



Report

Laboratory 5



December 7, 2017

Author:

Caroline Nilsson (*cn222nd*)

Daniel Alm Grundström (*dg222dw*)

Term: HT 2017

Course: 1DT301 - Computer
Technology I

Contents

1	Assignment 1	1
2	Assignment 2	3
3	Assignment 3	6
4	Assignment 4	9

1 Assignment 1

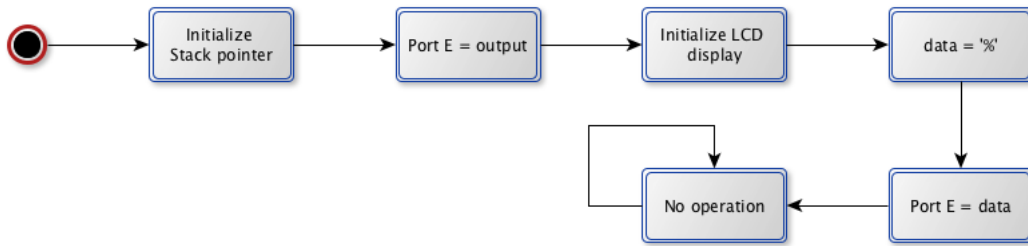


Figure 1: Display the character %

[illegible]

```

75
76 ; Write character to LCD
77 ldi data, PERCENT_CHAR
78 rcall write_char
79
80 main_loop:
81 nop
82 rjmp main_loop
83
84 ; Display subroutines
85 init_display:
86 rcall power_up_wait ; Wait for display to power up
87
88 ldi data, BITMODE4 ; Set 4-bit operation
89 rcall write_nibble
90 rcall short_wait
91 ldi data, DISP_CTRL
92 rcall write_cmd
93 rcall short_wait
94
95 clear_display:
96 ldi data, CLEAR
97 rcall write_cmd
98 rcall long_wait
99 ret
100
101 ; Write subroutines
102 write_char:
103 ldi RS, RS_ON
104 rjmp write
105
106 write_cmd:
107 clr RS
108
109 write:
110 mov temp, data
111 andi data, 0b1111_0000 ; Clear lower nibble
112 swap data
113 or data, RS ; Add RS to command to write
114 rcall write_nibble ; send high nibble
115 mov data, temp
116 andi data, 0b0000_1111 ; Clear high nibble
117 or data, RS
118
119 write_nibble:
120 rcall switch_output
121 nop
122 sbi LCD_PORT, 5
123 nop
124 nop
125 cbi LCD_PORT, 5
126 nop
127 nop
128 ret
129
130 ; Wait subroutines
131 short_wait:
132 clr ZH
133 ldi ZL, 30
134 rjmp wait_loop
135 long_wait:
136 ldi ZH, HIGH(1000)
137 ldi ZH, LOW(1000)
138 rjmp wait_loop
139 dbnc_wait:
140 ldi ZH, HIGH(4600)
141 ldi ZL, LOW(4600)
142 rjmp wait_loop
143 power_up_wait:
144 ldi ZH, HIGH(9000)
145 ldi ZL, LOW(9000)
146
147 wait_loop:
148 sbiw Z, 1
149 brne wait_loop
150 ret
151
152 ; Modify output to fit LCD JHD202C
153 switch_output:
154 push temp
155 clr temp
156
157 sbrc data, 0 ; If D4 set
158 ori temp, 0b0000_0100 ; then set PIN3
159 sbrc data, 1 ; If D5 set
160 ori temp, 0b0000_1000 ; then set PIN4
161 sbrc data, 2 ; If D6 set
162 ori temp, 0b0000_0001 ; then set PIN0
163 sbrc data, 3 ; If D7 set
164 ori temp, 0b0000_0010 ; then set PIN1
165 sbrc data, 4 ; If E set
166 ori temp, 0b0010_0000 ; then set PIN5
167 sbrc data, 5 ; If RS set
168 ori temp, 0b1000_0000 ; then set PIN7
169
170 out LCD_PORT, temp
171 pop temp
172 ret

```

2 Assignment 2

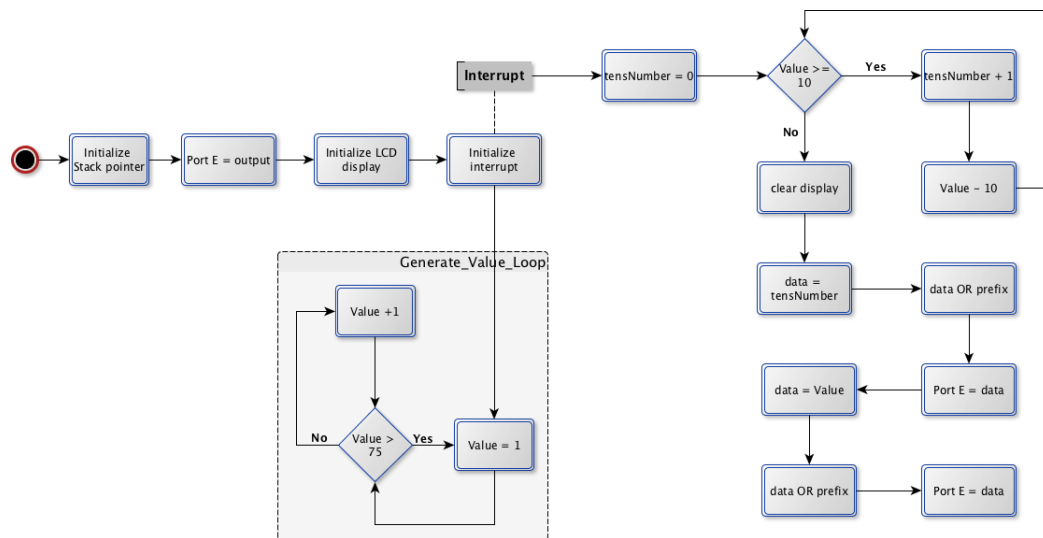


Figure 2: Generate random numbers: 1 – 75

[illegible]

```

.equ CLEAR = 0b0000_0001
61 .equ DISP_CTRL = 0b0000_1111 ; Display on, cursor on, blink on.
62 .equ RS_ON = 0b0010_0000
63 .equ LCD_PORT = PORTE ; Port LCD is connected to
64 .equ LCD_DATA_DIR = DDRE ; Data dir. of port LCD is connected to
65 .equ SWITCH_PORT = PORTD
66 .equ SWITCH_DATA_DIR = DDRD
67 .equ PREFIX = 0b0011_0000 ; Prefix for outputting number on LCD
68 .equ VAL_MAX = 75
69 .equ VAL_MIN = 1
70
71 .cseg
72 .org 0x00
73 jmp reset
74
75 .org int0addr
76 jmp switch0_interrupt
77
78 .org 0x72
79
80 reset:
81
82 ; Init stack pointer
83 ldi temp, HIGH(RAMEND)
84 out SPH, temp
85 ldi temp, LOW(RAMEND)
86 out SPL, temp
87
88 ; set LCD output port
89 ser temp
90 out LCD_DATA_DIR, temp
91
92 clr temp
93 out SWITCH_DATA_DIR, temp
94
95 ; Initialize display
96 rcall init_display
97
98 ;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
99 ; enable external interrupt on PIND0
100 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
101 ldi temp, (1<<int0)
102 out EIMSK, temp
103
104 ;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
105 ; set interrupt sense control to "Falling edge"
106 <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<
107 ldi temp, (3<<ISC00)
108 sts EICRA, temp
109
110 sei
111
112 rjmp reset_value
113
114 generate_value_loop:
115 cpi value, VAL_MAX
116 brge reset_value
117 inc value
118 rjmp generate_value_loop
119
120 reset_value:
121 ldi value, VAL_MIN
122 rjmp generate_value_loop
123
124 ; Display subroutines
125 init_display:
126 rcall power_up_wait ; Wait for display to power up
127
128 ldi data, BITMODE4 ; Set 4-bit operation
129 rcall write_nibble
130 rcall short_wait
131 ldi data, DISP_CTRL
132 rcall write_cmd
133 rcall short_wait
134
135 clear_display:
136 ldi data, CLEAR
137 rcall write_cmd
138 rcall long_wait
139 ret
140
141 ; Write subroutines
142 write_char:
143 ldi RS, RS_ON
144 rjmp write
145
146 write_cmd:
147 clr RS
148
149 write:
150 mov temp, data
151 andi data, 0b1111_0000 ; Clear lower nibble
152 swap data
153 or data, RS ; Add RS to command to write
154 rcall write_nibble ; send high nibble
155 mov data, temp
156 andi data, 0b0000_1111 ; Clear high nibble
157 or data, RS
158
159 write_nibble:
160 rcall switch_output

```

```

161     nop
162     sbi LCD_PORT, 5
163     nop
164     nop
165     cbi LCD_PORT, 5
166     nop
167     nop
168     ret
169
170 ; Wait subroutines
171 short_wait:
172     clr ZH
173     ldi ZL, 30
174     rjmp wait_loop
175 long_wait:
176     ldi ZH, HIGH(1000)
177     ldi ZH, LOW(1000)
178     rjmp wait_loop
179 dbnc_wait:
180     ldi ZH, HIGH(4600)
181     ldi ZL, LOW(4600)
182     rjmp wait_loop
183 power_up_wait:
184     ldi ZH, HIGH(9000)
185     ldi ZL, LOW(9000)
186
187 wait_loop:
188     sbiw Z, 1
189     brne wait_loop
190     ret
191
192 ; Modify output to fit LCD JHD202C
193 switch_output:
194     push temp
195     clr temp
196
197     sbrc data, 0                ; If D4 set
198     ori temp, 0b0000_0100      ; then set PIN3
199     sbrc data, 1                ; If D5 set
200     ori temp, 0b0000_1000      ; then set PIN4
201     sbrc data, 2                ; If D6 set
202     ori temp, 0b0000_0001      ; then set PIN0
203     sbrc data, 3                ; If D7 set
204     ori temp, 0b0000_0010      ; then set PIN1
205     sbrc data, 4                ; If E set
206     ori temp, 0b0010_0000      ; then set PIN5
207     sbrc data, 5                ; If RS set
208     ori temp, 0b1000_0000      ; then set PIN7
209
210     out LCD_PORT, temp
211     pop temp
212     ret
213
214 switch0_interrupt:
215     in temp, SREG
216     push temp
217
218     mov tempValue, value
219     lds temp, PORTD
220
221 sw0_loop:
222     ldi r31, 130
223     ldi r30, 222
224 L1: dec r30
225     brne L1
226     dec r31
227     brne L1
228     nop
229
230     lds r29, PORTD
231     cp temp, r29
232     brne sw0_loop
233
234     ldi tensNumber, 0
235
236 subtract_loop:
237     cpi tempValue, 10
238     brge subtract
239
240     rcall clear_display
241
242     mov data, tensNumber
243     ori data, PREFIX
244     rcall write_char
245     rcall long_wait
246
247     mov data, tempValue
248     ori data, PREFIX
249     rcall write_char
250
251     pop temp
252     out SREG, temp
253     reti
254
255 subtract:
256     subi tempValue, 10
257     inc tensNumber
258     rjmp subtract_loop

```

3 Assignment 3

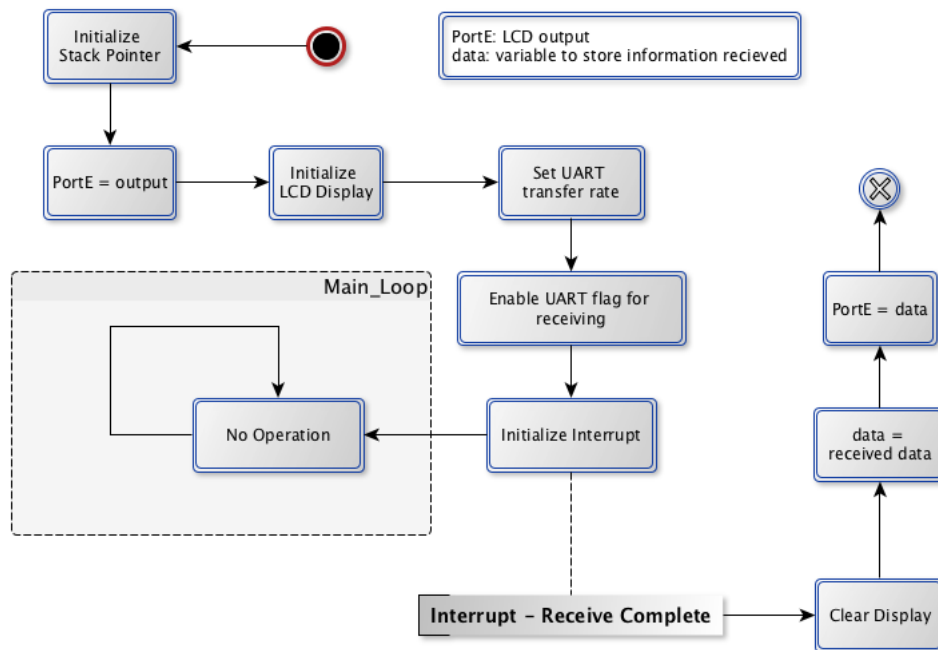


Figure 3: Receive character from serial port and display it

[illegible]


```

154     ldi ZH, LOW(1000)
155     rjmp wait_loop
156 dbnc_wait:
157     ldi ZH, HIGH(4600)
158     ldi ZL, LOW(4600)
159     rjmp wait_loop
160 power_up_wait:
161     ldi ZH, HIGH(9000)
162     ldi ZL, LOW(9000)
163
164 wait_loop:
165     sbiw Z, 1
166     brne wait_loop
167     ret
168
169 ; Modify output to fit LCD JHD202C
170 switch_output:
171     push temp
172     clr temp
173
174     sbrc data, 0           ; If D4 set
175     ori temp, 0b0000_0100 ; then set PIN3
176     sbrc data, 1           ; If D5 set
177     ori temp, 0b0000_1000 ; then set PIN4
178     sbrc data, 2           ; If D6 set
179     ori temp, 0b0000_0001 ; then set PIN0
180     sbrc data, 3           ; If D7 set
181     ori temp, 0b0000_0010 ; then set PIN1
182     sbrc data, 4           ; If E set
183     ori temp, 0b0010_0000 ; then set PIN5
184     sbrc data, 5           ; If RS set
185     ori temp, 0b1000_0000 ; then set PIN7
186
187     out LCD_PORT, temp
188     pop temp
189     ret
190
191 data_received_interrupt:
192     rcall clear_display
193
194     lds data, UDRI
195     out PORTB, data
196     rcall write_char
197
198     reti

```

4 Assignment 4

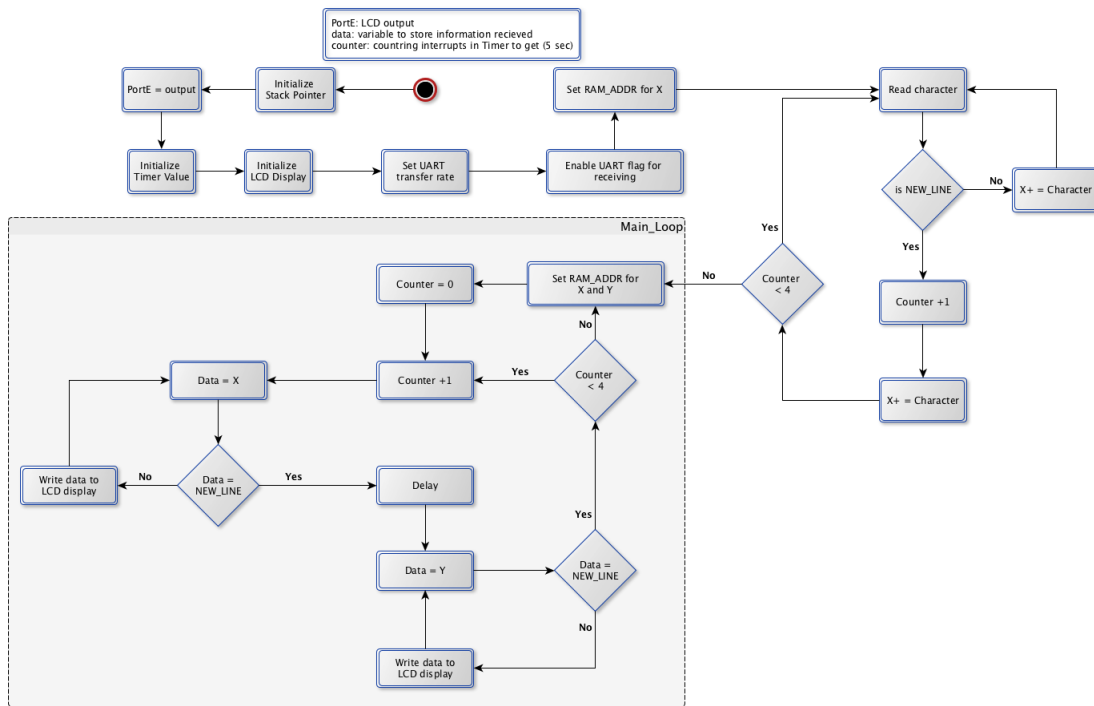


Figure 4: Receive 4 lines from serial port and display each line 5 sec in a loop

```

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
2 ; IDT301, Computer Technology I
3 ; Date: 2017-10-30
4 ; Author:
5 ; Caroline Nilsson (cn222nd)
6 ; Daniel Alm Grundström (dg222dw)
7 ;
8 ; Lab number: 5
9 ; Title: Display JHD202
10 ;
11 ; Hardware: STK600, CPU ATmega2560, LCD JHD202
12 ;
13 ; Function: Recieve 4 lines on serial port show each
14 ; line during 5 seconds
15 ;
16 ; Input ports: RS232
17 ;
18 ; Output ports: PORTE
19 ;
20 ; Subroutines: four_row_loop: writes 4 lines to display
21 ; then start from the beginning
22 ; init_display: initialize Display
23 ; clear_display: clear display
24 ; write_char: set RS = RS_ON
25 ; write_cmd: clear RS
26 ; write: write to display
27 ; write_nibble: write nibble to display
28 ; (subroutine of write)
29 ; short_wait: delay
30 ; long_wait: delay
31 ; dbnc_wait: delay
32 ; power_up_wait: delay
33 ; wait_loop: delay
34 ; switch_output: modify output to fit display
35 ; read_lines: collect char, check if it is
36 ; NEW_LINE
37 ; store_char: store char to X
38 ; read_lines_end: store NEW_LINE to X
39 ; write_main: write char from X
40 ; write_lines_end: return
41 ; write_new_lines: clear display
42 ; write_new_line: write 40 space to display
43 ; (to write on second line)
44 ; write_second_line: write char from Y
45 ; write_second_line_end: return
46 ; delay_5sec: 5 sec delay
47 ;
48 ; Included files: m2560def.inc
49 ;
50 ; Other information: "#" = NEW_LINE

```



```

152     ret
153
154 ; Write subroutines
155 write_char:
156     ldi RS, RS_ON
157     rjmp write
158
159 write_cmd:
160     clr RS
161
162 write:
163     mov temp, data
164     andi data, 0b1111_0000           ; Clear lower nibble
165     swap data
166     or data, RS                      ; Add RS to command to write
167     rcall write_nibble              ; send high nibble
168     mov data, temp
169     andi data, 0b0000_1111         ; Clear high nibble
170     or data, RS
171
172 write_nibble:
173     rcall switch_output
174     nop
175     sbi LCD_PORT, 5
176     nop
177     nop
178     cbi LCD_PORT, 5
179     nop
180     nop
181     ret
182
183 ; Wait subroutines
184 short_wait:
185     clr ZH
186     ldi ZL, 30
187     rjmp wait_loop
188 long_wait:
189     ldi ZH, HIGH(1000)
190     ldi ZH, LOW(1000)
191     rjmp wait_loop
192 dbnc_wait:
193     ldi ZH, HIGH(4600)
194     ldi ZL, LOW(4600)
195     rjmp wait_loop
196 power_up_wait:
197     ldi ZH, HIGH(9000)
198     ldi ZL, LOW(9000)
199
200 wait_loop:
201     sbiw Z, 1
202     brne wait_loop
203     ret
204
205 ; Modify output to fit LCD JHD202C
206 switch_output:
207     push temp
208     clr temp
209
210     sbrc data, 0           ; If D4 set
211     ori temp, 0b0000_0100 ; then set PIN3
212     sbrc data, 1           ; If D5 set
213     ori temp, 0b0000_1000 ; then set PIN4
214     sbrc data, 2           ; If D6 set
215     ori temp, 0b0000_0001 ; then set PIN0
216     sbrc data, 3           ; If D7 set
217     ori temp, 0b0000_0010 ; then set PIN1
218     sbrc data, 4           ; If E set
219     ori temp, 0b0010_0000 ; then set PIN5
220     sbrc data, 5           ; If RS set
221     ori temp, 0b1000_0000 ; then set PIN7
222
223     out LCD_PORT, temp
224     pop temp
225     ret
226
227 read_lines:
228     lds temp, UCSRIA
229     sbrs temp, RXCI ;if RXC flag is clear
230     rjmp read_lines ;then jump to start
231
232     lds data, UDRI ;load received data to ledState
233
234     cpi data, NEW_LINE
235     brne store_char
236
237     inc counter
238
239     cpi counter, 4
240     brge read_lines_end
241
242 store_char:
243     st X+, data
244
245     rjmp read_lines
246
247 read_lines_end:
248     ldi data, NEW_LINE
249     st X+, data
250     ret
251
252 ; write from X to the Display

```

```

253 ; until NEW_LINE
254 write_main:
255     ld data, X+
256
257     cpi data, NEW_LINE
258     breq write_lines_end
259
260     rcall write_char
261     rcall long_wait
262     rjmp write_main
263
264 write_lines_end:
265     ret
266
267 write_new_lines:
268     push counter
269     rcall clear_display
270     ldi counter, 40
271
272 write_new_line:
273     ldi data, 0b0010_0000
274     rcall write_char
275     rcall short_wait
276
277     dec counter
278     cpi counter, 1
279     brge write_new_line
280
281     rcall write_second_line
282
283     ldi data, 0b0000_0010
284     rcall write_cmd
285     rcall long_wait
286
287     pop counter
288     ret
289
290 write_second_line:
291     ld data, Y+
292
293     cpi data, NEW_LINE
294     breq write_second_line_end
295
296     rcall write_char
297     rcall long_wait
298
299     rjmp write_second_line
300
301 write_second_line_end:
302     ret
303
304 delay_5sec:
305     push r18
306     push r19
307     push r20
308
309     ldi r18, 26
310     ldi r19, 94
311     ldi r20, 111
312 L1: dec r20
313     brne L1
314     dec r19
315     brne L1
316     dec r18
317     brne L1
318     nop
319
320     pop r20
321     pop r19
322     pop r18
323     ret

```