

```

1
2
3 .INCLUDE "2313def.inc"
4 .INCLUDE "LCDinstr.inc"
5
6 ;definities van de stuurlijnen
7 .equ    E    = PD0
8 .equ    RW   = PD1
9 .equ    RS   = PD2
10 ;.equ    LCD_cntr= PORTD
11 ;.equ    LCD_data= PORTB
12 .equ    BF   = PB7
13
14 ;definities van de variabelen
15 .def    teken    = r0
16 .def    buffer   = r16
17 .def    counter  = r17
18 .def    counter2 = r19
19 .def    temp     = r18
20
21 .CSEG
22 .ORG    0x00
23     rjmp    main
24
25
26
27
28 ;*****
29 ; delay
30 ;*****
31
32 delay:
33     ; =====
34     ;     delay loop generator
35     ;     12000000 cycles:
36     ; -----
37     ; delaying 11999976 cycles:
38         ldi    R16, $3E
39     WGLOOP0: ldi    R17, $FD
40     WGLOOP1: ldi    R18, $FE
41     WGLOOP2:  dec   R18
42         brne   WGLOOP2
43         dec   R17
44         brne   WGLOOP1
45         dec   R16
46         brne   WGLOOP0
47     ; -----
48     ; delaying 24 cycles:
49         ldi    R16, $08
50     WGLOOP3:  dec   R16
51         brne   WGLOOP3
52     ; =====
53     ret
54
55
56 ;*****
57 ; busy_flag
58 ; deze sub test of LCD beschikbaar

```

```

59 ; is om nieuwe data of instructie
60 ; te ontvangen
61 ;*****
62
63 busy_flag: ldi temp,0b00000000
64           out DDRB,temp
65           cbi PORTD,RS
66           sbi PORTD,RW
67           sbi PORTD,E
68           nop
69           nop
70           sbic  PINB,BF
71           rjmp  busy_flag
72           cbi PORTD,E
73
74           ret
75
76
77 ;*****
78 ; init
79 ; deze sub initialiseerd poortD
80 ;*****
81
82 init:    ldi temp,0b11111111
83           out DDRD,temp
84           cbi PORTD,E
85           cbi PORTD,RS
86           cbi PORTD,RW
87
88           ret
89
90
91 ;*****
92 ; write_data
93 ; deze sub schrijft het teken
94 ; dat in buffer zit naar de LCD
95 ;*****
96
97 write_data: rcall  busy_flag
98             ldi temp,0b11111111
99             out DDRB,temp
100            sbi PORTD,RS
101            cbi PORTD,RW
102            sbi PORTD,E
103            out PORTB,buffer
104            cbi PORTD,E
105
106            ret
107
108 ;*****
109 ; write_instr
110 ; deze sub schrijft de instructie
111 ; die in buffer zit naar de LCD
112 ;*****
113
114 write_instr: rcall  busy_flag
115             ldi temp,0b11111111
116             out DDRB,temp

```

```

117         cbi PORTD,RS
118         cbi PORTD,RW
119         sbi PORTD,E
120         out PORTB,buffer
121         cbi PORTD,E
122
123         ret
124
125         ;*****
126         ; main programma
127         ; bevat de hoofdstructuur
128         ;*****
129
130 main:
131         ldi temp,RAMEND
132         out SPL,temp           ;stack pointer initialiseren
133
134         rcall  init
135
136         ldi buffer,set_lcd     ;stel de LCD-functie in
137         rcall  write_instr
138
139         ldi buffer,on_lcd      ;schakel LCD in
140         rcall  write_instr
141
142         ldi buffer,clear_lcd   ;wis display
143         rcall  write_instr
144
145         ldi buffer,entry_lcd   ;breng de LCD in invoer modus
146         rcall  write_instr
147
148         ; nu de data uit tabel1 halen en weergeven op regel 1
149
150         ldi counter,16        ;aantal tekens in counter laden
151         ldi ZL,LOW(tabel1*2)   ;low-pointer op begin tabel1 brengen
152         ldi ZH,HIGH(tabel1*2) ;high pointer op begin tabel1 brengen
153 loop1: lpm                    ;teken uit tabel halen
154         mov  buffer,teken      ;teken in buffer zetten
155         rcall  write_data
156         inc  ZL
157         brcc  no_carry1
158         inc  ZH
159 no_carry1:
160         dec  counter
161         brne  loop1
162
163         ; eerst een delay
164
165
166
167         ; eerst een delay van 3seconden
168
169         ; rcall  delay
170
171         ; nu de data uit tabel2 halen en weergeven op regel 2
172
173         ldi buffer,regel2_LCD
174         rcall  write_instr

```

```

175     ldi counter,11
176     ldi ZL,LOW(tabel2*2)    ;low-pointer op begin tabel2 brengen
177     ldi ZH,HIGH(tabel2*2)  ;high pointer op begin tabel2 brengen
178 loop2: lpm
179     mov  buffer,teken
180     rcall write_data
181     adiw ZL,1
182 no_carry2:
183     dec counter
184     brne loop2
185
186     ldi buffer,0b10001000
187     rcall write_instr
188
189     ldi buffer,0b11011111
190     rcall write_data
191
192     ldi buffer,0b11111111
193     rcall write_instr
194
195
196 nel:   rjmp nel
197
198
199
200 ;TABELLEN
201
202 tabel1: .DB "Temp: 21 C    AL"
203
204 tabel2: .DB "Tijd: 16u34"
205
206

```