1a)

start:

ldi r16, 0xff

out DDRB, r16

ldi r16, 0xFE;

sec

call loop;

end: rjmp end

loop:

out portb, r16;

rol r16;

clr r0

ldi r18,0x90

ldi r19, 0x86

ldi r20, 0x01 ; loads a total value of 100,000

in r17, sreg;

call delay

rjmp loop

delay:

Ldi r17, 2

Ldi r16, 0x05

Out TCCR0, r16; Timer starts counting from 0 at Microcontroller clock rate/1024

Timer1:

Ldi r16, 6

Out TCNT0, R16 ; Corrects from 512ms to 500ms

Timer:

IN r16, TIFR

SBRS R16, 0; Checks the Top Overflow flag (TOV0) in the TIFR

Rjmp timer; loops until the flag gets set

Ldi r16, 0x01

Out TIFR, R16 ; Write a '1' over the TOV0 flag to clear it

Dec r17;

Brne timer ; Controls the number of loops for the delay. Using 2 to reach a .5s delay

ret

2a)

Init:

; initialize Stack Pointer (SP)

ldi r16, LOW(RAMEND)

out SPL, r16

ldi r16, HIGH(RAMEND)

out SPH, r16

; initialize Port B

ser r16 ; set all PortB pins to output

out DDRB, r16 ;

; initialize PortA

ldi r16, 0b11110000 ; set PA4-PA7 as outputs, PA0-PA3 as inputs

out DDRA, r16

ldi r16, 0b11111111 ; enable pull-up registers on PA4-PA7

out PORTA, r16

;=================

; Main body of program

Start:

ser r25 ; initialize LEDs to off

out PORTB , r25

Checkpress1:

Ldi ZL, low(2\*TextArray) ;R30

Ldi ZH, high(2\*TextArray); R31

call GetKeyPress ; check for a keypress

mov r24, r0 ; if key not pressed, check again

cpi r24, 0

breq Checkpress1

mov r25, r1 ; if key pressed, update display

ldi r16, 4

Mul r16,r25 ; multiply the button pos by 4

Add r30, r0;

Adc r31, r1; Add that value to the memory location of the text array

Release1:

call GetKeyPress ; check for button release

mov r24, r0 ; if key not released, check again

cpi r24, 0

brne Release1

Checkpress2:

call GetKeyPress ; check for a keypress

mov r24, r0 ; if key not pressed, check again

cpi r24, 0

breq Checkpress2

mov r25, r1 ; if key pressed, update display

clr r16

lsr r25

lsr r25 ; A = 0, D = 3

add r30, r25;

Adc R31, r16;

Lpm R25, Z;

com r25

out PORTB, r25 ; Displays the char selected

Release2:

call GetKeyPress ; check for button release

mov r24, r0 ; if key not released, check again

cpi r24, 0

brne Release

rjmp Checkpress1; repeat indefinitely

End:

rjmp End ; end of program