```
// Bitmuster der Ausgaenge und Eingaenge darstellen

void disp_ios(void)       //      Darstellmodul der I/Os            ANFANG
{
  idata unsigned char     i,j,x1;
  for(i=0;i<max_port;i++)
         {
         if(def_feld[i]==1)      // Ausgabebaustein
           {
            j=0;
             for(x1=0x80;x1!=0x08;x1=x1>>1)
                   { if(ein_aus[i]&x1)   { text[j]='1'; }
                      else                { text[j]='0'; }
                     j++;
       }
            text[j]='|'; j++;
            for(;x1>0x00;x1=x1>>1)
                  { if(ein_aus[i]&x1)   { text[j]='1'; }
                    else                  { text[j]='0'; }
                    j++;
         }
           printf("%c%c%c%c%c%c%c%c ",text[0],text[1],text[2],text[3],
                                      text[4],text[5],text[6],text[7],
                                      text[8]);
           }


    else if(def_feld[i]==0)// Eingabebaustein
           {
            j=0;
             for(x1=0x80;x1!=0x08;x1=x1>>1)
                   { if(ein_aus[i]&x1)   { text[j]='H'; }
                     else                  { text[j]='L'; }
                     j++;
       }
            text[j]='-'; j++;
            for(;x1>0x00;x1=x1>>1)
                  { if(ein_aus[i]&x1)   { text[j]='H'; }
                    else                  { text[j]='L'; }
                    j++;  }
           printf("%c%c%c%c%c%c%c%c ",text[0],text[1],text[2],text[3],
                                      text[4],text[5],text[6],text[7],
                                      text[8]);
           }

         else                  // Nicht benuetzt
           {
                 printf("----|---- ");
           }
       }
  printf("\n");
}       //       Darstellmodul der I/Os            ENDE

v24_ausw()      //  V24_Auswertung       ANFANG
{
  idata unsigned char     i, x1;
       temp[0]=*aus_z;
       diff--;
       temp[0]&=0x31;
       if(++aus_z >= &feld[max_s_len]) { aus_z=&feld[0]; }
       temp[1]=*aus_z;
       diff--;
       if(++aus_z >= &feld[max_s_len]) { aus_z=&feld[0]; }
       temp[2]=*aus_z;
       printf("\tAusgang %c%c%c ",temp[0],temp[1],temp[2]);
       diff--;
       if(++aus_z >= &feld[max_s_len]) { aus_z=&feld[0]; }
       temp[3]=*aus_z;
       if(*aus_z=='+')
       { printf(" ON\n");}
       else
       { printf(" OFF\n");}
       diff--;
       if(++aus_z >= &feld[max_s_len]) { aus_z=&feld[0]; }
  temp[0]=(temp[0]&0x01)*100;
  temp[0]=(temp[1]&0x0f)*10;
  temp[0]+=temp[2]&0x0f;
  temp[2]=temp[0]/8;            // Baustein bestimmen
  x1=temp[0]%8;                 // Bit-Position
  if(temp[2]<24)                // Baustein-Max begrenzen
  {
  if(def_feld[temp[2]]==1)      // Dann ist es eine Ausgabe
         {
    i=0x01; i=i<<x1;            // Bitposition einstellen
    if(temp[3]=='+')            // Ausgang ON
          {
          ein_aus[temp[2]] | =i;
          }
    else                          // Ausgang OFF, es ist eine Eingabe
          {
          i^=0xff;
          ein_aus[temp[2]] & =i;
          }
      }
  else
         { printf("\t\t\t\t U N G U E L T I G\n"); }
    printf("\n");
       }
  disp_ios();
  printf("Ihre Eingabe:  ");
}       //      V24_Auswertung          ENDE


void main(void)
{
  init_mpio();
  init_int0();
  b9600();
  RI0=0;
  init_sp();
  init_t();
  zaehler=0;
  printf("%c\n\t\t\tI/O-Karte mit 192 Ein- Ausgaengen",temp[0]);
  printf("\n\t\t\tgesteuert von MP-Karte 'TEST517'\n");
  printf("\n\t\tAusgang  ON   mit Ausgangnummer 0-191 und + , z.B.: 001+\n");
  printf("\t\tAusgang  OFF  mit Ausgangnummer 0-191 und - , z.B.: 001-\n\n");
  printf("Darstellung der Eingaenge mit H/L, der Ausgaenge mit 1/0\n\n");
  printf("Ihre Eingabe:  ");

  while(1) {
      P52=0;
      if(ungleich)
             {
              ungleich=0;
              printf("\n\n");
              disp_ios();
             }
      if(diff>=4)           // xxx = Ausgang   y=+  == ON  y!=+  == OFF
             {
             v24_ausw();
       }
      }
}
□
```

## *Real programmers don't.....*

- *Real programmers don't write specs. Users should consider themselves lucky to get any programs at all and take what they get.*
- *Real programmers don't comment their code. If it was hard to write, it should be hard to read.*
- *Real programmers don't write application programs, they program right down on the bare metal. Application programming is for feebs who can't do systems programming.*
- *Real programmers don't eat quiche. Real programmers don't even know how to spell quiche. They eat Twinkies, Coke and palate-scorching Szechwan food.*
- *Real programmers don't draw flowcharts. Flowcharts are, after all, the illiterate's form of documentation. Cavemen drew flowcharts; look how much it did for them.*
- *Real programmers don't read manuals. Reliance on a reference is a hallmark of the novice and the coward.*
- *Real programmers programs never work right the first time. But if you throw them on the machine they can be patched into working in only a few 30-hours debugging sessions.*
- *Real programmers don't use Fortran. Fortran is for wimpy engineers who wear white socks, pipe stress freaks, and crystallography weenies. They get excited over finite state analysis and nuclear reactor simulation.*
- *Real programmers don't use COBOL. COBOL is for wimpy application programmers.*
- *Real programmers never work 9 to 5. If any real programmers are around at 9 am, it's because they were up all night.*
- *Real programmers don't write in BASIC. Actually, no programmers write in BASIC, after the age of 12.*
- *Real programmers don't document. Documentation is for simps who can't read the listings or the object deck.*
- *Real programmers don't write in Pascal, or Bliss, or Ada, or any of those pinko computer science languages. Strong typing is for people with weak memories.*
- *Real programmers know better than the users what they need.*
- *Real programmers think structured programming is a communist plot.*
- *Real programmers don't use schedules. Schedules are for manager's toadies. Real programmers like to keep their manager in suspense.*
- *Real programmers think better when playing adventure.*
- *Real programmers don't use PL/I. PL/I is for insecure momma's boys who can't choose between COBOL and Fortran.*
- *Real programmers don't use APL, unless the whole program can be written on one line.*
- *Real programmers don't use LISP. Only effeminate programmers use more parentheses than actual code.*
- *Real programmers disdain structured programming. Structured programming is for compulsive, prematurely toilet-trained neurotics who wear neckties and carefully line up sharpened pencils on an otherwise uncluttered desk.*
- *Real programmers don't like the team programming concept. Unless, of course, they are the Chief Programmer.*
- *Real programmers have no use for managers. Managers are a necessary evil. Managers are for dealing with personnel bozos, bean counters, senior planners and other mental defectives.*
- *Real programmers scorn floating point arithmetic. The decimal point was invented for pansy bedwetters who are unable to "think big."*
- *Real programmers don't drive clapped-out Mavericks. They prefer BMWs, Lincolns or pick-up trucks with floor shifts. Fast motorcycles are highly regarded.*
- *Real programmers don't believe in schedules. Planners make up schedules. Managers "firm up" schedules. Frightened coders strive to meet schedules. Real programmers ignore schedules.*
- *Real programmers like vending machine popcorn. Coders pop it in the microwave oven. Real programmers use the heat given off by the cpu. They can tell what job is running just by listening to the rate of popping.*
- *Real programmers know every nuance of every instruction and use them all in every real program. Puppy architects won't allow execute instructions to address another execute as the target instruction. Real programmers despise such petty restrictions.*
- *Real programmers don't bring brown bag lunches to work. If the vending machine sells it, they eat it. If the vending machine doesn't sell it, they don't eat it. Vending machines don't sell quiche.*
- *Real programmers know that the word is disk, not disc. Disc is a definite commie plot put forth by blubbering quiche eaters.*