**Task 1**

;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

; 1DT301, Computer Technology I

; Date: 2017-10-09

; Author:

; Student name 1 Ruth Dirnfeld

; Student name 2 Alexandra Bjäremo

;

; Lab number: 5

; Title: Display JHD202

;

; Hardware: STK600, CPU ATmega2560

;

; Function: A program that displays a character on the LCD display.

;

; Input ports: None.

;

; Output ports: LCD display connected to DDRE.

;

; Subroutines: Display initialization.

; Included files: m2560def.inc

;

; Other information: Clock set at 1MHz.

;

; Changes in program:

;

;<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

.include "m2560def.inc"

.def Temp = r16

.def Data = r17

.def RS = r18

.equ BITMODE4 = 0b00000010 ; 4..bit operation

.equ CLEAR = 0b00000001 ; Clear display

.equ DISPCTRL = 0b00001111 ; Display on, cursor on, blink on.

.cseg

.org 0x0000 ; Reset vector

jmp reset

.org 0x0072

reset:

ldi Temp, HIGH (RAMEND) ; Temp = high byte of ramend address

out SPH, Temp ; sph = Temp

ldi Temp, LOW (RAMEND) ; Temp = low byte of ramend address

out SPL, Temp ; spl = Temp

ser Temp ; r16 = 0b11111111

out DDRE, Temp ; port E = outputs (Display JHD202A)

clr Temp ; r16 = 0

out PORTE, Temp

init\_disp:

rcall power\_up\_wait ; wait for display to power up

ldi Data, BITMODE4 ; 4..bit operation

rcall write\_nibble ; (in 8..bit mode)

rcall short\_wait ; wait min. 39 us

ldi Data, DISPCTRL ; disp. on, blink on, curs. On

rcall write\_cmd ; send command

rcall short\_wait ; wait min. 39 us

rcall clr\_disp

ldi Data, 0b00100101

rcall write\_char

loop: nop

rjmp loop ; loop forever

clr\_disp:

ldi Data, CLEAR ; clr display

rcall write\_cmd ; send command

rcall long\_wait ; wait min. 1.53 ms

ret

;

; write char/command

;

write\_char:

ldi RS, 0b00100000 ; RS = high

rjmp write

write\_cmd:

clr RS ; RS = low

write:

mov Temp, Data ; copy Data

andi Data, 0b11110000 ; mask out high nibble

swap Data ; swap nibbles

or Data, RS ; add register select

rcall write\_nibble ; send high nibble

mov Data, Temp ; restore Data

andi Data, 0b00001111 ; mask out low nibble

or Data, RS ; add register select

write\_nibble:

rcall switch\_output ; Modify for display JHD202A, port E

nop ; wait 542nS

sbi PORTE, 5 ; enable high, JHD202A

nop

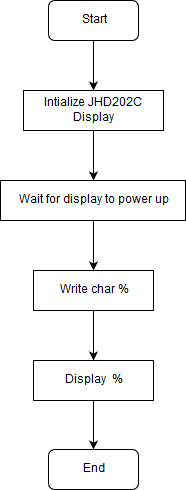
nop ; wait 542nS

cbi PORTE, 5 ; enable low, JHD202A

nop

nop ; wait 542nS

ret

;

; busy\_wait loop

;

short\_wait:

clr zh ; approx 50 us

ldi zl, 30

rjmp wait\_loop

long\_wait:

ldi zh, HIGH (1000) ; approx 2 ms

ldi zl, LOW (1000)

rjmp wait\_loop

dbnc\_wait:

ldi zh, HIGH (4600) ; approx 10 ms

ldi zl, LOW (4600)

rjmp wait\_loop

power\_up\_wait:

ldi zh, HIGH (9000) ; approx 20 ms

ldi zl, LOW (9000)

wait\_loop:

sbiw z, 1 ; 2 cycles

brne wait\_loop ; 2 cycles

ret

switch\_output:

push Temp

clr Temp

sbrc Data, 0 ; D4 = 1?

ori Temp, 0b00000100 ; Set pin 2

sbrc Data, 1 ; D5 = 1?

ori Temp, 0b00001000 ; Set pin 3

sbrc Data, 2 ; D6 = 1?

ori Temp, 0b00000001 ; Set pin 0

sbrc Data, 3 ; D7 = 1?

ori Temp, 0b00000010 ; Set pin 1

sbrc Data, 4 ; E = 1?

ori Temp, 0b00100000 ; Set pin 5

sbrc Data, 5 ; RS = 1?

ori Temp, 0b10000000 ; Set pin 7 (wrong in previous version)

out porte, Temp

pop Temp

ret

; source: lecture slides 9

/\*Description

\*The program displays the character % on the LCD display, that is connected to PORTE.

\*/

**Task 2**

;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

; 1DT301, Computer Technology I

; Date: 2017-10-09

; Author:

; Student name 1 Ruth Dirnfeld

; Student name 2 Alexandra Bjäremo

;

; Lab number: 5

; Title: Display JHD202

;

; Hardware: STK600, CPU ATmega2560

;

; Function: Electronic bingo machine

;

; Input ports: None.

;

; Output ports: LCD display connected to DDRE.

;

; Subroutines: Display initialization.

; Included files: m2560def.inc

;

; Other information: Clock set at 1MHz.

;

; Changes in program:

;

;<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

.include "m2560def.inc"

.def Temp = r16

.def Data = r17

.def RS = r18

.def small\_num = r19

.def tens\_num = r20

.equ BITMODE4 = 0b00000010 ; 4-bit operation

.equ CLEAR = 0b00000001 ; Clear display

.equ DISPCTRL = 0b00001111 ; Display on, cursor on, blink on. //DISP\_CTRL

.equ VAL\_MAX = 75

.equ VAL\_MIN = 1

.equ LCD = 0b0011\_0000 ; Prefix for outputting number on LCD

.cseg

.org 0x00

jmp reset

.org int0addr

jmp int\_generateRandom

.org 0x72

reset:

; Init stack pointer

ldi Temp, HIGH(RAMEND) ; Temp = high byte of ramend address

out SPH, Temp ; sph = Temp

ldi Temp, LOW(RAMEND) ; Temp = low byte of ramend address

out SPL, Temp ; spl = Temp

; set LCD output port

ser Temp ; r16 = 0b11111111

out DDRE, Temp ; port E = outputs ( Display JHD202A)

clr Temp ; r16 = 0

out DDRD, Temp

; Initialize display

rcall init\_disp

ldi Temp, (1<<int0)

out EIMSK, Temp

ldi Temp, (3<<ISC00)

sts EICRA, Temp

sei

rjmp reset\_value

value\_loop:

cpi small\_num, VAL\_MAX

brge reset\_value

inc small\_num

rjmp value\_loop

reset\_value:

ldi small\_num, VAL\_MIN

rjmp value\_loop

; Display subroutines

init\_disp:

rcall power\_up\_wait ; wait for display to power up

ldi Data, BITMODE4 ; 4-bit operation

rcall write\_nibble ; (in 8-bit mode)

rcall short\_wait ; wait min. 39 us

ldi Data, DISPCTRL ; disp. on, blink on, curs. On

rcall write\_cmd ; send command

rcall short\_wait ; wait min. 39 us

clr\_display:

ldi Data, CLEAR ; clr display

rcall write\_cmd ; send command

rcall long\_wait ; wait min. 1.53 ms

ret

; \*\*

; \*\* write char/command

; \*\*

write\_char:

ldi RS, 0b00100000 ; RS = high

rjmp write

write\_cmd:

clr RS ; RS = low

write:

mov Temp, Data ; copy Data

andi Data, 0b11110000 ; mask out high nibble

swap Data ; swap nibbles

or Data, RS ; add register select

rcall write\_nibble ; send high nibble

mov Data, Temp ; restore Data

andi Data, 0b00001111 ; mask out low nibble

or Data, RS ; add register select

write\_nibble:

rcall switch\_output ; Modify for display JHD202A, port E

nop ; wait 542nS

sbi PORTE, 5 ; enable high, JHD202A

nop

nop ; wait 542nS

cbi PORTE, 5 ; enable low, JHD202A

nop

nop ; wait 542nS

ret

; \*\*

; \*\* busy\_wait loop

; \*\*

short\_wait:

clr zh ; approx 50 us

ldi zl, 30

rjmp wait\_loop

long\_wait:

ldi zh, HIGH(1000) ; approx 2 ms

ldi zh, LOW(1000)

rjmp wait\_loop

dbnc\_wait:

ldi zh, HIGH(4600) ; approx 10 ms

ldi zl, LOW(4600)

rjmp wait\_loop

power\_up\_wait:

ldi zh, HIGH(9000) ; approx 20 ms

ldi zl, LOW(9000)

wait\_loop:

sbiw z, 1 ; 2 cycles

brne wait\_loop ; 2 cycles

ret

; \*\*

; \*\* modify output signal to fit LCD JHD202A, connected to port E

; \*\*

switch\_output:

push Temp

clr Temp

sbrc Data, 0 ; D4 = 1?

ori Temp, 0b00000100 ; Set pin 2

sbrc Data, 1 ; D5 = 1?

ori Temp, 0b00001000 ; Set pin 3

sbrc Data, 2 ; D6 = 1?

ori Temp, 0b00000001 ; Set pin 0

sbrc Data, 3 ; D7 = 1?

ori Temp, 0b00000010 ; Set pin 1

sbrc Data, 4 ; E = 1?

ori Temp, 0b00100000 ; Set pin 5

sbrc Data, 5 ; RS = 1?

ori Temp, 0b10000000 ; Set pin 7 (wrong in previous version)

out PORTE, Temp

pop Temp

ret

int\_generateRandom:

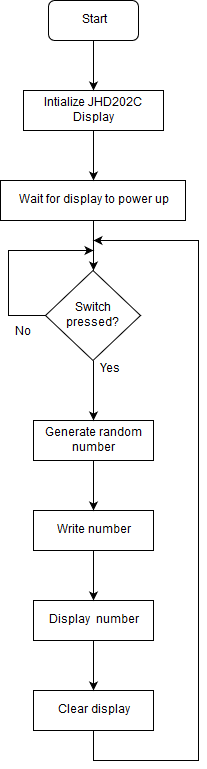
lds Temp, PORTD

delay:

ldi r31, 130

ldi r30, 222

L1: dec r30

 brne L1

dec r31

brne L1

nop

lds r29, PORTD

cp Temp, r29

brne delay

ldi tens\_num, 0

increase\_loop:

cpi small\_num, 10

brge increase

rcall clr\_display

ldi Data, LCD

or Data, tens\_num

rcall write\_char

ldi Data, LCD

or Data, small\_num

rcall write\_char

reti

increase:

subi small\_num, 10

inc tens\_num

rjmp increase\_loop

/\*Description

\*A program that simulates a bingo machine,

\* as in each time the switch is pressed a random

\*number \*between 0-75 will be generate and

\* shown on the LCD display.

\*/

**Task 3**

;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

; 1DT301, Computer Technology I

; Date: 2017-10-09

; Author:

; Student name 1 Ruth Dirnfeld

; Student name 2 Alexandra Bjäremo

;

; Lab number: 5

; Title: Display JHD202

;

; Hardware: STK600, CPU ATmega2560

;

; Function: Serial communication and display

;

; Input ports: TX, RX on PIND2, respective PIND3.

;

; Output ports: LCD display connected to DDRE.

;

; Subroutines: Display and serial communication initialization.

; Included files: m2560def.inc

;

; Other information: Clock set at 1MHz.

;

; Changes in program:

;

;<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

.include "m2560def.inc"

.def Temp = r16

.def Data = r17

.def RS = r18

.def char = r23

.equ BITMODE4 = 0b00000010 ; 4-bit operation

.equ CLEAR = 0b00000001 ; Clear display

.equ DISPCTRL = 0b00001111 ; Display on, cursor on, blink on.

.equ ubrr\_value = 12 ; use 4800 speed, set to 1 MHz

.cseg

.org 0x0000 ; Reset vector

jmp reset

.org URXC1addr ; interrupt address for USART

rjmp get\_char

.org 0x0072

reset:

ldi Temp, HIGH(RAMEND) ; Temp = high byte of ramend address

out SPH, Temp ; sph = Temp

ldi Temp, LOW(RAMEND) ; Temp = low byte of ramend address

out SPL, Temp ; spl = Temp

ser Temp ; r16 = 0b11111111

out DDRE, Temp ; port E = outputs ( Display JHD202A)

clr Temp ; r16 = 0

out PORTE, Temp

ldi Temp, ubrr\_value

sts UBRR1L, Temp

ldi Temp, (1<<TXEN1) | (1<<RXEN1) | (1<<RXCIE1) ; enable interrupt in USART

sts UCSR1B, Temp

; \*\*

; \*\* init\_display

; \*\*

init\_disp:

rcall power\_up\_wait ; wait for display to power up

ldi Data, BITMODE4 ; 4-bit operation

rcall write\_nibble ; (in 8-bit mode)

rcall short\_wait ; wait min. 39 us

ldi Data, DISPCTRL ; disp. on, blink on, curs. On

rcall write\_cmd ; send command

rcall short\_wait ; wait min. 39 us

sei

rcall clr\_disp

loop:

nop

rjmp loop ; loop forever

clr\_disp:

ldi Data, CLEAR ; clr display

rcall write\_cmd ; send command

rcall long\_wait ; wait min. 1.53 ms

ret

; \*\*

; \*\* write char/command

; \*\*

write\_char:

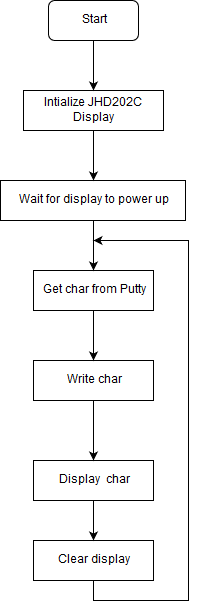
ldi RS, 0b00100000 ; RS = high

rjmp write

write\_cmd:

clr RS ; RS = low

write:

****mov Temp, Data ; copy Data

andi Data, 0b11110000 ; mask out high nibble

swap Data ; swap nibbles

or Data, RS ; add register select

rcall write\_nibble ; send high nibble

mov Data, Temp ; restore Data

andi Data, 0b00001111 ; mask out low nibble

or Data, RS ; add register select

write\_nibble:

rcall switch\_output ; Modify for display JHD202A, port E

nop ; wait 542nS

sbi PORTE, 5 ; enable high, JHD202A

nop

nop ; wait 542nS

cbi PORTE, 5 ; enable low, JHD202A

nop

nop ; wait 542nS

ret

; \*\*

; \*\* busy\_wait loop

; \*\*

short\_wait:

clr zh ; approx 50 us

ldi zl, 30

rjmp wait\_loop

long\_wait:

ldi zh, HIGH(1000) ; approx 2 ms

ldi zl, LOW(1000)

rjmp wait\_loop

dbnc\_wait:

ldi zh, HIGH(4600) ; approx 10 ms

ldi zl, LOW(4600)

rjmp wait\_loop

power\_up\_wait:

ldi zh, HIGH(9000) ; approx 20 ms

ldi zl, LOW(9000)

wait\_loop:

sbiw z, 1 ; 2 cycles

brne wait\_loop ; 2 cycles

ret

; \*\*

; \*\* modify output signal to fit LCD JHD202A, connected to port E

; \*\*

switch\_output:

push Temp

clr Temp

sbrc Data, 0 ; D4 = 1?

ori Temp, 0b00000100 ; Set pin 2

sbrc Data, 1 ; D5 = 1?

ori Temp, 0b00001000 ; Set pin 3

sbrc Data, 2 ; D6 = 1?

ori Temp, 0b00000001 ; Set pin 0

sbrc Data, 3 ; D7 = 1?

ori Temp, 0b00000010 ; Set pin 1

sbrc Data, 4 ; E = 1?

ori Temp, 0b00100000 ; Set pin 5

sbrc Data, 5 ; RS = 1?

ori Temp, 0b10000000 ; Set pin 7 (wrong in previous version)

out porte, Temp

pop Temp

ret

get\_char:

lds char, UCSR1A

lds Data, UDR1

rcall outLCD

reti

outLCD :

rcall write\_char

ret

/\*Description

\*A program that uses serial communication to retrieve characters that have been inputted from the keyboard into the Putty terminal so that they can then be sent and displayed on the JHD202C LED display.

\*/

**Task 4**

;>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

; 1DT301, Computer Technology I

; Date: 2017-10-09

; Author:

; Student name 1 Ruth Dirnfeld

; Student name 2 Alexandra Bjäremo

;

; Lab number: 5

; Title: Display JHD202

;

; Hardware: STK600, CPU ATmega2560

;

; Function: Serial communication while displaying with delay on different lines.

;

; Input ports: TX, RX on PIND2, PIND3.

;

; Output ports: LCD display connected to DDRE.

;

; Subroutines: If applicable.

; Included files: m2560def.inc

;

; Other information: Clock set at 1MHz.

;

; Changes in program:

;

;<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<

.include "m2560def.inc"

.def Temp = r16

.def Data = r17

.def RS = r18

.def counter = r19

.def spaces = r20

.equ BITMODE4 = 0b00000010 ; 4-bit operation

.equ CLEAR = 0b00000001 ; Clear display

.equ DISPCTRL = 0b00001111 ; Display on, cursor on

; blink on. //DISP\_CTRL

.equ TIMER\_VAL = 6

.cseg

.org 0x00

jmp reset

.org URXC1addr

jmp int\_interrupt

.org ovf0addr

jmp count\_interrupt

.org 0x72

reset:

; Init stack pointer

ldi Temp, HIGH(RAMEND) ; Temp = high byte of ramend address

out SPH, Temp ; sph = Temp

ldi Temp, LOW(RAMEND) ; Temp = low byte of ramend address

out SPL, Temp ; spl = Temp

; set LCD output port

ser Temp ; r16 = 0b11111111

out DDRE, Temp ; port E = outputs (Display JHD202A)

; Init display

rcall init\_disp

; Init Serial Communication

ldi Temp, 12 ; = 4800 bps (1MHz)

sts UBRR1L, Temp ;set transfer rate

ldi Temp, (1<<RXEN1) | (1<<RXCIE1)

sts UCSR1B, Temp ;enable UART flag for receiving

; Init Timer

ldi Temp, 0x05 ;set prescale to 1024

out TCCR0B, Temp

ldi Temp, (1<<TOIE0) ;enable overflow flag

sts TIMSK0, Temp

ldi Temp, TIMER\_VAL ;set default val for timer

out TCNT0, Temp

sei

main:

nop

rjmp main

; Display subroutines

init\_disp:

rcall power\_up\_wait ; wait for display to power up

ldi Data, BITMODE4 ; 4-bit operation

rcall write\_nibble ; (in 8-bit mode)

rcall short\_wait ; wait min. 39 us

ldi Data, DISPCTRL ; disp. on, blink on, curs. On

rcall write\_cmd ; send command

rcall short\_wait ; wait min. 39 us

clr\_display:

ldi Data, CLEAR ; clr display

rcall write\_cmd ; send command

rcall long\_wait ; wait min. 1.53 ms

ret

; \*\*

; \*\* write char/command

; \*\*

write\_char:

ldi RS, 0b00100000 ; RS = high

rjmp write

write\_cmd:

clr RS ; RS = low

write:

mov Temp, Data ; copy Data

andi Data, 0b11110000 ; mask out high nibble

swap Data ; swap nibbles

or Data, RS ; add register select

rcall write\_nibble ; send high nibble

mov Data, Temp ; restore Data

andi Data, 0b00001111 ; mask out low nibble

or Data, RS ; add register select

write\_nibble:

rcall switch\_output ; Modify for display JHD202A, port E

nop ; wait 542nS

sbi PORTE, 5 ; enable high, JHD202A

nop

nop ; wait 542nS

cbi PORTE, 5 ; enable low, JHD202A

nop

nop ; wait 542nS

ret

; \*\*

; \*\* busy\_wait loop

; \*\*

short\_wait:

clr zh ; approx 50 us

ldi zl, 30

rjmp wait\_loop

long\_wait:

ldi zh, HIGH(1000) ; approx 2 ms

ldi zh, LOW(1000)

rjmp wait\_loop

dbnc\_wait:

ldi zh, HIGH(4600) ; approx 10 ms

ldi zl, LOW(4600)

rjmp wait\_loop

power\_up\_wait:

ldi zh, HIGH(9000) ; approx 20 ms

ldi zl, LOW(9000)

wait\_loop:

sbiw z, 1 ; 2 cycles

brne wait\_loop ; 2 cycles

ret

; \*\*

; \*\* modify output signal to fit LCD JHD202A, connected to port E

; \*\*

switch\_output:

push Temp

clr Temp

sbrc Data, 0 ; D4 = 1?

ori Temp, 0b00000100 ; Set pin 2

sbrc Data, 1 ; D5 = 1?

ori Temp, 0b00001000 ; Set pin 3

sbrc Data, 2 ; D6 = 1?

ori Temp, 0b00000001 ; Set pin 0

sbrc Data, 3 ; D7 = 1?

ori Temp, 0b00000010 ; Set pin 1

sbrc Data, 4 ; E = 1?

ori Temp, 0b00100000 ; Set pin 5

sbrc Data, 5 ; RS = 1?

ori Temp, 0b10000000 ; Set pin 7

out PORTE, Temp

pop Temp

ret

int\_interrupt:

lds Data, UDR1

rcall write\_char

reti

count\_interrupt:

push Temp

ldi Temp, TIMER\_VAL

out TCNT0, Temp

inc counter

cpi counter, 20 ; five seconds

brlo timer\_end

ldi spaces, 0

ldi Data, 0b0000\_0101 ;move

rcall write\_cmd

move\_row:

ldi Data, 0b0010\_0000 ;space cmd

rcall write\_char

inc spaces

cpi spaces, 20

brlo move\_row

ldi data, 0b0000\_0100 ;default

rcall write\_cmd

clr counter

timer\_end:

pop Temp

reti