAY ON,OFF of two LEDs.

<table>
<tr><td>DATA BITS</td><td>PC$_7$
0</td><td>PC$_6$
0</td><td>PC$_5$
0</td><td>PC$_4$
0</td><td>PC$_3$
0</td><td>PC$_2$
0</td><td>PC$_1$
0</td><td>PC$_0$
1</td></tr>
</table>

TB1’s both terminal gets shorted & TB2’s both terminal remain open.</td></tr>
<tr><td>LOOP1</td><td>CALL DELAY</td><td>Call delay subprogram.</td></tr>
<tr><td></td><td>CMA</td><td>A=01 is complemented to get A=10.</td></tr>
</table>

		REGISTERS **A** **10** 10 F B XX XX C D XX XX E H XX XX L
	OUT PORTC	Output through Port C DATA BITS: PC_7 PC_6 PC_5 PC_4 PC_3 PC_2 **PC_1** **PC_0** = 0 0 0 0 0 0 **0** **1**
	JMP LOOP1	Repeat the process.
DELAY	**PUSH PSW**	PSW(Program Status Word) represents the content of the accumulator and flag register; the accumulator is the high order register and the flags are the low order register. **STACK MEMORY** C300 XX ← Stack pointer C2FF 01 **C2FE** **01** C2FD XX C2FC XX *NOTE:* *The content of the source register, flag are not altered after the PUSH instruction.*
	MVI D,$0A_H$	$0A_H$ is moved to D register. **REGISTERS** A 01 01 F B XX XX C **D** **0A** XX E H XX XX L
START	**LXI B,$208E_H$**	Initialize B register at $208E_H$ .i.e. loads the 16-bit data in the register pair designated. **REGISTERS** A 01 01 F **B** **20** **8E** **C** D 0A XX E H XX XX L **MEMORY** 208C 23 208D 11 **208E** **3A** ← BC Memory pointer 208F 33 2090 51
LOOP2	**DCX B**	Decrement the BC register pair. **REGISTERS** A 01 01 F **B** **20** **8D** **C** D 0A XX E H XX XX L

		MEMORY 208C 23 **208D 11** ← BC Memory pointer 208E 3A 208F 33 2090 51
	MOV A,C	$8D_H$ is moved to accumulator. **REGISTERS** **A 8D** 8D F B 20 8D C D 0A XX E H XX XX L
	ORA B	OR the B Register content with accumulator content.
	JNZ LOOP2	Jump if no zero to Labeled LOOP2.
	DCR D	Decrement the D register content by one. **REGISTERS** A 01 XX F B XX XX C **D 09** XX E H XX XX L
	JNZ START	Jump if no zero to Labeled START.
	POP PSW	**STACK MEMORY** **C300** XX ← Stack pointer C2FF 01 C2FE 01 C2FD XX C2FC XX The content of the source register, flag are retrieved from stack using POP instruction.
	RET	Return to main program.

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.