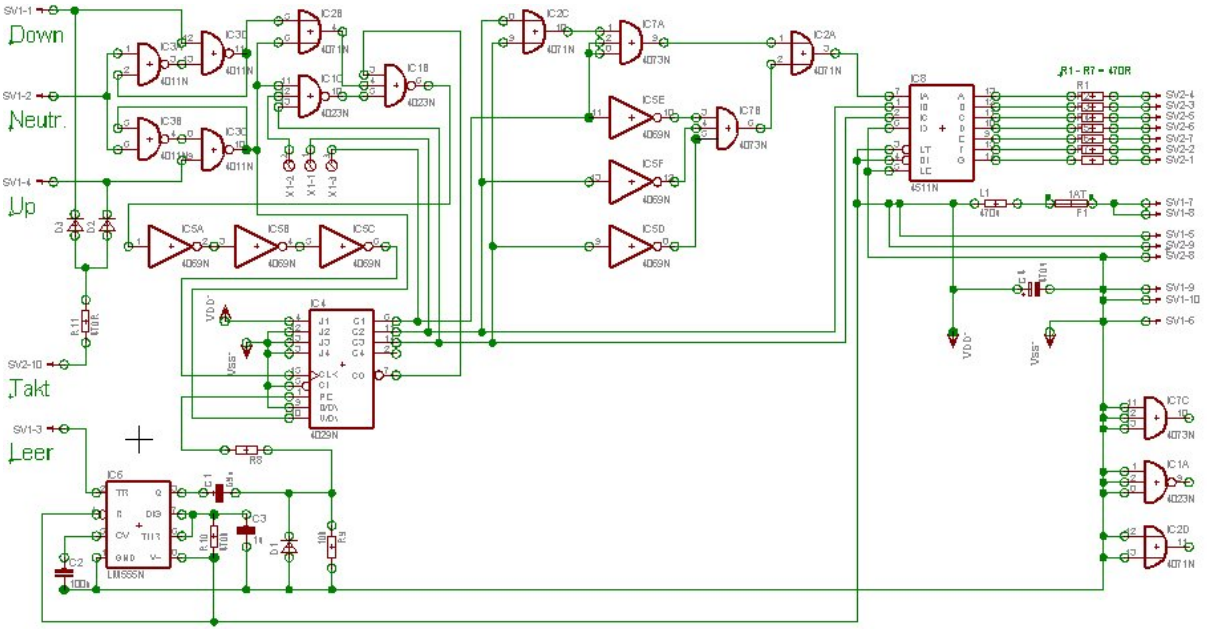
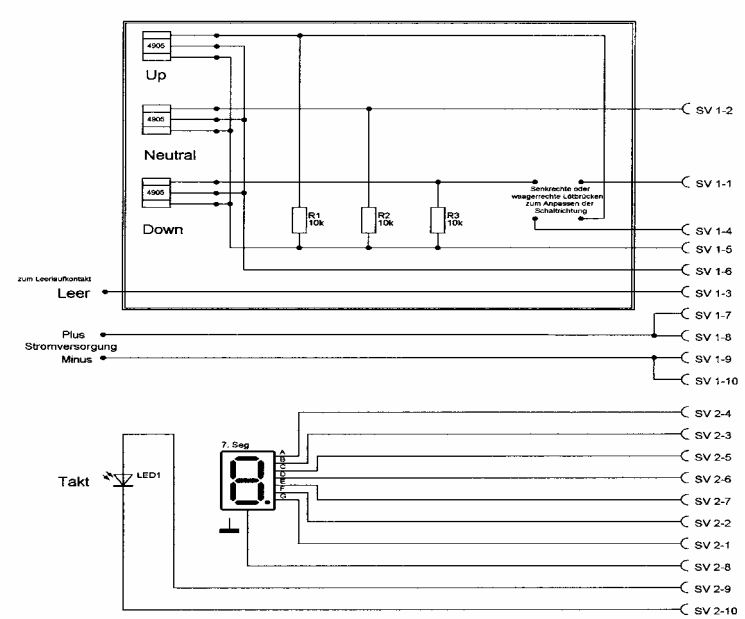
Ausgehend von Deinem Vorschlag auf der Homepage www.alpentourer.de habe ich mir einige Gedanken gemacht wie man die Ganganzeige besser aufbauen kann.

1. Die alte Schaltung mit C- Mos



Ich habe sie damals erst mal in C-Mos aufgebaut und damit kann man die Funktion ganz gut erklären.
An die Anschlüsse Down, Neutral und Up werden die Geber angeschlossen, die durch den Ganghebel betätigt werden. Wenn man z. B. einen Gang hochschaltet könnte es ja durch Mehrfachkontakt sein dass die Schaltung mehrere Gänge hochzählt. Ein weiteres Hochzählen ist aber nur möglich wenn dazwischen der Schalthebel den Neutralgeber betätigt hat. Die Signale werden über die Gatter von IC 1, IC 2 und IC 3 zum Vorwärts- Rückwärtszähler IC 4 geführt. Am Anschluss 10 wird dieses IC von Vorwärts- auf Rückwärtszählen geschaltet. Das Taktsignal wird durch IC 5 A- C verzögert damit der Zähler rechtzeitig von Vor- auf Rückwärtszählen geschaltet hat, bevor das Taktsignal kommt. Das Timer IC 6 resettet und blockiert den Zähler kurzzeitig wenn die Schaltung über den Leerlauf geht, damit in dieser Zeit keine Zählimpulse wirksam werden können. Dadurch wird die Schaltung im Leerlauf immer wieder automatisch synchronisiert. Die Codierung und Aufbereitung des Zählerausgangs auf 7- Segmentanzeige ist ja schon in Deiner Schaltung beschrieben. Ich setzte da allerdings ein BCD zu 7 Segment- Decoder- IC 8 ein damit ich an die Schaltung direkt eine LED Anzeige mit gemeinsamer Kathode anschließen kann.
Die Leitung Takt gibt ein Signal an eine LED damit man erkennen kann ob einer der Geber Signal gibt. Das ist hilfreich beim Einjustieren der Geber am Ganghebel des Motorrades. Den Anschluss „ Leerlauf“ muss ich ja wohl nicht erklären.
Mit den Kontakten X1 wird von 5- auf 6-Gang Getriebe umgeschaltet. Und zwar bei Brücke von X1-1 nach X 1-2 auf 5- Gang und X1-2 nach X1-3 auf 6- Gang.



Die Schaltung des Gebers und der Anzeige

Die Geber sind Hallgeber d. h. sie reagieren auf einen Magneten und schalten unter Einfluss des Magnetfeldes den Ausgang auf Masse. Die beiden anderen Leitungen sind an 12-Volt und Masse angeschlossen. Die Lötbrücken können entweder senkrecht oder waagerecht eingebaut werden und passen die Geberschaltung an die Mechanik des Schaltgestänges an.

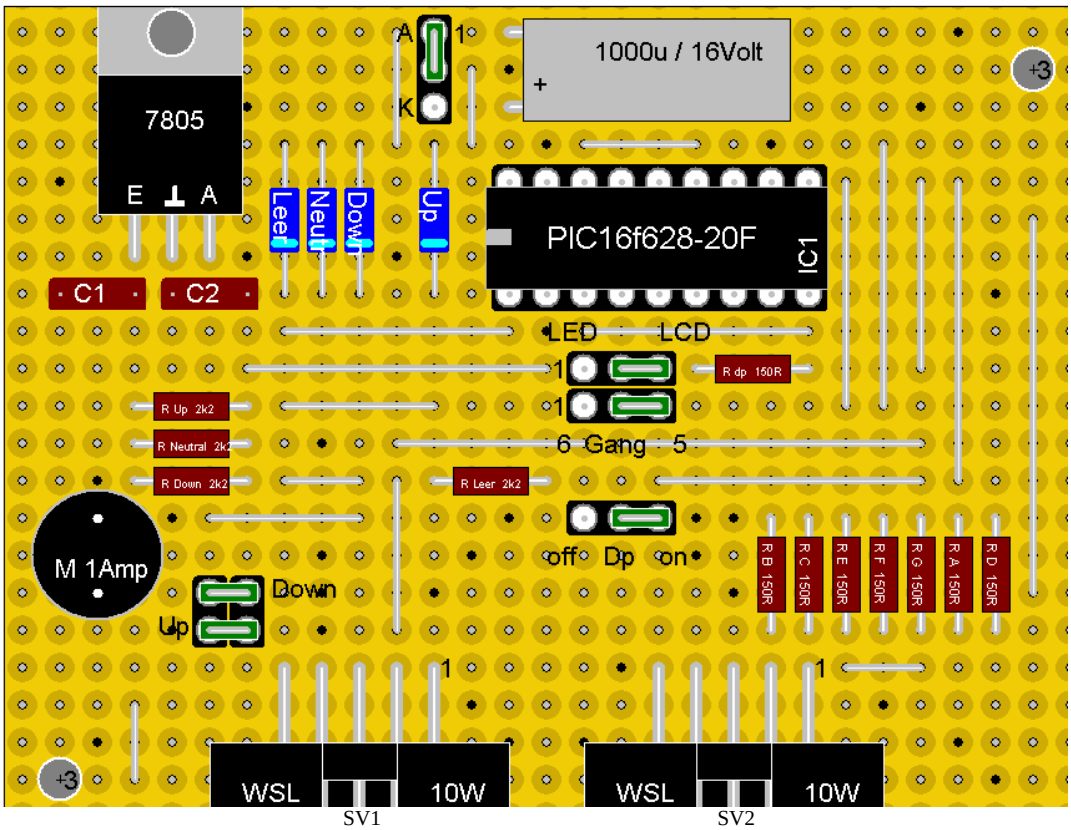
2. Die neue Schaltung mit PIC

Es zeigte sich allerdings dass eine LED-Anzeige bei Dunkelheit zwar gut zu lesen ist, bei Sonnenschein aber überhaupt nicht. Da ich mit meiner Diva aber fast nie bei Dunkelheit fahre sollte also eine LCD Anzeige her. Da diese aber einen höheren Aufwand in der Ansteuerung benötigt haben mein Sohn und ich begonnen die Schaltung in einen PIC zu integrieren um die Menge der ICs zu begrenzen. Er hat die Software geschrieben und ich die Hardware gebaut.
Diese Schaltung ist nun fertig und wir haben noch einige zusätzliche Funktionen, die über Steckbrücken wählbar sind, integriert.
Dies sind:

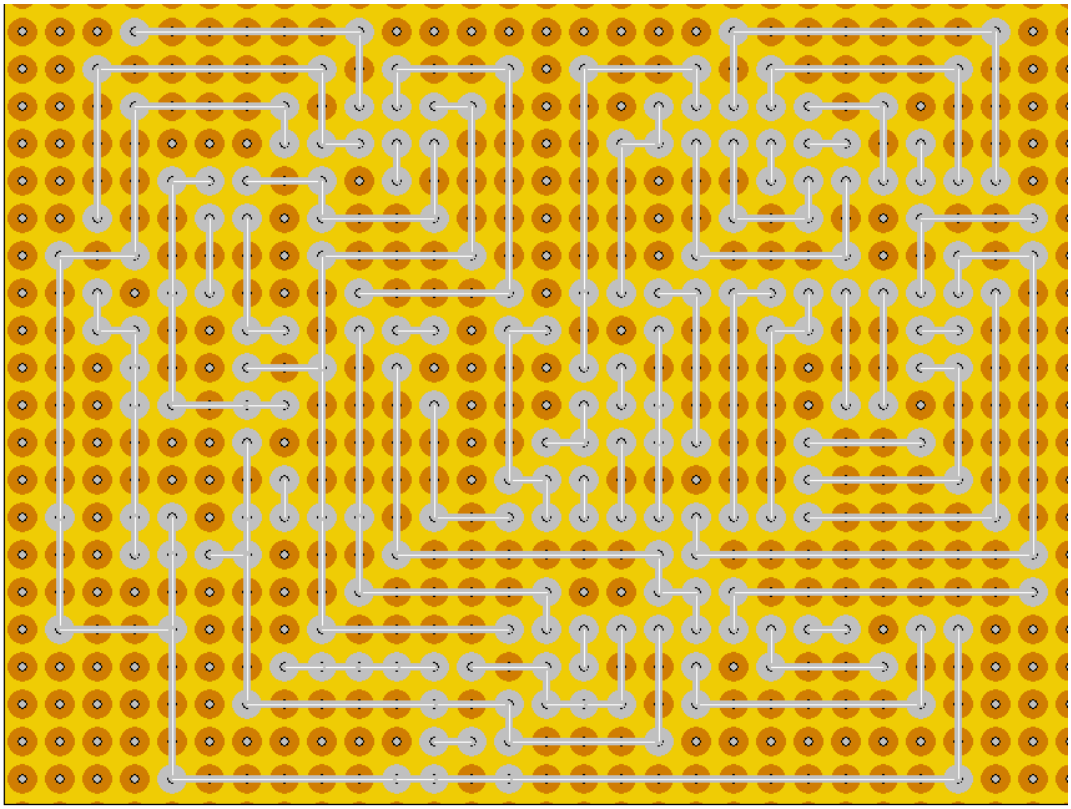
1. wahlweise LED mit gemeinsamer Anode oder Kathode (Steckbrücke A–K)
2. wahlweise LED oder LCD Anzeige (Steckbrücke LED-LCD)
3. 5- oder 6-Gang Getriebe (Steckbrücke 5-6 Gang)
4. abschaltbare Anzeige der Kontaktgabe der Geber (Steckbrücke Dp. On- off)
5. die Lötbrücken der Geberplatine sind als Steckbrücken auf der Hauptplatine ausgeführt. (Steckbrücke Up- Down). Die Brücken können hier waagerecht oder senkrecht gesteckt werden.

Die linke Steckerleiste ist die Leiste SV1 und die rechte SV2 analog zum Schaltplan weiter oben.
Die Schaltung passt genau in ein Gehäuse welches z. B. von der Fa. www.reichelt.de unter der Bezeichnung **SP 2029 GR** angeboten wird.

Die Geberschaltung kann fast unverändert übernommen werden. Nur die Lötbrücken sind ja jetzt überflüssig und können direkt waagerecht verbunden werden und die Widerstände sind überflüssig. Es empfiehlt sich diese Schaltung wasserdicht zu vergießen.



Die PIC- Schaltung mit Lötbrücken auf Lochrasterplatte



Die Lötseite mit Lötbrücken der Lochrasterplatte.

Man sieht hier dass sich der Bauteileaufwand drastisch verringert hat obwohl viel mehr Funktionen integriert sind.

Eine LCD- Anzeige muss mit wechselnder Polarität angesteuert werden. Dieses wird auch im PIC generiert. Ein Segment wird sichtbar wenn es Spannung gegenüber dem Common-Anschluss hat. Bei gleicher Spannung ist es nicht zu sehen. Wenn der PIC auf LCD eingestellt ist kann die LCD- Anzeige direkt an die Ausgänge der Steckleiste SV2 angeschlossen werden. Wird der Dezimalpunkt mit an SV1-9 angeschlossen zeigt er wenn er mit dem Schalter Dp aktiviert ist die Betätigung eines der 3 Geber an. Wie schon oben beschrieben, sehr hilfreich beim Justieren der Geber.
Das gleiche gilt analog auch bei LED- Anzeigen mit gemeinsamer Anode oder Kathode.

Und hier ist das Programm v.1.2.301103 zur Programmierung des PIC
(Ist auch einzeln als asm Datei erhältlich)

```
        list p=16f628                   ;der Prozessortyp wird festgelegt
        #include "p16f628.inc"          ;die include-Datei mit vielen Festlegungen wird
geladen
        errorlevel -302                         ;Bank-Fehler-Meldungen ausschalten

;######Gangschaltungs Anzeige für Motorräder mit 5 oder 6 Gang Getriebe.######
;#                                                                            #
;#                          +---* *---+                                       #
;#                     Up |01 +-+ 18|        Neutral                          #
;#       Leerlauf     |02   17|       Down                                    #
        AAAA                                                                  #
;#        /5-6 Gang   |03   16|       DP                                      F
        B                                                                     #
;#        /LCD LED    |04   15|       Gemeinsame /Anode Kathode          F    B
                                                                              #
;#                Vss  |05   14|       Vdd                                     #
              GGGG                                                            #
;#              Anz A  |06   13|       Anz Common                              E
        C                                                                     #
;#              Anz B  |07   12|       Anz    G                                #
          E   C                                                               #
;#              Anz C  |08   11|       Anz    F                                #
              DDDD  DP                                                        #
;#              anz D|09   10|        Anz    E                                #
                                                                              #
;#                          +---------+                                       #
                                                                              #
;#                                                                            #
;
################################################################################
```

```
;#                                                                      #
;#       Daten:                                                         #
;#               PIC:            16F628-20                            #
;#               Takt:           intern                               #
;#                                                                      #
;#       Version:                1.2.301103                           #
;#       Erstellt:               19.10.03                             #
;#       Letze Änderungen:       30.11.03                             #
;#                                                                      #
;#                                                                      #
;
;#######################################################################
;#                                                                      #
;#       Beschreibung:                                                  #
;#               Wenn an Pin 4 Lo anliegt,                              #
;#                wird einen Rechteckspannung für das LCD erzeugt,      #
;#               diese liegt an PortB.                                  #
;#               Liegt an Pin 4 High an, kann über Pin 15 zwischen      #
;#                gemeinsamer Anode(Lo) oder Kathode(High) gewählt werden.
;#                                                                      #
;#               An Pin 3 kann die Getriebeart(5 oder 6 Gang) eingestellt werden, #
;#               5 Gang bei Lo-Pegel und 6 Gang bei high Pegel.         #
;#                                                                      #
;#               Über die Pins 1 und 17 wird die Anzeige erhöht, bzw gesenkt.  #
;#                                                                      #
;#               Nach jedem Up/Down impulse muss ein Neutral impulse kommen,
;#               damit Up/Down wieder funktioniert.                     #
```

```
;#                                                                              #
;#              DP:                                                             #
;#                      Der Dezimalpunkt wird immer dann eingeschaltet, wenn    #
;#                      einer der Kontakte(Up/Down/Neutral/Leerlauf) gechlossen ist.    #
;#                                                                              #
;#                                                                              #
;#              Hinweis:                                                        #
;#                      Der Pic schaltet erst zwischen gem Anode/Kathode um,    #
;#                       wenn eine neue Zahl ausgegebn wird.                    #
;#                                                                              #
;#                                                                              #
; ################################Copyright#####################################
;#                                                                              #
;#              Copyright 2003 by. Jörg Vehlow                                  #
;#                      Joerg@JV-Coder.de                                       #
;#                                                                              #
;################################################################################
##

        __CONFIG _CP_ALL & _MCLRE_OFF & _INTRC_OSC_NOCLKOUT         &
_WDT_OFF & _LVP_OFF ;Copyprotection & Kein Reset Anschluss & Interner Oscilator &
Watchdog aus & LowV programmiereung aus

loop    Equ     0x20                    ;Variable für die Warteschleife
loop2   Equ 0x21                        ;Variable für die Warteschleife2
gang    Equ 0x23                        ;Variable für den aktuellen Gang
check   Equ 0x24                        ;Variable zum merken des Neutralkontaktes

        org    0x00             ;Startadresse nach Reset ist 0, hier startet der PIC
        clrf    PORTA                           ;PortA initialisieren

        BSF     CMCON, CM0                      ;Comperatoren abschalten 1
   BSF     CMCON, CM1              ;2
   BSF     CMCON, CM2                      ;3

   BSF          STATUS, RP0                    ;Auf Bank 1 umschalten
```

```
   clrf  TRISB                    ;PortB alles Ausgänge

   movlw      b'01111111'         ;PortA(7:0) sind Eingänge
        movwf TRISA

        bcf          STATUS, RP0          ;Auf Bank 0 zurück schalten

        movlw 0x00                ;initalisierung der Anzeige
        call  Segmente            ;Die Richtige Bitkombination für die in w
gespscherte Zahl laden
        movwf PORTB                    ; und auf PortB ausgeben

        movlw 0x01
        movwf gang                ;Für den Aktuellen Gang = 1
        call  Segmente
        movwf PORTB
        clrf  check               ;Für den neutral-merker = 0


MainLoop                          ;Der start des Eigentlichen Programms
        Call  Ausgabe             ;Anz-Ausgabe

        BTFSSPORTA,3              ;Leerlauf
        Call  Kontrollstelle2

        BTFSSPORTA,1              ;Neutral
        Call  Kontrollstelle2

        BTFSSPORTA,0              ;Down
        Call  Kontrollstelle2

        BTFSSPORTA,2              ;Up
        Call  Kontrollstelle2

        BTFSSPORTA,3              ;Leerlauf
        goto  RA3

        BTFSSPORTA,1              ;Neutral
        goto  RA1

        btfsc check,0                  ;War der Schalthebel wieder auf neutral?
        goto  Kontrollstelle3     ;Nein? Dann wieder zurück

        BTFSSPORTA,0              ;Down
        goto  RA0

        BTFSSPORTA,2              ;Up
        goto  RA2

Kontrollstelle
```

```
        movf   PORTB,W                  ;PORTB in W kopiren, da Bit 7 von
PORTB Common und Bit 7 von PORTA DP
        movwf PORTA                     ;PORTB auf PORTA kopieren, somit ist
DP = Common, also aus

        goto   MainLoop                 ;Und noch eine Runde

Kontrollstelle2

        comf   PORTB,W                  ;PORTB invertiet nach W kopieren
        movwf PORTA                     ;W nach PORTA kopieren -> Durch das
invertieren: DB ungleich Common
        return

Kontrollstelle3
        BTFSSPORTA,0                    ;Wenn Down
        goto   MainLoop
        BTFSSPORTA,1                    ;oder Neutral
        goto   MainLoop
        BTFSSPORTA,2                    ;oder Up
        goto   MainLoop
        BTFSSPORTA,3                    ;oder Leerlauf
        goto   MainLoop             ;dann muss der DP an bleiben
        goto   Kontrollstelle       ;sonst wird der DP wieder ausgeschaltet



RA0                                         ;Down
        incf   check,1                  ;Check auf 1 setzen

        movlw 0x02                  ;Wenn wir im 2ten Gang sind,
        subwf  gang,0               ; müssen wir in den ersten springen
        btfss  STATUS,Z
        goto   RA0s0                ;Fals nicht, geht es normal weiter,
        clrf   gang                 ; dann wird der Gangspeicher auf 0 gesetzt
        goto   RA0s2                ; und wir machen am Ende weiter

RA0s0
        movf   gang,1               ;Wenn wir im Leerlauf sind,
        btfss  STATUS,Z             ; müssen wir in den 1ten springen
        goto   RA0s1                ;Fals nicht, geht es normal weiter,
        movlw 0x01                  ; dann wird der Gangspeicher
        movwf gang                  ; auf 1 gesetzt
        goto   RA0s2                ; und wir machen am Ende weiter

RA0s1
        movlw 0x01                  ;Sind wir im 1ten,
        subwf  gang,0               ; geht es nicht mehr tiefer
        btfsc  STATUS,Z
        goto   MainLoop             ;Also springen wir gleich zum Anfang zurück
```

```
        decf    gang,1                  ;Wenn alles ganz normal ist, erniedrigen wir gang
um 1

RA0s2
        movf    gang,0                  ;gang wird in w geschrieben,
        call    Segmente                ; dann wird wieder die entsprechende
Bitkombination geladen
        movwf   PORTB                           ; und auf PortB geschrieben

        goto    MainLoop                ;Und wieder zum anfang zurück


RA1
        clrf    check                   ;Der neutral-check wird auf null zurück gesetzt
        goto    MainLoop                ;Und es geht wieder an den Anfang



RA2                                             ;Up
        incf    check,1                         ;Check auf 1 setzen

        movlw   0x01                    ;Sind wir in Gang 1,
        subwf   gang,0                  ; dann müssen wir in Gang 0 springen
        btfss   STATUS,Z
        goto    RA2s0                   ;Fals nicht, geht es weiter,
        clrf    gang                    ; sonst wird gang auf 0 gesetzt
        goto    RA2s2                   ; und wir springen zum Ende

RA2s0
        movf    gang,1                  ;Sind wir in Gang 0,
        btfss   STATUS,Z                ; dann müssen wir in Gang 2 springen
        goto    RA2s1                   ;Fals nicht, geht es weiter,
        movlw   0x02                    ; sonst wird gang
        movwf   gang                    ; auf 2 gesetzt
        goto    RA2s2                   ; und wir springen zum ende

RA2s1
        movlw   0x05                    ;Sind wir in Gang 5,
        btfsc   PORTA,4                         ; (oder 6 Gang?
        movlw   0x06                    ; bei 6 Gang müssen wir 6 abziehen)
        subwf   gang,0                  ; geht es nicht mehr höher
        btfsc   STATUS,Z
        goto    MainLoop                ; und wir springen zurück zum Anfang

        incf    gang,1                  ;Ist alles ganz normal, wird gang um 1 erhöht

RA2s2
        movf    gang,0                  ;gang wird in w geschrieben,
        call    Segmente                ; dann wird wieder die entsprechende
Bitkombination geladen
        movwf   PORTB                           ; und auf PortB geschrieben
```

```
        goto    MainLoop                ;Und wieder zum Anfang zurück


RA3                                             ;Leerlauf
        clrf    gang                    ;Gang auf null setzten
        movlw 0x00                      ;W auf 0
        call    Segmente                ;Zahl zu null holen
        movwf PORTB                             ;und auf PortB ausgeben

        movlw d'200'                    ;Warteschleife
        movwf loop2

RA3s0
        call    Ausgabe
        nop
        nop
        nop
        nop
        nop
        nop
        decfsz  loop2,1                         ;loop wird um einen erniedrigt
        goto    RA3s0                   ;wenn loop noch nicht null ist, geht es weiter

        goto MainLoop                           ;Und zurück zum Anfang


Segmente                                        ;Festlegung der Bitfolgen für die Zahlen
        BTFSSPORTA,6                            ;Bei gemeinsamer Kathode nächtsten
befehl überspringen
        goto    Segmente2
        addwf  PCL, 1                   ;Zu PCL(Programm Counter) wird die zahl aus w
addiert
        retlw   B'00111111'             ; 0
        retlw   B'00000110'             ; 1
        retlw   B'01011011'             ; 2
        retlw   B'01001111'             ; 3
        retlw   B'01100110'             ; 4
        retlw   B'01101101'             ; 5
        retlw   B'01111101'             ; 6
        retlw   B'00000111'             ; 7
        retlw   B'01111111'             ; 8
        retlw   B'01101111'             ; 9

Segmente2
        addwf  PCL, 1                   ;Zu PCL(Programm Counter) wird die zahl aus w
addiert
        retlw   B'11000000'             ; 0
        retlw   B'11111001'             ; 1
        retlw   B'10100100'             ; 2
        retlw   B'10110000'             ; 3
```

```
        retlw   B'10011001'             ; 4
        retlw   B'10010010'             ; 5
        retlw   B'10000010'             ; 6
        retlw   B'11111000'             ; 7
        retlw   B'10000000'             ; 8
        retlw   B'10010000'             ; 9


Ausgabe                                                 ;Ausgeben und invertieren
        movlw   d'110'          ;Die Schleife wird 220mal durchlaufen
        movwf   loop                    ;Das wird in loop gespeichert

Ausgabes0
        nop                                     ;dann warten wir ein wenig
        nop                                     ;und warten...
        nop                                     ;und warten...
        nop                                     ;und warten...
        nop                                     ;und warten...
        nop                                     ;und warten...

        decfsz  loop,1                  ;loop wird um einen erniedrigt
        goto    Ausgabes0               ;wenn loop noch nicht null ist, geht es weiter

        btfsc   PORTA,5                 ;LED oder LCD, bei LED skip next
        return
        comf    PORTB,1                 ;PortB wird invertiert, nur bei LCD!
        comf    PORTA,F                 ;ebenso PORTA, bzw nur der DP-Pin


        return                          ;und zurück

        end                                     ;ende des programms
```