

The chart below illustrates how to access the various locations and sizes. IN is a way of reading multiple inputs simultaneously.



Sending or receiving multiple bits of data in parallel: (The book seems to use the high byte in all of its examples).

In order to set groups of PINS to be output (1) or input (0) you use the DIRH command

DIRH = %00000000 would set PINS P8-P15 to be inputs

DIRH= %11111111 would set PINS P8-P15 to be outputs

OUTH = %00000000 would set BITS 8-15 to be low

OUTH= %11111111 would set BITS 8-15 to be high

Y = INH would read the high byte into the variable Y

IF YOU WERE USING THE LOW BYTE

DIRL = %00000000 would set PINS P0-P7 to be inputs

DIRL= %11111111 would set PINS P0-P7 to be outputs

OUTL = %00000000 would set BITS 0-7 to be low

OUTL= %11111111 would set BITS 0-7 to be high

Y = INL would read the low byte into the variable Y

Finally, if you wanted to set the high byte to be all outputs and the low byte to be all inputs you could do so with the single command:

DIRS = %1111111100000000

The Hexadecimal number system is based upon the idea that long strings of 1's and 0's are very hard to remember, so long strings are broken down into groups of 4:

0	–	0000	
1	–	0001	
2	–	0010	
3	–	0011	
4	–	0100	
5	–	0101	
6	–	0110	
7	–	0111	
8	–	1000	
9	–	1001	Decimal Equivalent
A	–	1010	10
B	–	1011	11
C	–	1100	12
D	–	1101	13
E	–	1110	14
F	–	1111	15

Therefore, you can use the hex numbering system in place of binary. However, you need to tell the compiler you are using Hex by using the prefix \$ in place of %, which is the prefix for binary.

For example:

\$E would be the same as %1110

\$EF would be the same as %11101111

\$FF would be the same as %11111111

Questions:

%11001101 would be what Hex number? What Decimal number?

\$A3 would be what binary number? What decimal number?