Chapter 9 LV-100 integrate with 8255 PPI

Overview

8255 is a component of programmable control input/output device, and it is generally taken as an interface for microprocessor or peripheral device. Generally speaking, if we want to expand the I/O port, then 8255 is the best choice. In this chapter, we will expand parallel port with addition of 8255 to expand the I/O port, and now let's tell you how to apply 8255 with parallel ports.

Objective

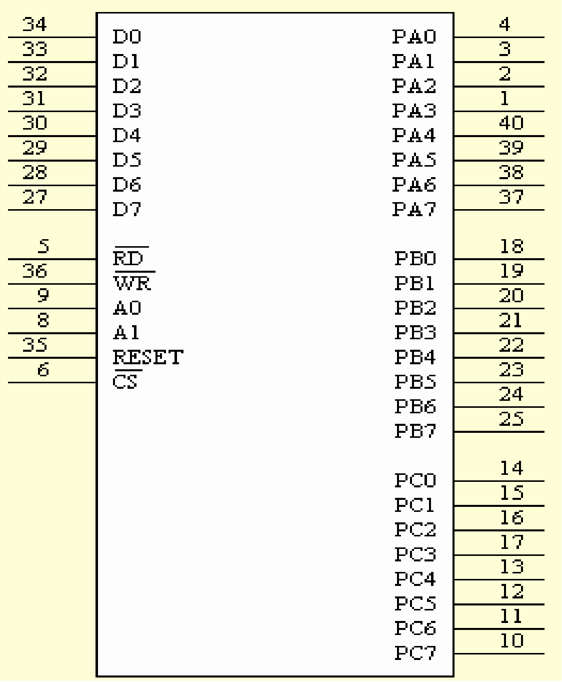
- To understand the control principle of 8255
- To study how to create sub .vi with LabVIEW
- To study how to uses the MODE 0 of 8255 with the parallel port to control the output

Keyword

- Programmable Peripheral Interface (PPI)
- Handshaking
- Integrated circuit (IC)

Brief

To make use of 8255 as the interface of the input/output, it is usually programmed as Mode 0. Under this kind of mode, there are three ports named A、B、C that had the simple function of input/output, port A has eight bits, port B still has eight bits, and port C has higher four bits and lower four bits, they are all controlled by program as input or output port. This chip of INTEL 8255 (Programmable Peripheral Interface, PPI for short) could programmed either as the basic I/O bi-directional mode, or as handshaking control mode for input or output. The so-called handshaking control is that both sides though some interface to transmit message, by handshaking control mechanism to decide whether any data still inside the interface in order to continue the transmitting or receiving. The following is the diagram of 8255 pin assignment.



◆ *The function of each pin is shown as follows.*

1、D0~ D7：Bi-directional tri-state data bus

2、RESET：Reset (reset at high level)

3、/CS：Chip select (select at low level)

4、/RD：Read enable

5、/WR：Write enable

6、A0~A1：Address select

7、PA0~PA7：Port A input/output

8、PB0~PB7：Port B input/output

9、PC0~PC7：Port C input/output

Data bus buffer：Tri-state bi-directional 8 bits device is used to deliver data, as an interfaces between the CPU and 8255. When the CPU execute the “read" or the “write" command, all the data or control command no matter what received or transmitted are all through this data bus buffer.

Read/ write control logic: This function is to control the transmission of the data and set of control word. When the CPU has received A0, A1,/ RD,/ WR,/ CS, and RESET signal etc., this control logic will initiate different kind of actions such as: RESET was initiated then control buffer will be cleared and PA,PB,PC, will be set as input port. Therefore the RESET pin of control logic must be connected with the RESET pin of the CPU together in order to keep consistence action of reset. The /CS, /RD and /WR are all low potential enable signal, when /CS=0 then 8255 is enabled, and the A0 and A1 are the input pins of addressing register which will cooperate with /RD, /WR to use and select three input/output registers and control word register.

◆ *Basic operation.*

A1	A0	/RD	/WR	/CS	Function explanation
0	0	0	1	0	PA port➤data bus(read PA port data)
0	1	0	1	0	PB port➤data bus(read PB port data)
1	0	0	1	0	PC port➤data bus(read PC port data)
0	0	1	0	0	Data write in PA port➤ (write data into PA port)
0	1	1	0	0	Data write in PB port➤ (write data into PB port)
1	0	1	0	0	Data write in PC port➤ (write data into PC port)
1	1	1	0	0	Data bus➤ control word register : functional setting
X	X	X	X	1	Data bus➤ at high impedance
1	1	0	1	0	forbidden
X	X	1	1	0	Data bus➤ at high impedance

◆ *8255A Operational description:*

8255 has three basic modes of operation:

§MODE-0—Basic Input/Output (PA、PB、PC).

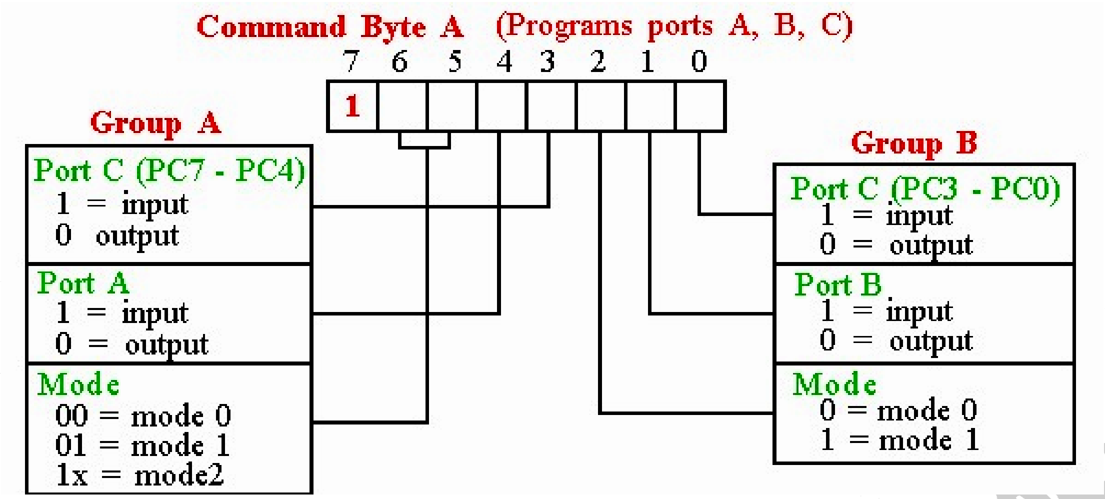
§MODE-1—Strobed Input/Output (PA、PB).

§MODE-2—Bi-directional bus(PA).

Note: The three above kinds of operation mode is completely set by the control word register.

Although 8255 has three kinds of operation mode, but we just interest in MODE-0 basic Input/Output on our experiment. How to begin? As we have mentioned to set the control word register first. Under the Mode-0, the each ports (PA、PB、PC) which we want to use as input or output must be programmed, those series of action are all set into the control word register of 8255.

■ *Set of control word*



Someone may ask where is the address of the control word register of 8255. There must be some specified address that pointed to those register. Sure, memory management distributes four address pointed to the registers of 8255, that is **00h assigned to PORT A**, **01h assigned to PORT B**, **02h assigned to PORT C**, and **03h assigned to the control word register**. Due to one single chip 8255 we use, so we don’t need any decoder to expand the address. But in some other books you will find that 80h assigned to PORT A, that is another IC to be enabled at the same circuits. At here we use one 8255, and it just bas address bus A0,A1 two lines which means four address 00h；01h；02h；03h available.

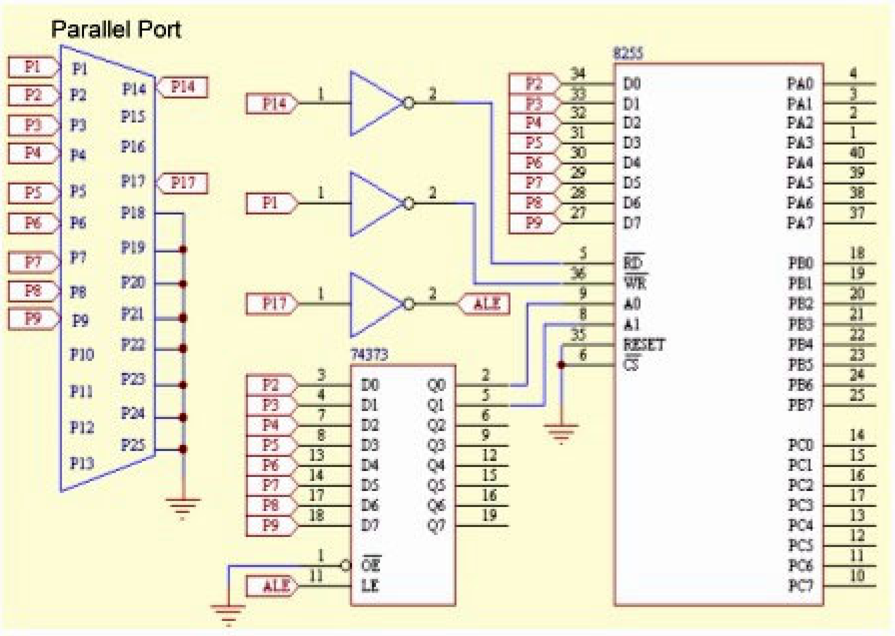
Therefore we want to design 8255 worked under the Mode-0 and the setting port A, port B, port C as output. From the set of control word table, the new control word is generated as 0x80 to fit the condition that we specify. Please remember that the address of control word register of 8255 is at 03h. Therefore before writing data we should define the address as 03h first, next we can send the data to the control word register of 8255.

The practice of this chapter is divided into three parts, the first one show you how to do a 8255 output's SubVI, the so-called SubVI is a subprogram. The second one show you how to do a 8255 input's SubVI, the third one show you how to apply a output's SubVI to complete the rolling LEDs. Let us start to connect the parallel port with 8255.

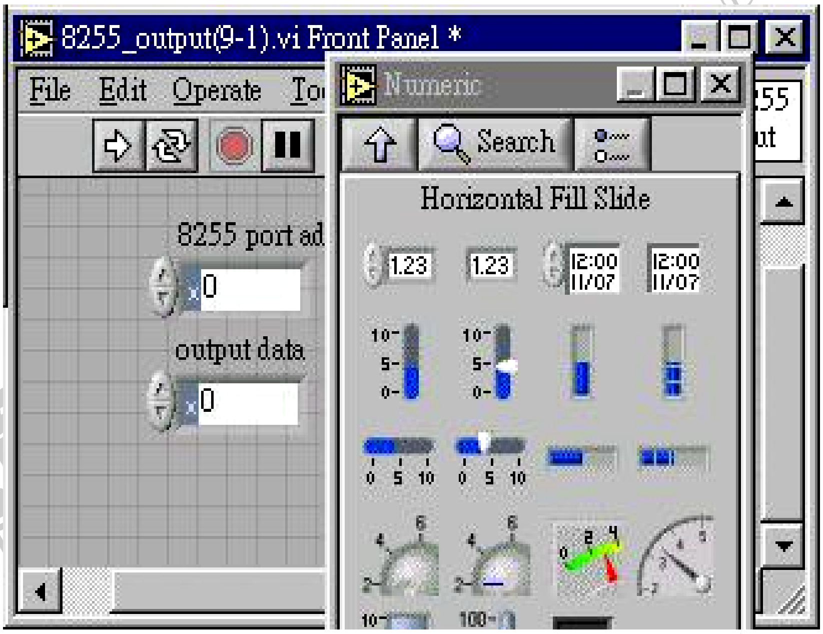
Practice 9-1 8255 output with user defined SubVI

We hope complete a 8255 output's SubVI to reduce the complex editing of the program, and it would look like this program more comfortable, neat and no loading. It would make student understand how to write control word in order to control the 8255 output.

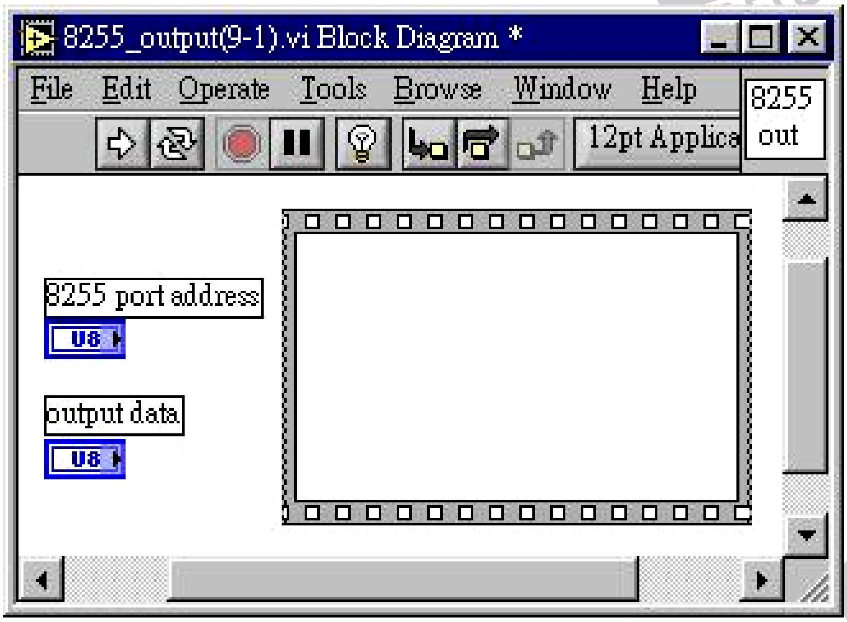
1、How to expand the parallel port with a 8255 circuit is shown as follows.


Parallel Port
P1 P2 P3 P4 P5 P6 P7 P8 P9 P10 P11 P12 P13
P14 P15 P16 P17 P18 P19 P20 P21 P22 P23 P24 P25
74373
D0 D1 D2 D3 D4 D5 D6 D7
Q0 Q1 Q2 Q3 Q4 Q5 Q6 Q7
OE
LE
ALE
8255
D0 D1 D2 D3 D4 D5 D6 D7
RD
WR
A0
A1
RESET
CS
PA0 PA1 PA2 PA3 PA4 PA5 PA6 PA7
PB0 PB1 PB2 PB3 PB4 PB5 PB6 PB7
PC0 PC1 PC2 PC3 PC4 PC5 PC6 PC7

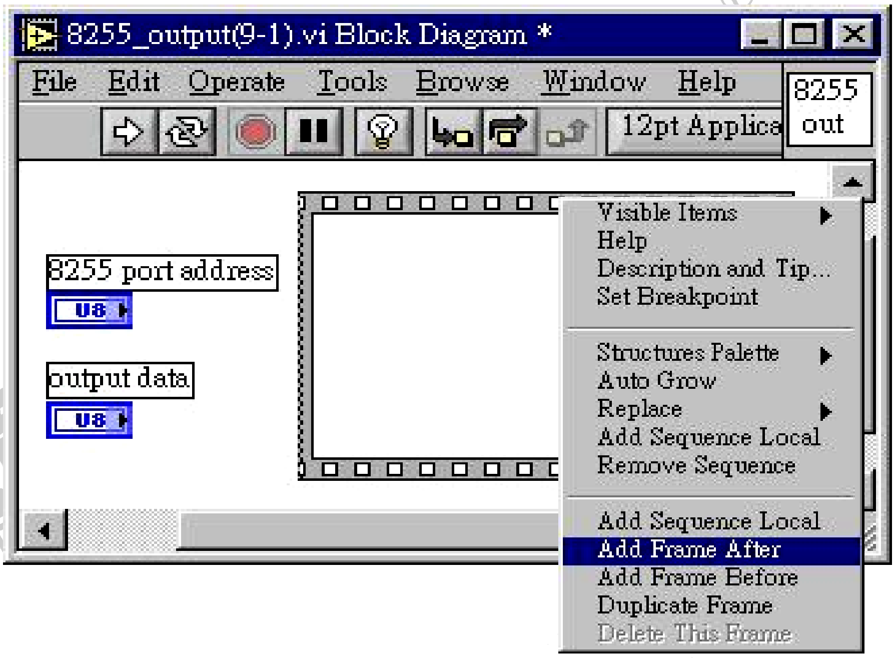
2、To execute the program of LabVIEW, please join two new icons of **Controls➤All Controls➤Numeric➤Numeric Control** into the **Front Panel** of LabView, and the name of icon is “8255 port address” and “output data” respectively. The so-called “8255 port address” includes four address which are 00h；01h；02h；03h. Therefore we can output data through PORT A、PORT B、PORT C or write in the control word to the control word register. Remember to set the type of those two icons as **hexadecimal** and **8 bits unsigned** integer respectively.



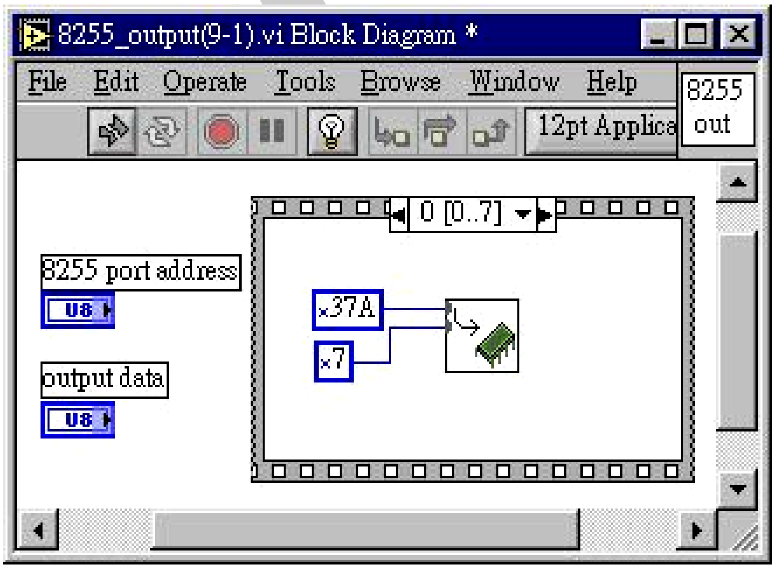
3、At the **Block Diagram** select **Functions➤All Functions➤Structures ➤Stacked Sequence Structured**, and draw a new block.


8255_output(9-1).vi Block Diagram *
File Edit Operate Tools Browse Window Help
12pt Applica
8255 out
8255 port address
U8
output data
U8

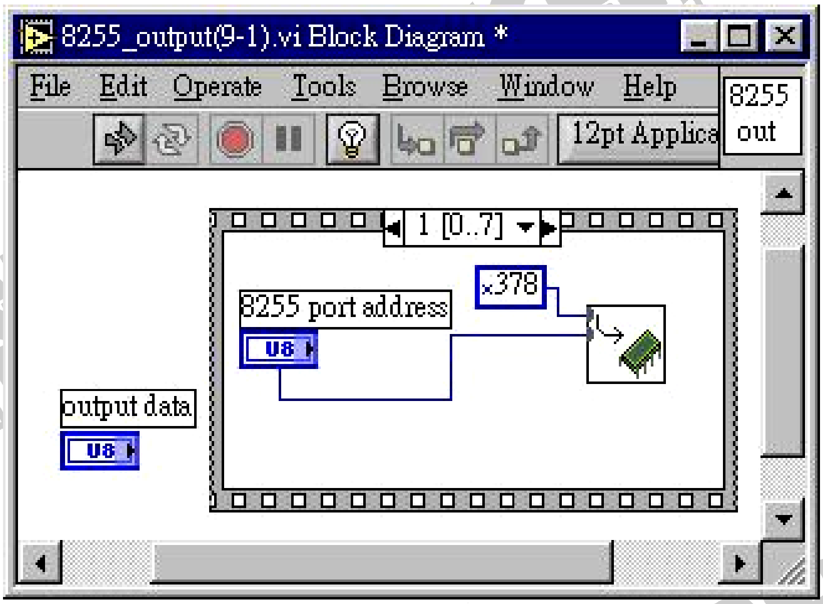
4、Click right hand button of mouse on **Sequence**, repeatedly choose **Add Frame After**, then until add 8 new page frames. (from No.0 to No. 7)



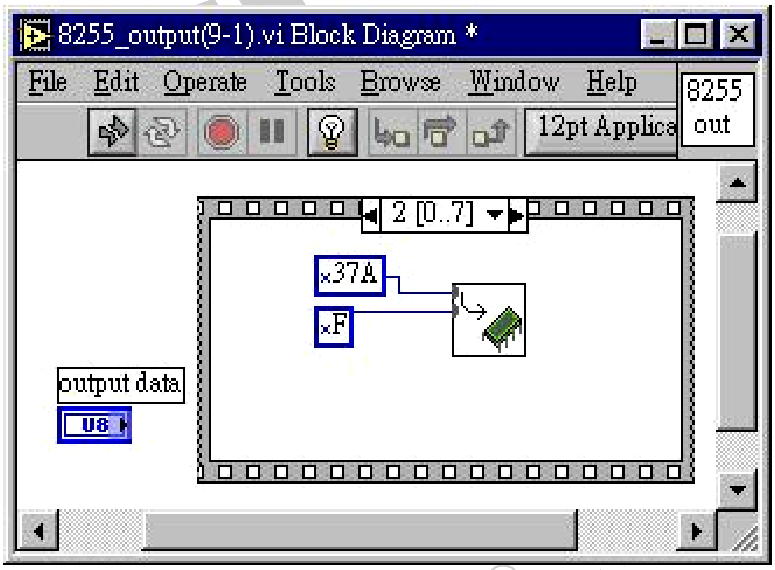
5、At the first page frame please add these steps as follows. Those steps are how to set 8 data bus of parallel port as output and disable ALE, /RD, /WR of 8255. (ALE becomes low level, /RD becomes high level, /WR becomes high level).


8255_output(9-1).vi Block Diagram *
File Edit Operate Tools Browse Window Help
12pt Applica
8255 out
0 [0..7]
8255 port address
U8
output data
U8
x37A
x7

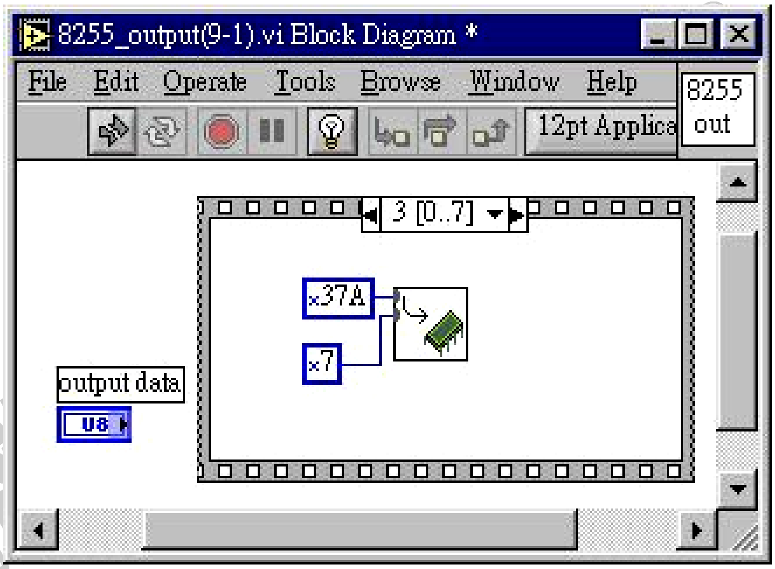
6、At the second page frame please add these steps as follows.　Those steps are how to send the 8255 address, the so-called “**8255 port address**” includes four address which are 00h；01h；02h；03h. they are pointed three ports and one control word register of 8255.


8255_output(9-1).vi Block Diagram *
File Edit Operate Tools Browse Window Help
12pt Applica
8255 out
1 [0..7]
x378
8255 port address
U8
output data
U8

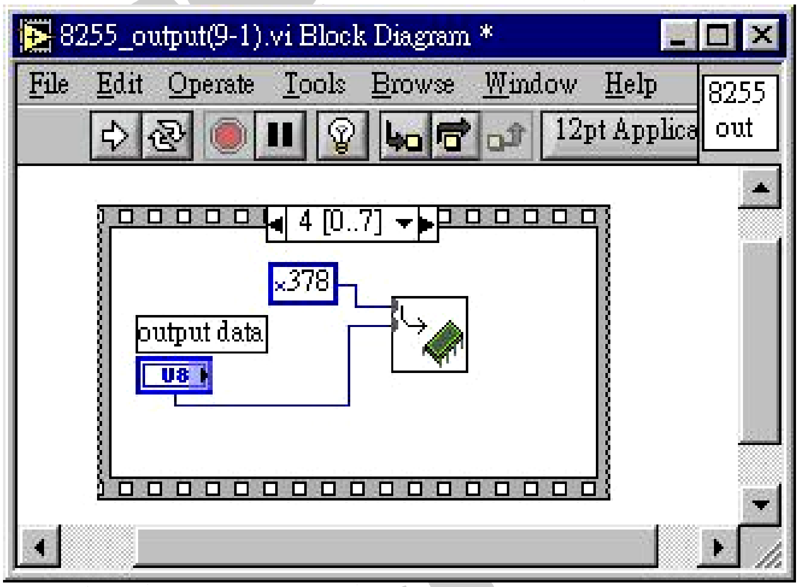
7、At the third page frame please add these steps as follows. Those steps are how to set ALE of 8255 as high level voltage. It means to use 74373 IC to latch the sending address.


8255_output(9-1).vi Block Diagram *
File Edit Operate Tools Browse Window Help
12pt Applica
8255 out
2 [0..7]
x37A
xF
output data
U8

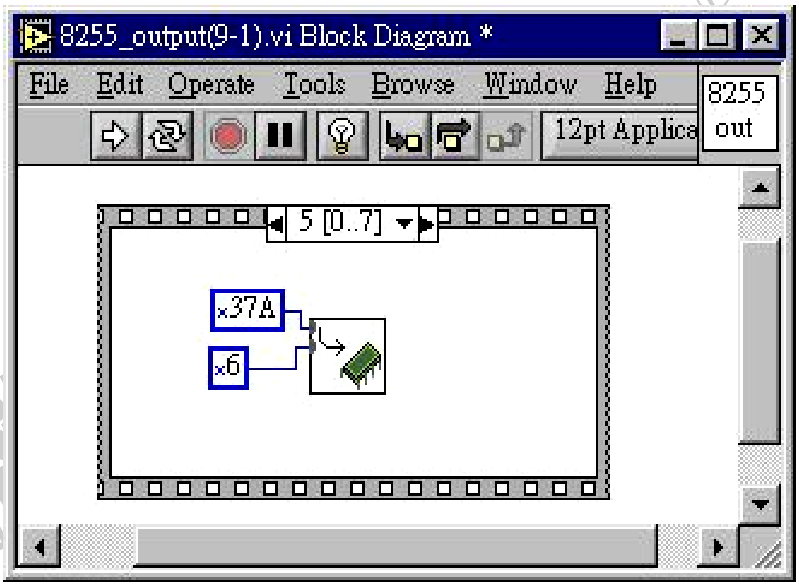
8、At the forth page frame please add this step as follows. This step is how to set ALE of 8255 as low level voltage.


8255_output(9-1).vi Block Diagram *
File Edit Operate Tools Browse Window Help
12pt Applica
8255
out
3 [0..7]
x37A
x7
output data
U8

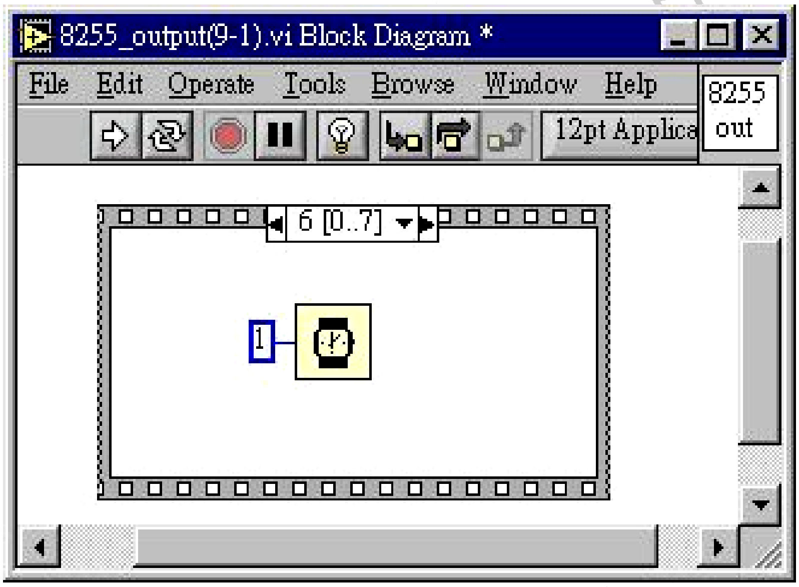
9、At the fifth page frame please add this step as follows. This step is how to send data. For example, if we want sending data to PORT A. Firstly, set the address as 00h, then sending data to PORT A of 8255. Another case if we want setting the control word register of 8255 and three PORTs as output, then the address should be set as 03h and output data should be 80h.


8255_output(9-1).vi Block Diagram *
File Edit Operate Tools Browse Window Help
12pt Applica
8255
out
4 [0..7]
x378
output data
U8

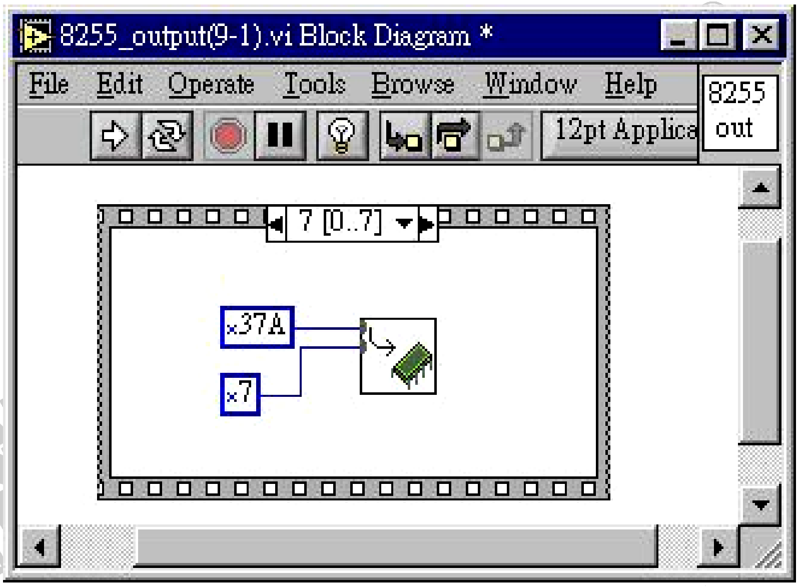
10、At the sixth page frame please add this step as follows. This step is how to set /WR as low level voltage. It means to inform 8255 for data input.


8255_output(9-1).vi Block Diagram *
File Edit Operate Tools Browse Window Help
12pt Applica
8255 out
5 [0..7]
x37A
x6

11、At the seventh page frame please add this step as follows. This step is how to delay 1 msec. It means /WR of 8255 has to be written during the specified time, so the data could be received correctly.

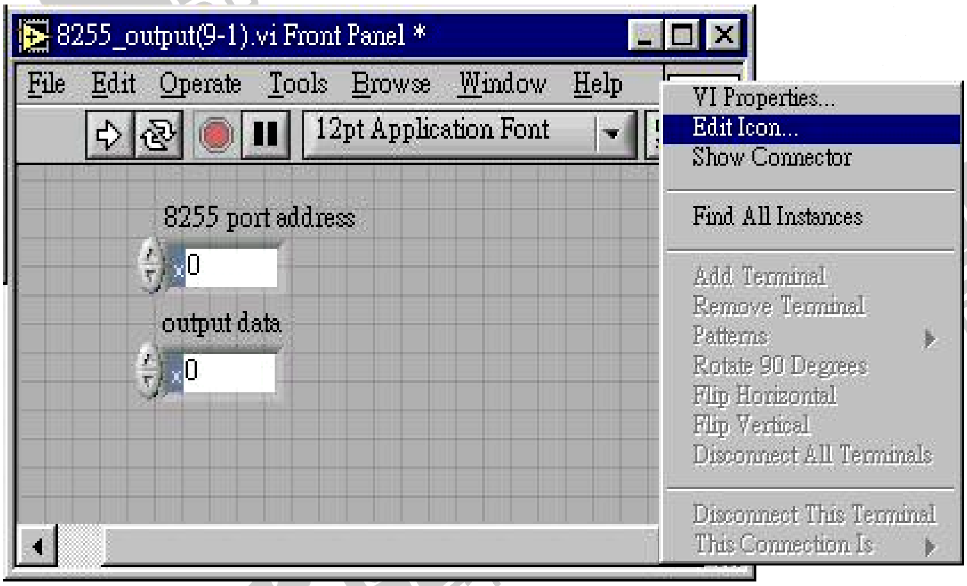

8255_output(9-1).vi Block Diagram *
File Edit Operate Tools Browse Window Help
12pt Applica
8255 out
6 [0..7]
1

12、At the eighth page frame please add this step as follows. This step is how to set /WR as high level voltage. It means to inform 8255 to finish writing action.

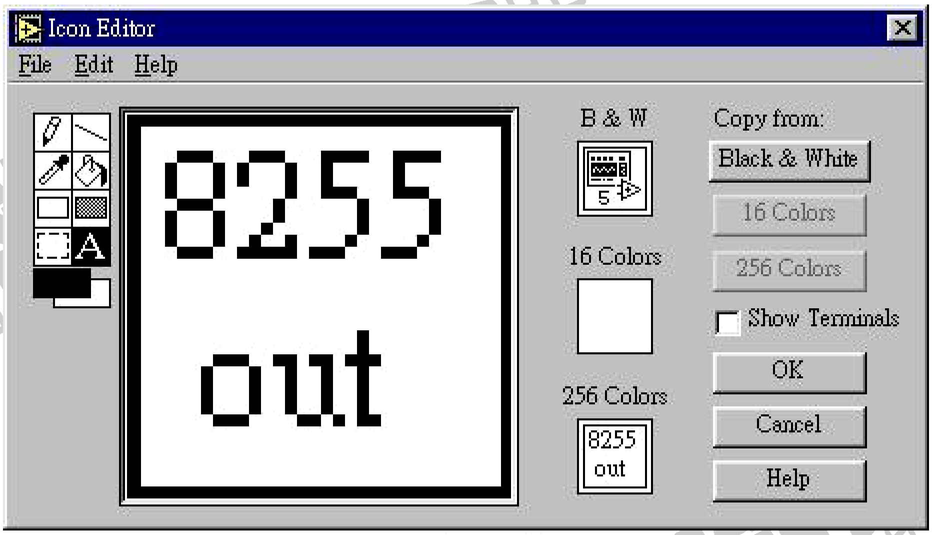

8255_output(9-1).vi Block Diagram *
File Edit Operate Tools Browse Window Help
12pt Applica
8255
out
7 [0..7]
x37A
x7

13、See, it's a tedious work for this program and it takes many memory spaces of PC! We would like to teach you how to make a subprogram instead of the above programs.

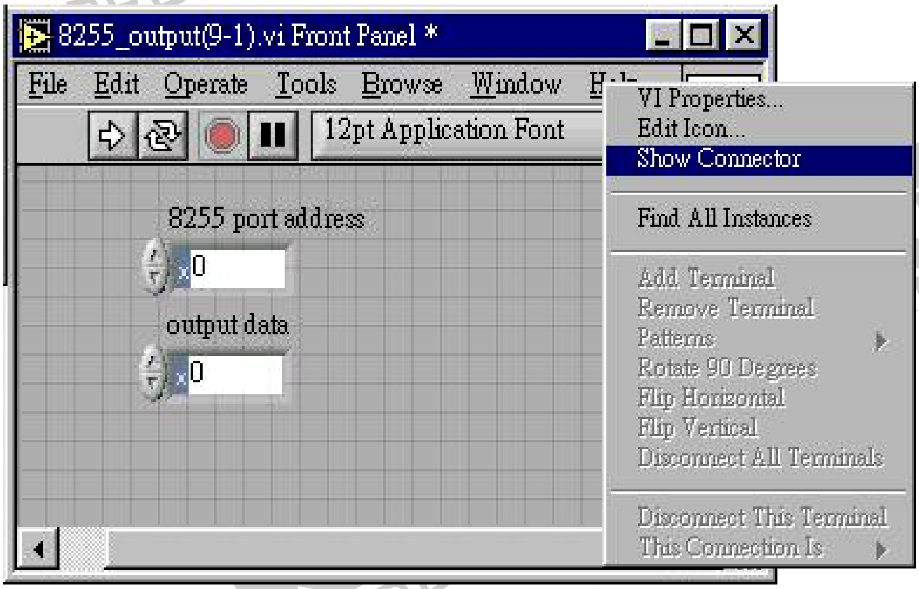
14、In the right hand side up corner of **Front Panel** of LabVIEW, please click the right button of mouse upon the icon. It will show as follows then click **Edit Icon**.

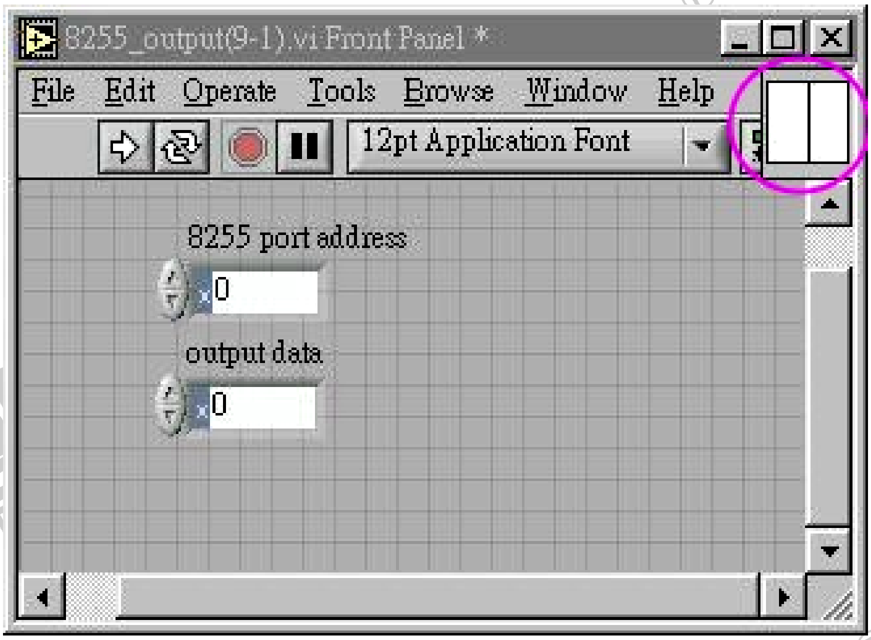


15、Then you can see the window as bellow. Please use the tool on the left hand side to erase original diagram. Write down words as bellow by text tool. Then click **OK**.

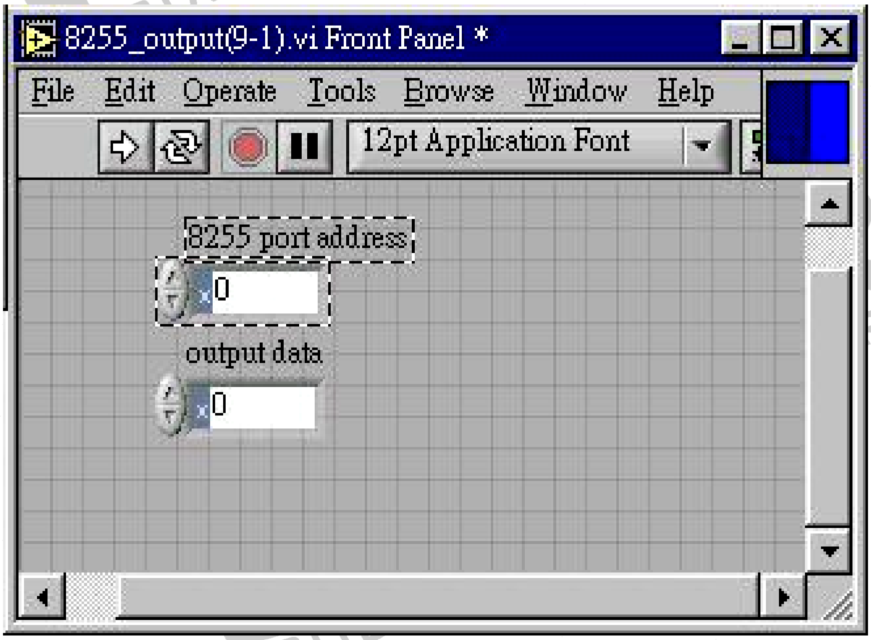


16、You can see the graphic picture on the right upon corner has changed. Now, we need to define which control element should be pulling out from the subprogram. Please click right button on the graphic interface upon the window. Then choose “**Show Connector**”, you can find the graphic picture is dissevered into two parts, just like picture No. 2.

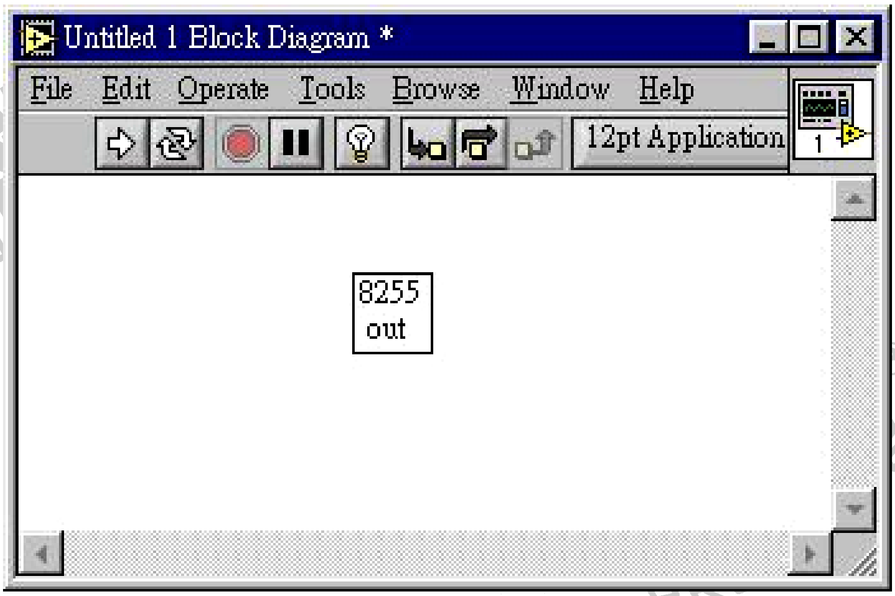




17、Please click **8255 port address**, then click the left hand side of the red circle from the above picture. Click "**output date**", and then click the right hand side of the red circle. You can see the color of graphic picture has changed. It means we define well the pins from the subprogram. Please save it as another new file and name it as "**8255_output.vi**". See, subprogram is completely generated.



18、The last step is how to call this subprogram. It's quite simple, from the **Block Diagram** window select "**Functions➤All Functions➤Select a vi…**" it will appear the path for searching files, and then we can call out "8255_output.vi". It will be shown your subprogram.

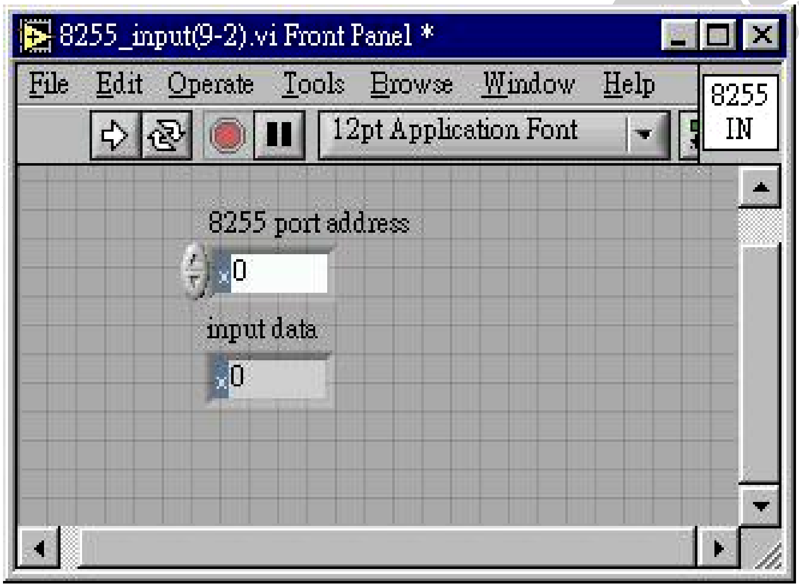


The end of practice 9-1

Practice 9-2 8255 intput with user defined SubVI

We've completed SubVI for 8255 output; at the same time, we need to make another SubVI of 8255 input in order to call function easily。

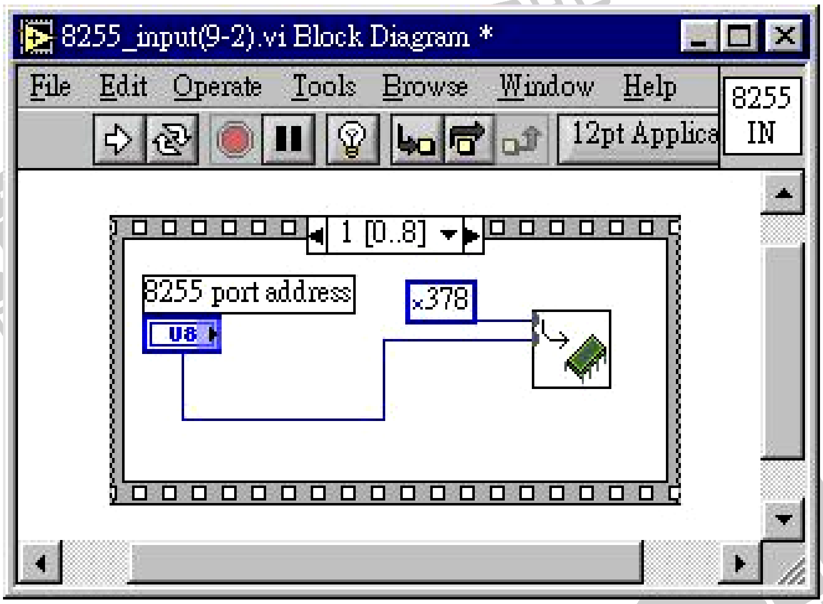
1、Open LabVIEW new program, select on the **Front Panel** "**Controls➤All Controls➤Numeric➤Numeric Control**", named it "**8255 port address**". As usual "8255 address" means 00h, 01h, 02h and 03h. Besides, select " **Controls➤All Controls➤Numeric➤Numeric Indicator**", named it "**input data**". So, we can make data output or input from PORT A , PORT B, PORT C. Please pay attention, firstly set these two control words as 8 bits **Unsigned Integer**, and as **Hexadecimal** format.


8255_input(9-2).vi Front Panel *
File Edit Operate Tools Browse Window Help
12pt Application Font
8255
IN
8255 port address
x0
input data
x0

2、Repeat practice 9-1 the step 3 and step 4 and add nine frames.

3、Add program into page frame 1 as the same action from practice 9-1: step 5.

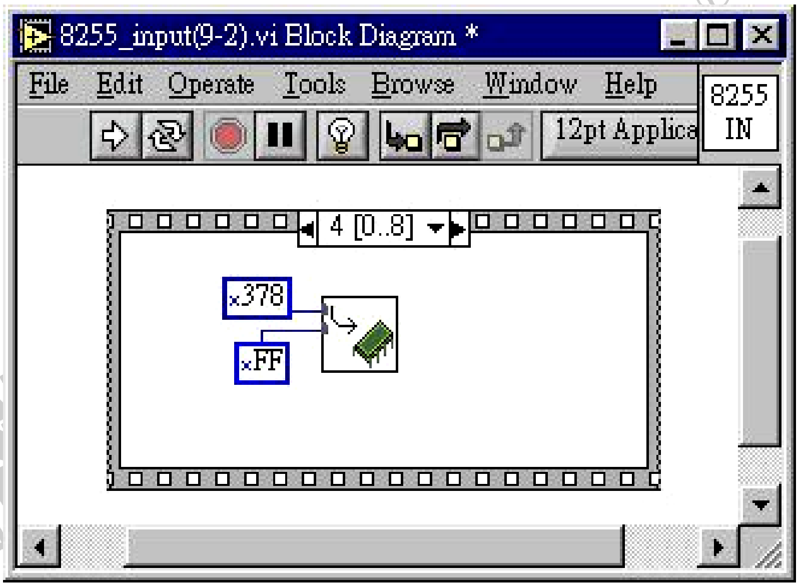
4、At the second page frame please add this step as follows. This is to send 8255 address. It means to send 00h, 01h, 02h, 03h, three PORTs and one control word register.


8255_input(9-2).vi Block Diagram *
File Edit Operate Tools Browse Window Help
12pt Applica
8255
IN
1 [0..8]
8255 port address
U8
x378

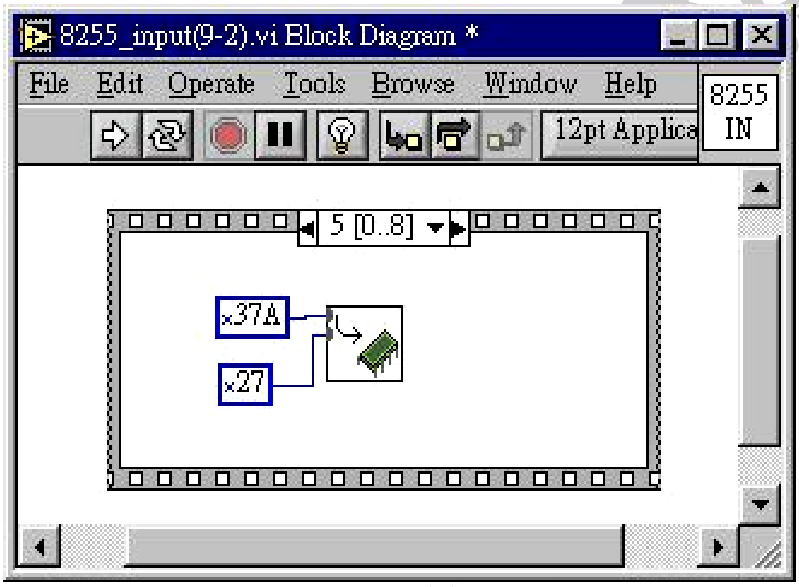
5 、At the third page frame as the same action from practice 9-1: step 7 please add those steps following that. This action is to set ALE of 74373 as high level voltage. It means to use 74373 IC to latch the sending address bus.

6 、At the forth page frame as the same action from practice 9-1: step 8 please add those steps following that. This action is to set ALE of 74373 as low level voltage.

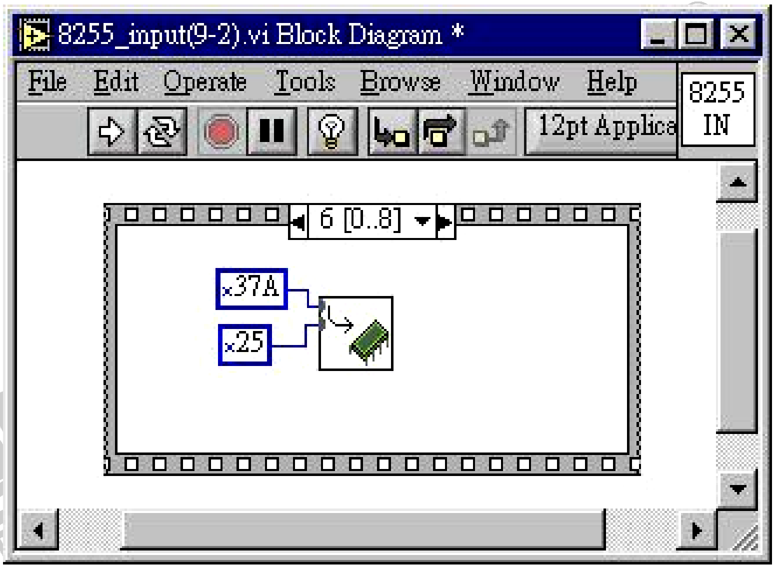
7 、At the fifth page frame please add this step as follows. This step is how to set 8255 PORT as data input.


8255_input(9-2).vi Block Diagram *
File Edit Operate Tools Browse Window Help
12pt Applica
8255
IN
4 [0..8]
x378
xFF

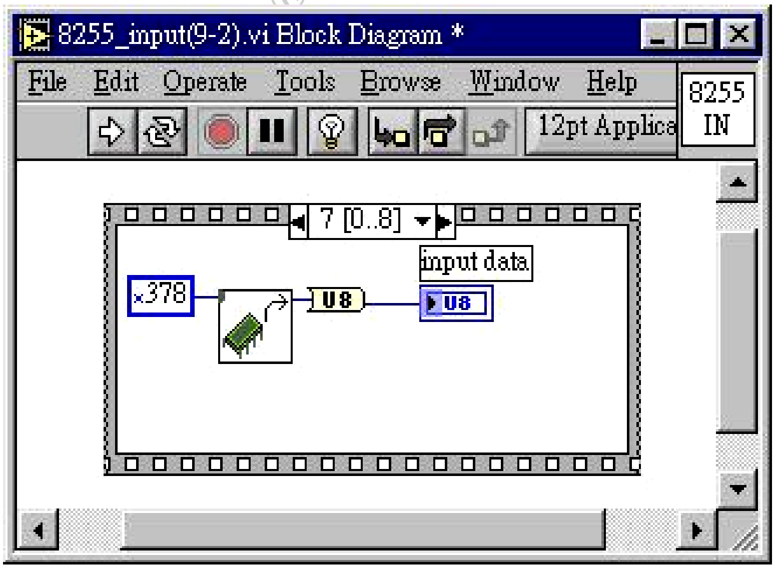
8、At the sixth page frame please add this step as follows. This step is how to set 8 bits data bus of parallel port as data input.


8255_input(9-2).vi Block Diagram *
File Edit Operate Tools Browse Window Help
12pt Applica
8255
IN
5 [0..8]
x37A
x27

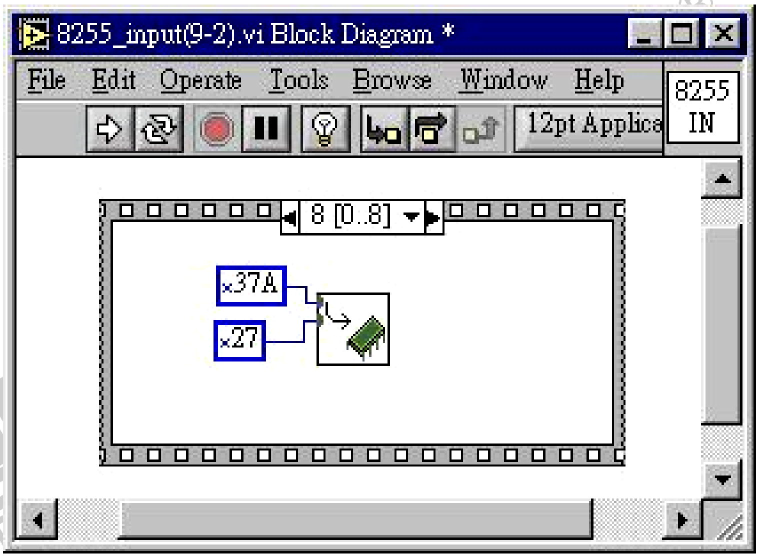
9、At the seventh page frame please add this step as follows. This step is how to send /RD signal as low level voltage and informs 8255 for reading external data.


8255_input(9-2).vi Block Diagram *
File Edit Operate Tools Browse Window Help
12pt Applica
8255
IN
6 [0..8]
x37A
x25

10、At the eighth page frame please add this step as follows. When we send /RD as low level voltage, 8255 will put the data of indicated PORT on the data bus of 8255. So, this program read directly the data from the parallel data bus. (Because of those data bus are connected together).


8255_input(9-2).vi Block Diagram *
File Edit Operate Tools Browse Window Help
12pt Applica
8255
IN
7 [0..8]
x378
U8
input data
U8

11、At the ninth page frame please add this step as follows. This action is to send /RD signal as high level voltage. It means to stop reading external data.


8255_input(9-2).vi Block Diagram *
File Edit Operate Tools Browse Window Help
12pt Applica
8255
IN
8 [0..8]
x37A
x27

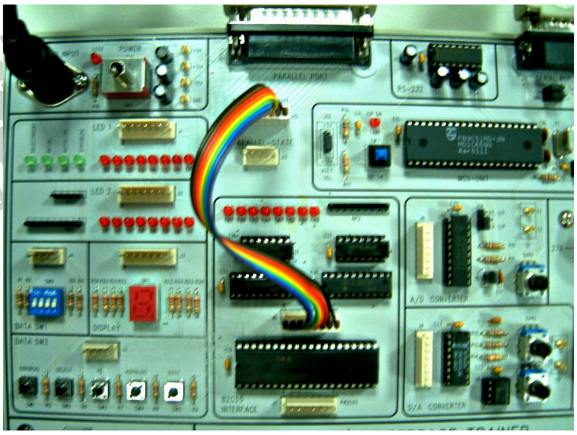
12、Then, repeat practice 9-1: step 13~17,change “8255OUT” into“8255 IN” Beside, save subprogram as “8255_input.vi”, so we can recall it.

13、The final action is to recall this subprogram. In the Block Diagram, select **Functions➤All Functions➤Select a vi…**, then it will appear path for searching file. Just recall “**8255_input.vi**”, you can see your subprogram easily!

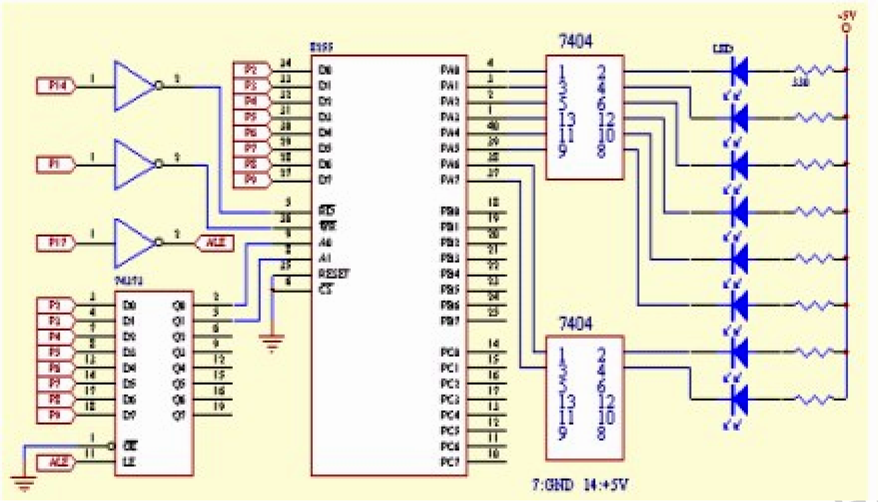
The end of practice 9-2

Practice 9-3 The rolling LED

In the chapter of parallel output, we connect directly the parallel port with 8 LEDs as variation of rolling LED. We use indirectly the parallel I/O extension to add one 8255, then use PORT A of 8255 to be a rolling LED.

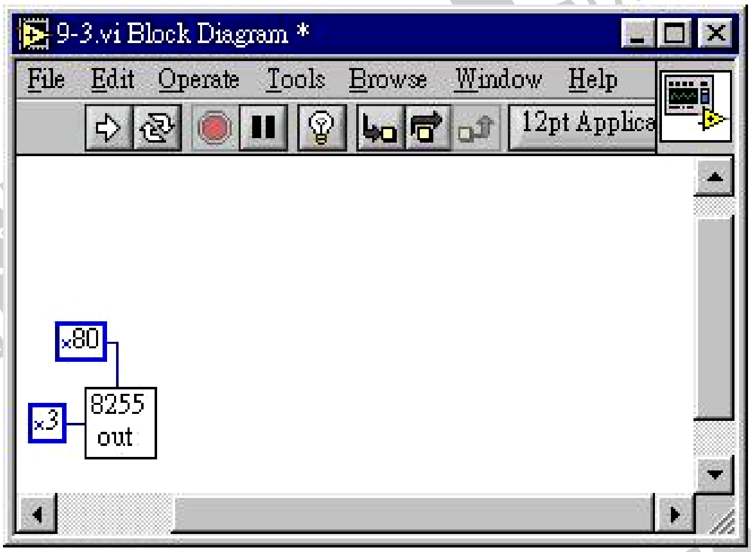


1、The hardware circuit is as bellow. Please pay attention: when LED 1 is on, LED 0 should be off.

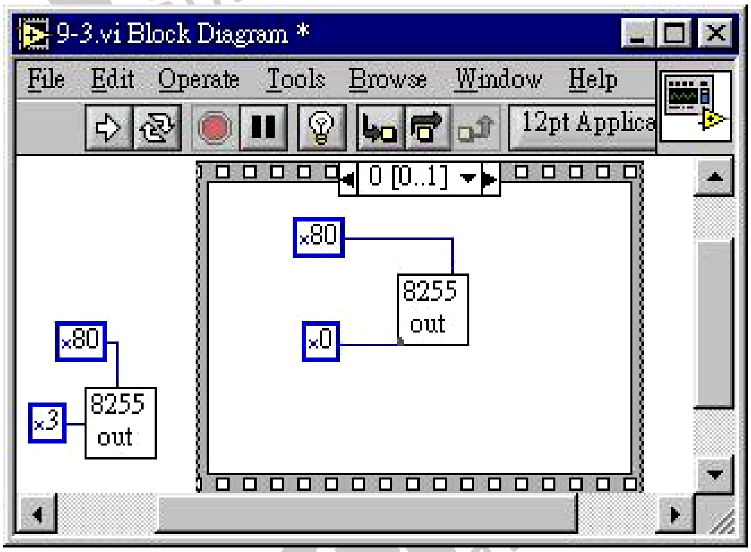


2、Open LabVIEW program, in the **Block Diagram** window select "**Functions➤All Functions➤Select a VI…**", add SubVI what we've made "**8255_output.vi**" and make 8255 address as 03h, output data as 80h.

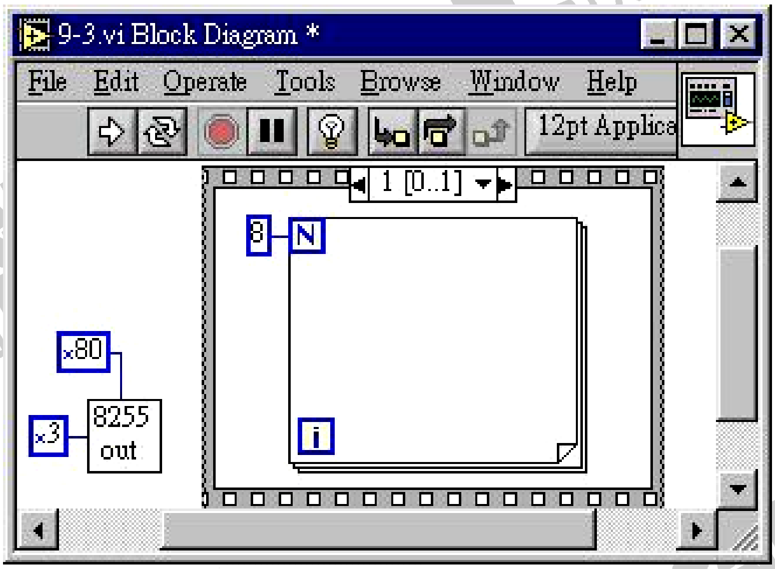
This action is to explain writing 80h into 8255 control word register (03h). It means all PORTs of 8255 are output state. (Please make all numbers type as **8 bits Unsigned Integer** and **hexadecimal**.)



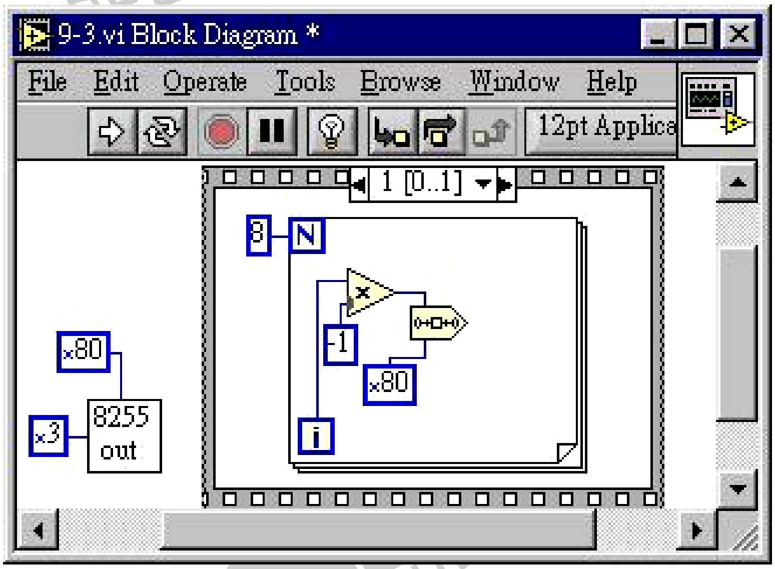
3. 、Select “**Functions➤ All Functions➤Structures➤Stacked Sequence Structured**”. Pull out one block to make it two new frames. Make the first frame to be connected as the following program. It means sending the first value as 80h. High bit LED output is high level voltage (LED on), the others are low level voltage (LED off).



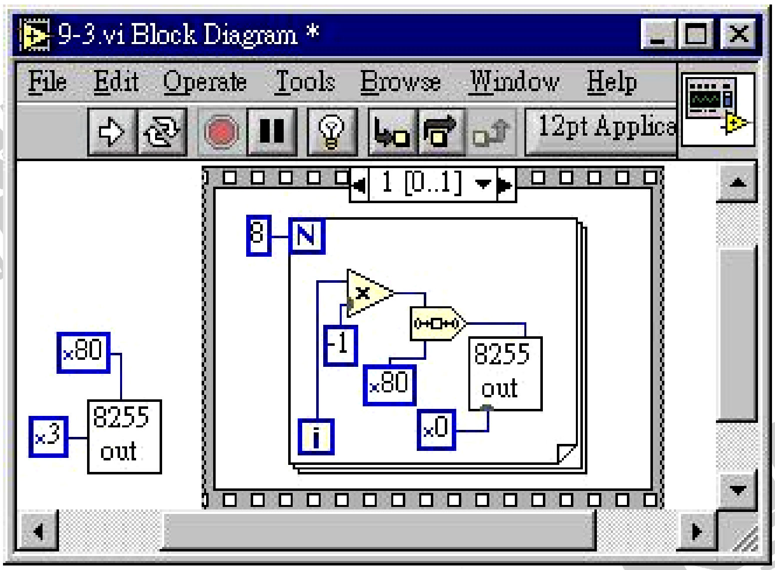
4、For the second frame, select “**Functions➤All Functions➤Structures➤For Loop**” and pullout a block. Set **N** as **8**, that’s because the flash light needs to move/shift 8 times to the right hand side in order to finish all.



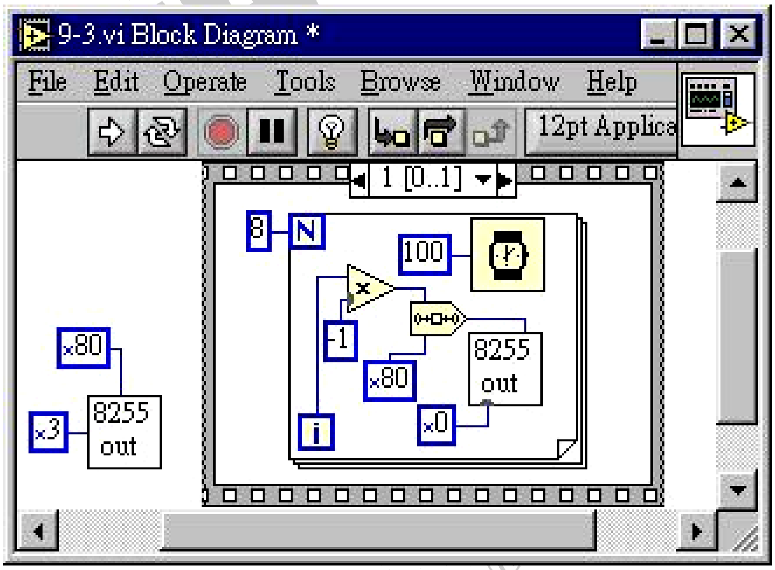
5、Select “**All Functions➤Numeric➤Multiply**” and select “**All Functions➤Advanced➤Data Manipulation➤Logical Shift**", the connecting block is as bellow. Multiply “-1” with multiplier and send to shift register. If the input of shift register is negative, it will shift right; otherwise it will shift left. Each time 80h will follow loop variable I-index to shift.



6、Select “**All Functions➤Select a VI…**”, then add SubVI of “**8255_output.vi**” that we’ve made. Add 00h as 8255 address; connect the output data to output after right shift. This action is to explain how to send data to PORT A (00h) of 8255 after shifting.



7、To the last, choose **All functions➤Time & Dialog➤Wait (ms)**,set 100 mille-second for delay to let LED rolling control doesn’t flash too fast.



Continuous Run Button

8、Press **Continuous Run Button** to watch whether the LED rolling control flashes from higher weight bit to lower weight bit . Try it now!

The end of practice 9-3

Exercise and discussion

1、What is the SubVI？is there any benefits to apply it？

2、How to connect parallel port with three chips of 8255 instead of single 8255, what is the addressing of each PORT for those 8255?

3、What is the reason for using the latch such as 74373 ?

4、Try to change the output from PORT A into PORT B at practice 9-3, do you know how to do ?

5、If we omit the inverter 7404 between 8255 and LEDs, then how to modify the program of LabVIEW in order to maintain the same function of the rolling LEDs.