

Integer Precision Example

```

//LongInt.c
// This program inputs 4 bytes from P0, adds them up, and finds
// the average.  The average is converted to BCD and displayed
// on P2 and P1.  Long ints are used to preserve precision
// so that the display always gives 3 digits for the average
// in the form of ddd., dd.d, or d.dd
#include<reg51.h>
void Convert2BCD(long avg);
void main(void)
{
    unsigned char w, x, y, z;
    long sum, avg;
    w = P0;           //Input 4 bytes as unsigned chars
    x = P0;
    y = P0;
    z = P0;
    sum = w + x + y + z; //Add them up in a long;
    avg = 100*sum/4;     //*100 and divide by 4 for avg
                        // this preserves precision
    Convert2BCD(avg);   //Convert to BCD and display
}
//
void Convert2BCD(long avg)
{
    unsigned char d0, d1, d2, d3, d4;
    d0 = avg % 10;     //Convert to BCD by using mod function
    avg = avg/10;      // Largest avg = 255*100 = 25500 so
    d1 = avg % 10;     // five digits
    avg = avg/10;
    d2 = avg % 10;
    avg = avg/10;
    d3 = avg % 10;
    avg = avg/10;
    d4 = avg %10;
    if(d4 != 0)        //If d4 is not 0 display d4,d3,d2
    {
        P2 = d4;      // decimal at the end
        P1 = d2 + d3*16;
    }
    else if(d3 != 0) //If d4 = 0 but d3 != 0 then display
    {
        P2 = d3;      // d3 d2.d1
        P1 = d1 + d2*16;
    }
    else                //if d4 = 0 and d3 = 0 then display
    {
        P2 = d2;      // d3.d2,d1
        P1 = d0 + d1*16;
    }
}
}

```