**Assembly Programs**

1. **Write a test program for the AVR to toggle all the bits of PORT A, PORT B and PORT D every 1/4th of a second.**

.INCLUDE "M32DEF.INC"

.ORG 00

LDI R16,HIGH(RAMEND)

OUT SPH,R16

LDI R16,LOW(RAMEND)

OUT SPL,R16

LDI R16,0x55

OUT PORTA,R16

OUT PORTB,R16

OUT PORTD,R16

HERE: LDI R16,0xFF

OUT PORTA,R16

OUT PORTB,R16

OUT PORTD,R16

CALL DELAY

COM R16

RJMP HERE

DELAY:

LDI R21,50

LOOP1: LDI R22,200

LOOP2: NOP

NOP

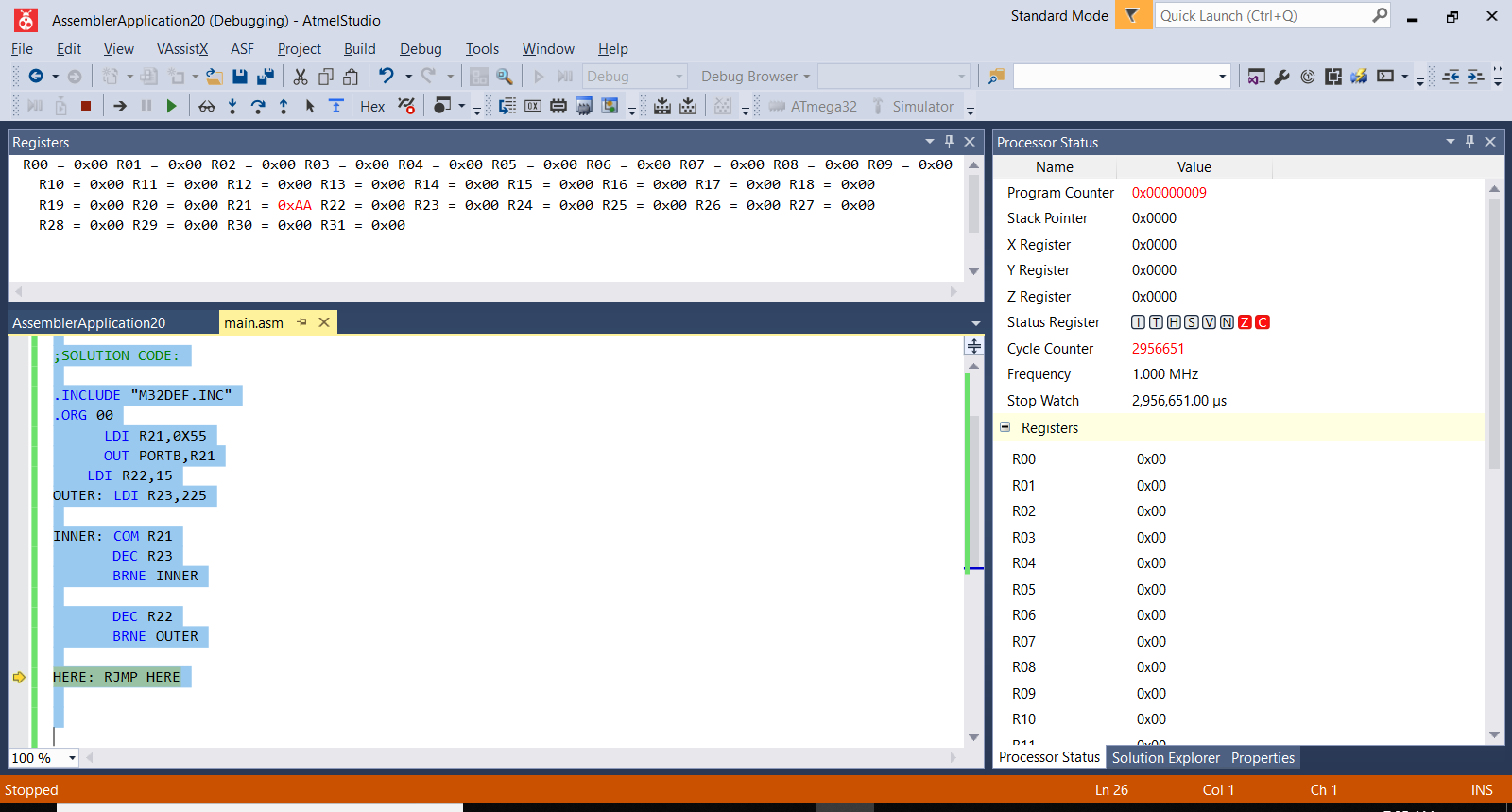
DEC R22

BRNE LOOP2

DEC R21

BRNE LOOP1

RET



1. **Toggle all the bits of PORT B by sending it to the values $11 and $AA continously.;For a time delay between each issuing of data to PORT B.**

.INCLUDE "M32DEF.INC"

.ORG 00

LDI R16,HIGH(RAMEND)

OUT SPH,R16

LDI R16,LOW(RAMEND)

OUT SPL,R16

BACK:

LDI R16,0x11

OUT PORTB,R16

CALL DELAY

LDI R16,0xAA

OUT PORTB,R16

CALL DELAY

RJMP BACK

delay subroutine

.ORG 0x300

DELAY: LDI R20,0xFF

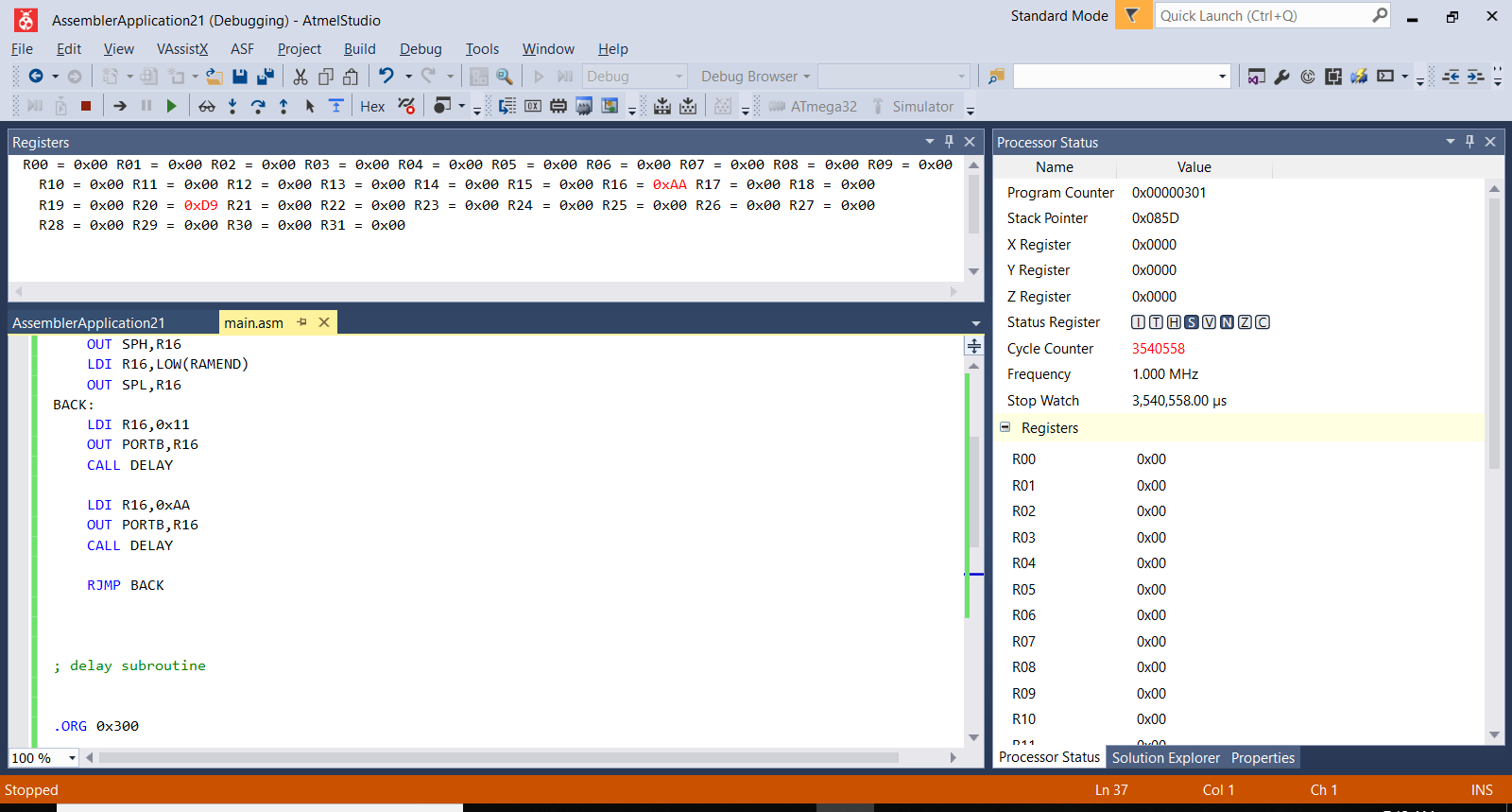
AGAIN:NOP

NOP

DEC R20

BRNE AGAIN

RET



1. **Find the sum of 0x44, 0x55, 0xFF and 0xE5. Put the sum into R21(low byte) and R22(high byte)**

.INCLUDE "M32DEF.INC"

.ORG 00

CLR R21

CLR R22

LDI R16, 0x44

ADD R21,R16

BRSH LOOP1

INC R22

LOOP1: LDI R16,0x55

ADD R21,R16

BRSH LOOP2

INC R22

LOOP2: LDI R16,0xFF

ADD R21,R16

BRSH LOOP3

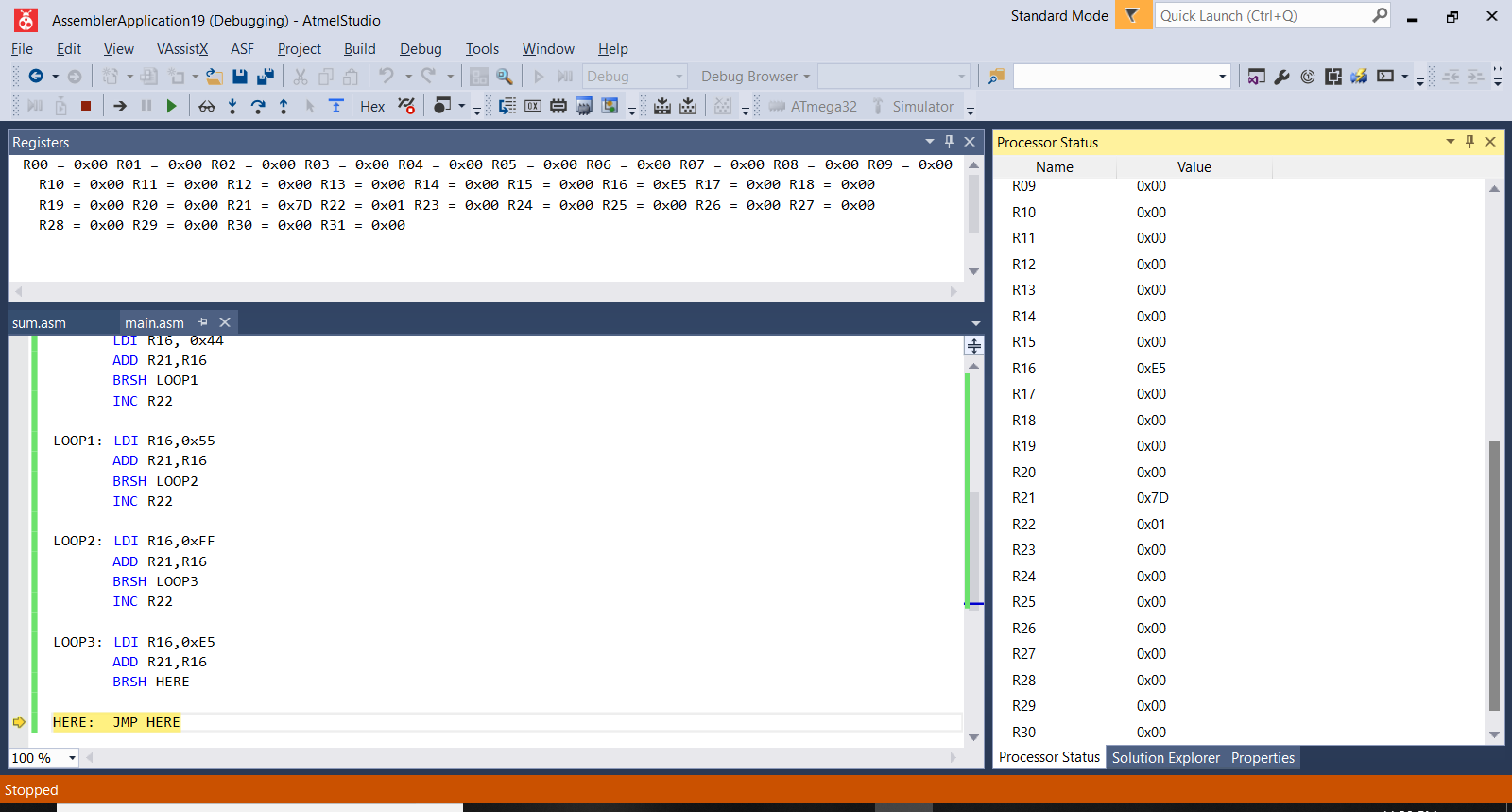
INC R22

LOOP3: LDI R16,0xE5

ADD R21,R16

BRSH HERE

HERE: JMP HERE



1. **A switch is connected to pin PB0 and an LED to pin PB7. Write a program to get the status of SW and send it to the LED**

.INCLUDE "M32DEF.INC"

.ORG 00

CBI DDRB, 0

SBI DDRB, 7

AGAIN: SBIC PINB, 0

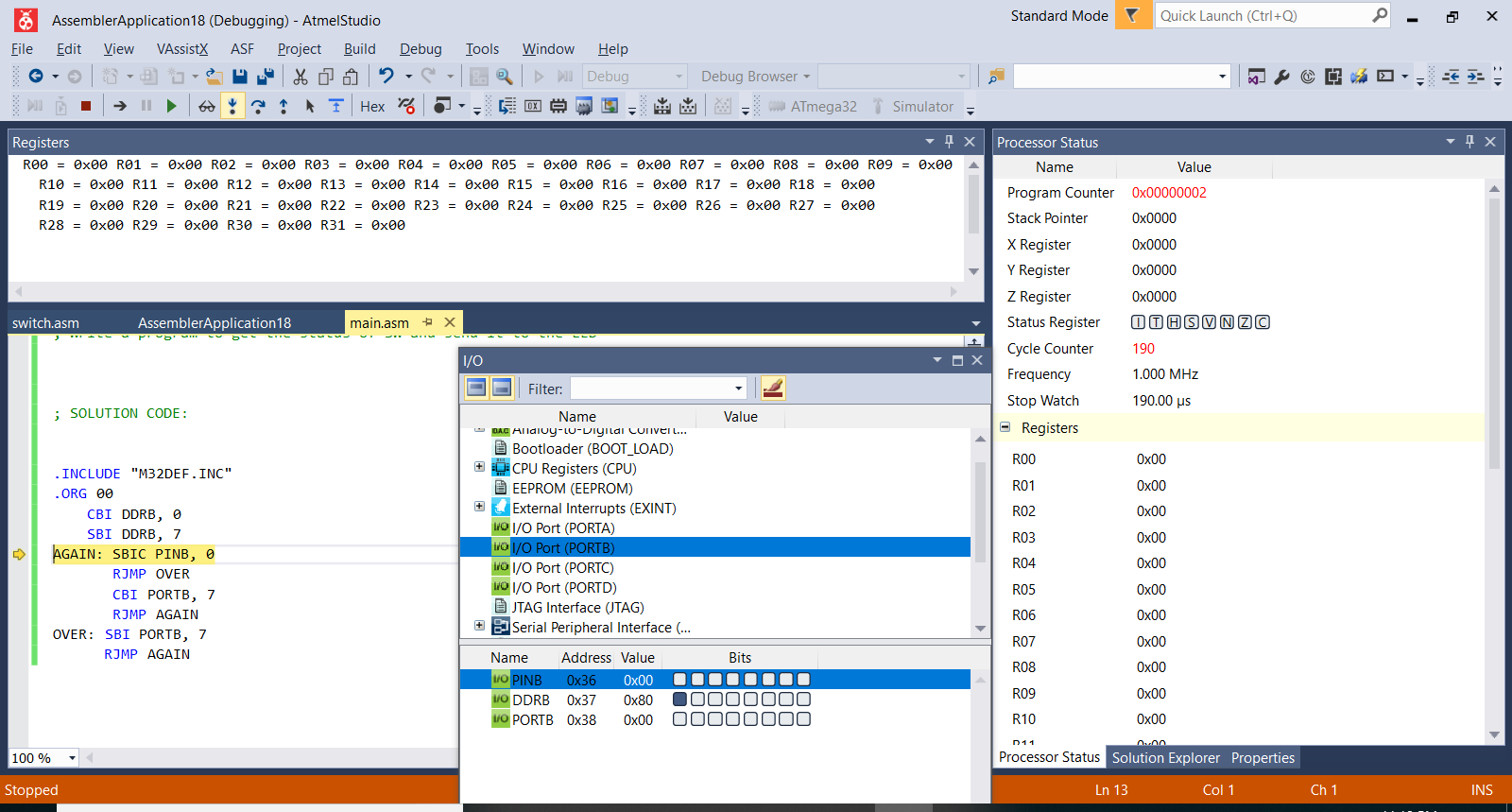
RJMP OVER

CBI PORTB, 7

RJMP AGAIN

OVER: SBI PORTB, 7

RJMP AGAIN



1. Write a program to

1. Load PORT A register with the value 0x55 an

2. Complement PORT A 3825 times.

.INCLUDE "M32DEF.INC"

.ORG 00

LDI R21,0X55

OUT PORTB,R21

LDI R22,15

OUTER: LDI R23,225

INNER: COM R21

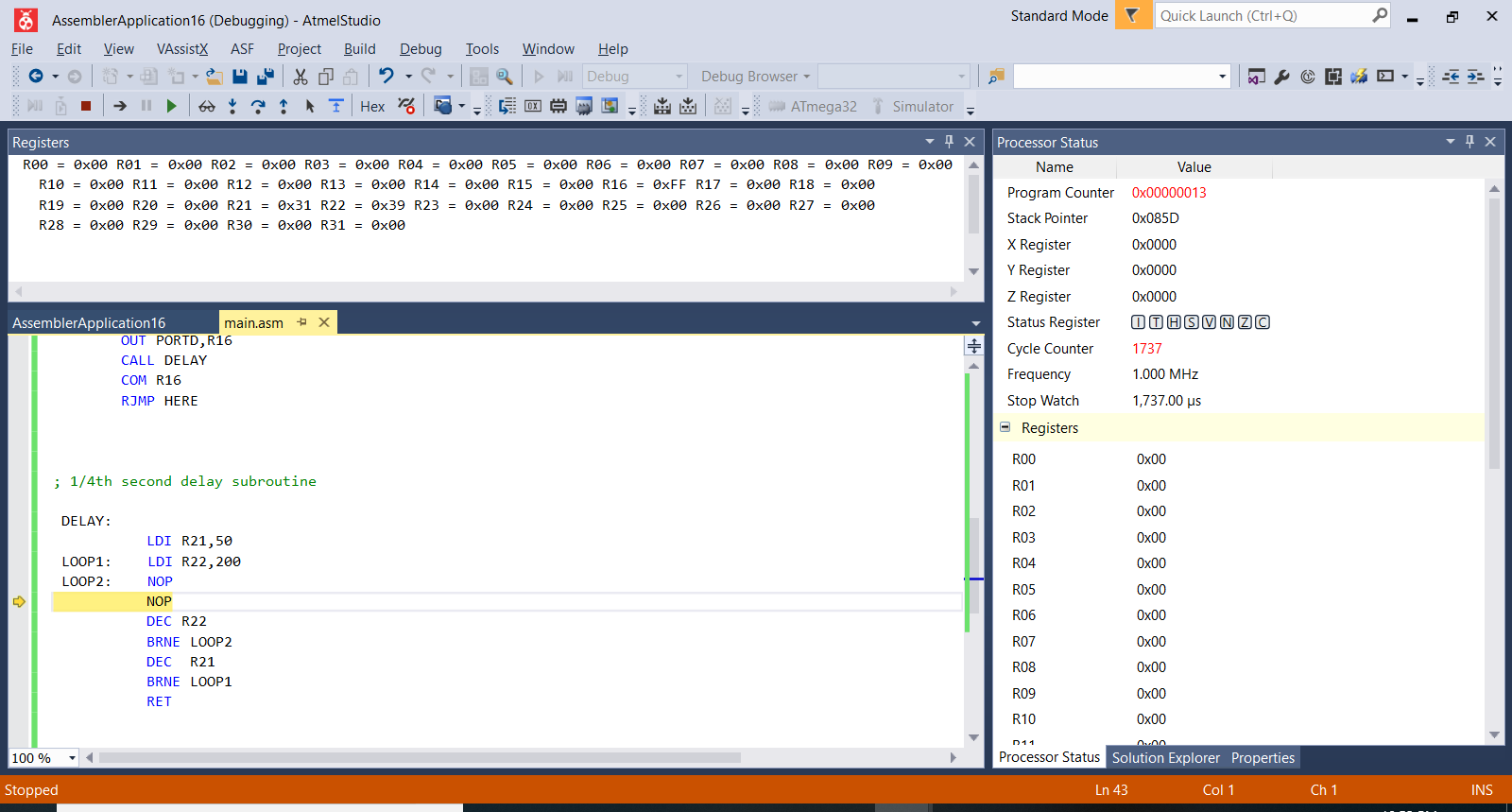
DEC R23

BRNE INNER

DEC R22

BRNE OUTER

HERE: RJMP HERE



4. PROGRAM WITH THE FOLLOWING PARTS:

a) SEND THE MESSAGE’YES’ ONCE TO THE PC SCREEN

b) GET DATA FROM SWITCHES ON PORTA AND TRANSMIT ITVIA THE SERIAL PORT TO THE PC’S SCREEN c) RECEIVE ANY KEY PRESS SENT BY HYPERTERMINAL AND PUT IT ON LEDS. PART b) AND c) SHOULD BE DONE REPEATEDLY

#include<avr/io.h>

#include<avr/interrupt.h>

ISR (USART\_UDRE\_vect)

{

UDR = PINA;

}

ISR (USART\_RXC\_vect)

{

PORTB = UDR;

}

int main(void)

{

UCSRB = (1<<RXEN)|(1<<RXCIE)|(1<<TXEN)|(1<<UDRIE);

UCSRC = (1<<UCSZ1)|(1<<UCSZ0)|(1<<URSEL);

DDRA = 0;

DDRB = 0XFF;

unsigned char str[3] = "YES";

unsigned char strlength = 3;

unsigned char i = 0;

if (UCSRA & (1<<UDRE)==1)

for (i = 0; i<strlength; i++)

UDR = str[i];

sei();

while(1);

}

