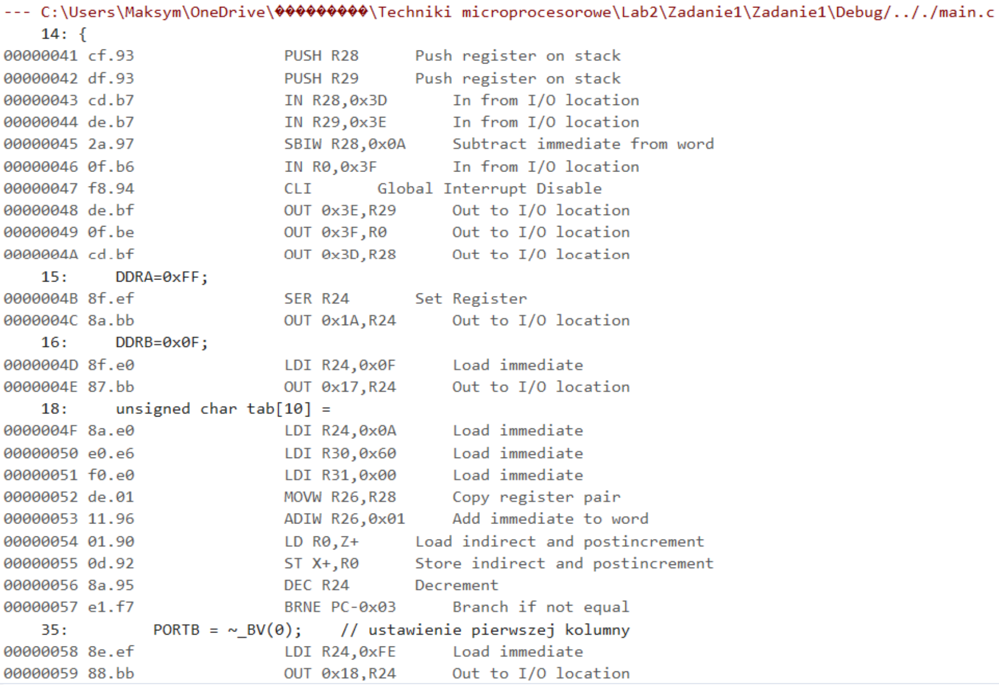
# Technika Mikroprocesorowa

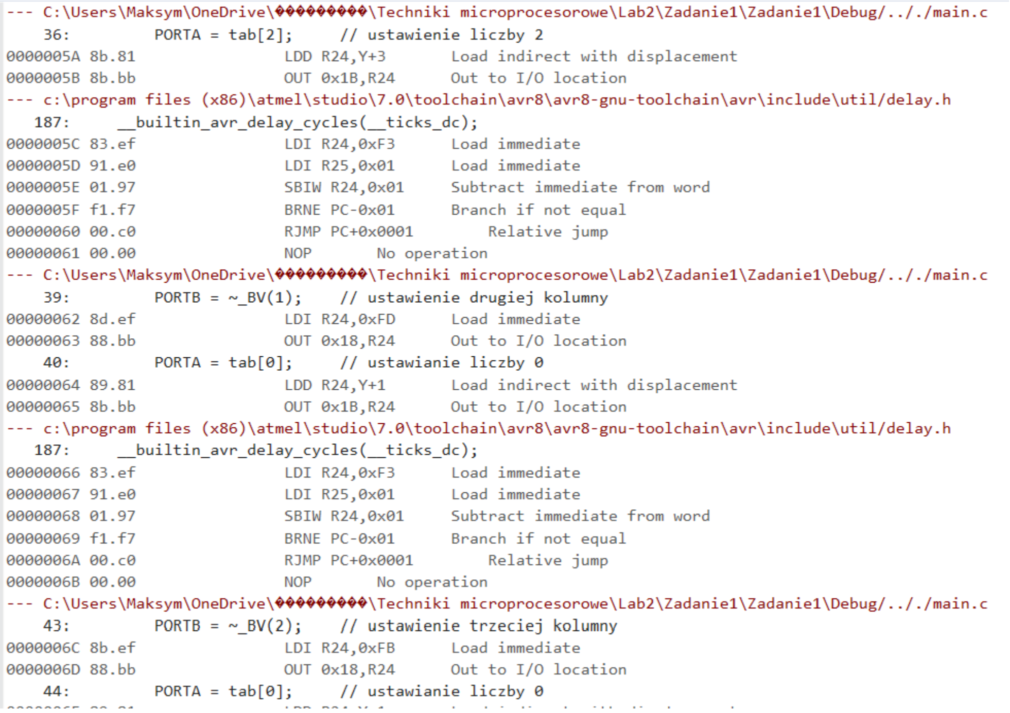
## Sprawozdanie z Laboratorium 2

## Maksym Pervov, grupa 4.7/13

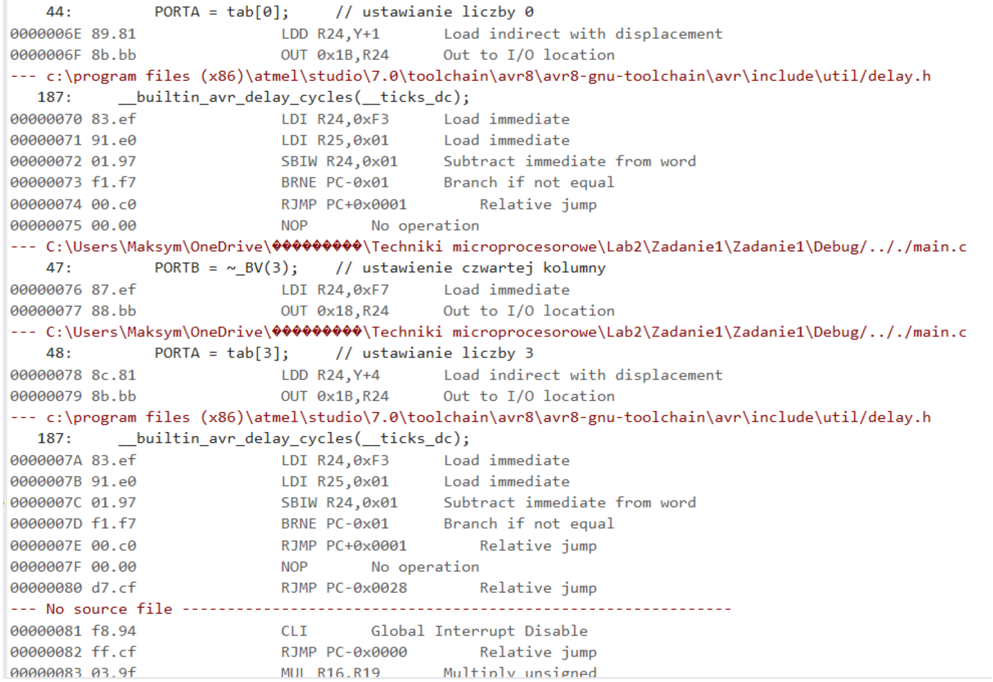
1. Zadanie 1
   Disassembly:



```
--- C:\Users\Maksym\OneDrive\���������\Techniki microprocesorowe\Lab2\Zadanie1\Zadanie1\Debug/.././main.c
    14: {
00000041 cf.93                PUSH R28      Push register on stack
00000042 df.93                PUSH R29      Push register on stack
00000043 cd.b7                IN R28,0x3D       In from I/O location
00000044 de.b7                IN R29,0x3E       In from I/O location
00000045 2a.97                SBIW R28,0x0A     Subtract immediate from word
00000046 0f.b6                IN R0,0x3F        In from I/O location
00000047 f8.94                CLI       Global Interrupt Disable
00000048 de.bf                OUT 0x3E,R29      Out to I/O location
00000049 0f.be                OUT 0x3F,R0       Out to I/O location
0000004A cd.bf                OUT 0x3D,R28      Out to I/O location
    15:     DDRA=0xFF;
0000004B 8f.ef                SER R24       Set Register
0000004C 8a.bb                OUT 0x1A,R24      Out to I/O location
    16:     DDRB=0x0F;
0000004D 8f.e0                LDI R24,0x0F      Load immediate
0000004E 87.bb                OUT 0x17,R24      Out to I/O location
    18:     unsigned char tab[10] =
0000004F 8a.e0                LDI R24,0x0A      Load immediate
00000050 e0.e6                LDI R30,0x60      Load immediate
00000051 f0.e0                LDI R31,0x00      Load immediate
00000052 de.01                MOVW R26,R28      Copy register pair
00000053 11.96                ADIW R26,0x01     Add immediate to word
00000054 01.90                LD R0,Z+      Load indirect and postincrement
00000055 0d.92                ST X+,R0      Store indirect and postincrement
00000056 8a.95                DEC R24       Decrement
00000057 e1.f7                BRNE PC-0x03      Branch if not equal
    35:         PORTB = ~_BV(0);    // ustawienie pierwszej kolumny
00000058 8e.ef                LDI R24,0xFE      Load immediate
00000059 88.bb                OUT 0x18,R24      Out to I/O location
```



```
--- C:\Users\Maksym\OneDrive\���������\Techniki microprocesorowe\Lab2\Zadanie1\Zadanie1\Debug/.././main.c
    36:         PORTA = tab[2];     // ustawienie liczby 2
0000005A 8b.81                LDD R24,Y+3       Load indirect with displacement
0000005B 8b.bb                OUT 0x1B,R24      Out to I/O location
--- c:\program files (x86)\atmel\studio\7.0\toolchain\avr8\avr8-gnu-toolchain\avr\include\util/delay.h
   187:     __builtin_avr_delay_cycles(__ticks_dc);
0000005C 83.ef                LDI R24,0xF3      Load immediate
0000005D 91.e0                LDI R25,0x01      Load immediate
0000005E 01.97                SBIW R24,0x01     Subtract immediate from word
0000005F f1.f7                BRNE PC-0x01      Branch if not equal
00000060 00.c0                RJMP PC+0x0001        Relative jump
00000061 00.00                NOP       No operation
--- C:\Users\Maksym\OneDrive\���������\Techniki microprocesorowe\Lab2\Zadanie1\Zadanie1\Debug/.././main.c
    39:         PORTB = ~_BV(1);    // ustawienie drugiej kolumny
00000062 8d.ef                LDI R24,0xFD      Load immediate
00000063 88.bb                OUT 0x18,R24      Out to I/O location
    40:         PORTA = tab[0];     // ustawianie liczby 0
00000064 89.81                LDD R24,Y+1       Load indirect with displacement
00000065 8b.bb                OUT 0x1B,R24      Out to I/O location
--- c:\program files (x86)\atmel\studio\7.0\toolchain\avr8\avr8-gnu-toolchain\avr\include\util/delay.h
   187:     __builtin_avr_delay_cycles(__ticks_dc);
00000066 83.ef                LDI R24,0xF3      Load immediate
00000067 91.e0                LDI R25,0x01      Load immediate
00000068 01.97                SBIW R24,0x01     Subtract immediate from word
00000069 f1.f7                BRNE PC-0x01      Branch if not equal
0000006A 00.c0                RJMP PC+0x0001        Relative jump
0000006B 00.00                NOP       No operation
--- C:\Users\Maksym\OneDrive\���������\Techniki microprocesorowe\Lab2\Zadanie1\Zadanie1\Debug/.././main.c
    43:         PORTB = ~_BV(2);    // ustawienie trzeciej kolumny
0000006C 8b.ef                LDI R24,0xFB      Load immediate
0000006D 88.bb                OUT 0x18,R24      Out to I/O location
    44:         PORTA = tab[0];     // ustawianie liczby 0
```



```
    44:          PORTA = tab[0];     // ustawianie liczby 0
0000006E 89.81                LDD R24,Y+1       Load indirect with displacement
0000006F 8b.bb                OUT 0x1B,R24      Out to I/O location
--- c:\program files (x86)\atmel\studio\7.0\toolchain\avr8\avr8-gnu-toolchain\avr\include\util/delay.h
   187:      __builtin_avr_delay_cycles(__ticks_dc);
00000070 83.ef                LDI R24,0xF3      Load immediate
00000071 91.e0                LDI R25,0x01      Load immediate
00000072 01.97                SBIW R24,0x01     Subtract immediate from word
00000073 f1.f7                BRNE PC-0x01      Branch if not equal
00000074 00.c0                RJMP PC+0x0001        Relative jump
00000075 00.00                NOP       No operation
--- C:\Users\Maksym\OneDrive\���������\Techniki microprocesorowe\Lab2\Zadanie1\Zadanie1\Debug/.././main.c
    47:          PORTB = ~_BV(3);    // ustawienie czwartej kolumny
00000076 87.ef                LDI R24,0xF7      Load immediate
00000077 88.bb                OUT 0x18,R24      Out to I/O location
--- C:\Users\Maksym\OneDrive\���������\Techniki microprocesorowe\Lab2\Zadanie1\Zadanie1\Debug/.././main.c
    48:          PORTA = tab[3];     // ustawianie liczby 3
00000078 8c.81                LDD R24,Y+4       Load indirect with displacement
00000079 8b.bb                OUT 0x1B,R24      Out to I/O location
--- c:\program files (x86)\atmel\studio\7.0\toolchain\avr8\avr8-gnu-toolchain\avr\include\util/delay.h
   187:      __builtin_avr_delay_cycles(__ticks_dc);
0000007A 83.ef                LDI R24,0xF3      Load immediate
0000007B 91.e0                LDI R25,0x01      Load immediate
0000007C 01.97                SBIW R24,0x01     Subtract immediate from word
0000007D f1.f7                BRNE PC-0x01      Branch if not equal
0000007E 00.c0                RJMP PC+0x0001        Relative jump
0000007F 00.00                NOP       No operation
00000080 d7.cf                RJMP PC-0x0028        Relative jump
--- No source file -------------------------------------------------------------
00000081 f8.94                CLI       Global Interrupt Disable
00000082 ff.cf                RJMP PC-0x0000        Relative jump
00000083 03.9f                MUL R16,R19       Multiply unsigned
```

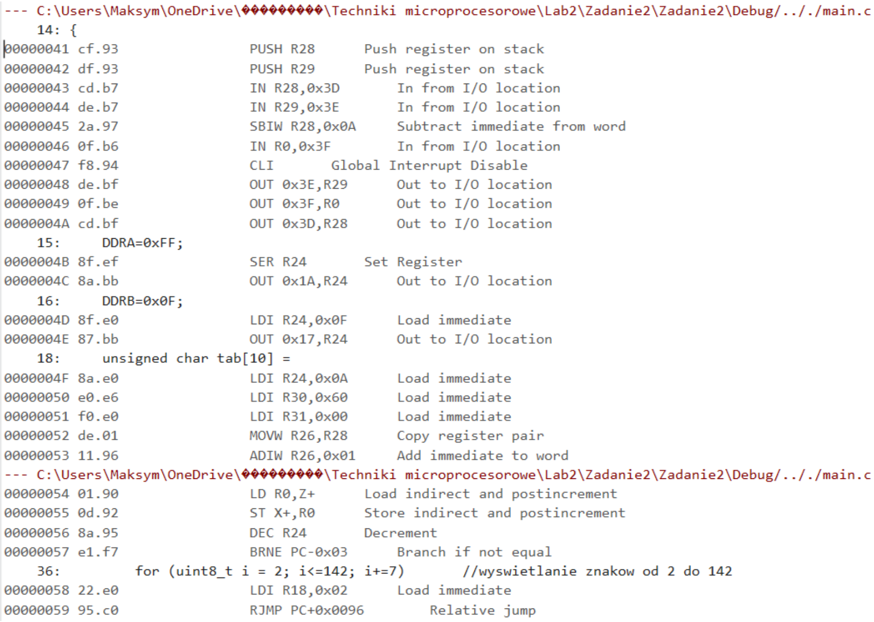
Source code:

```
/*
 * Zadanie1.c
 *
 * Created: 28.04.2022 12:25:47
 * Author : Student_PL
 */
#define F_CPU 1000000L
#include <avr/io.h>
#include <util/delay.h>
#include <stdlib.h>


int main(void)
{
    DDRA=0xFF;
       DDRB=0x0F;

       unsigned char tab[10] =
       {
              0b00000011,                 //wyswietlanie cyfry 0
              0b10011111,                 //wyswietlanie cyfry 1
              0b00100101,                 //wyswietlanie cyfry 2
              0b00001101,                 //wyswietlanie cyfry 3
              0b10011001,                 //wyswietlanie cyfry 4
              0b01001001,                 //wyswietlanie cyfry 5
              0b01000001,                 //wyswietlanie cyfry 6
              0b00011111,                 //wyswietlanie cyfry 7
              0b00000001,                 //wyswietlanie cyfry 8
              0b00001001                  //wyswietlanie cyfry 9
       };

    while (1)
    {
              //Rok urodzenia - 2003
              PORTB = ~_BV(0);     // ustawienie pierwszej kolumny
              PORTA = tab[2];      // ustawienie liczby 2
              _delay_ms(2);

              PORTB = ~_BV(1);     // ustawienie drugiej kolumny
```

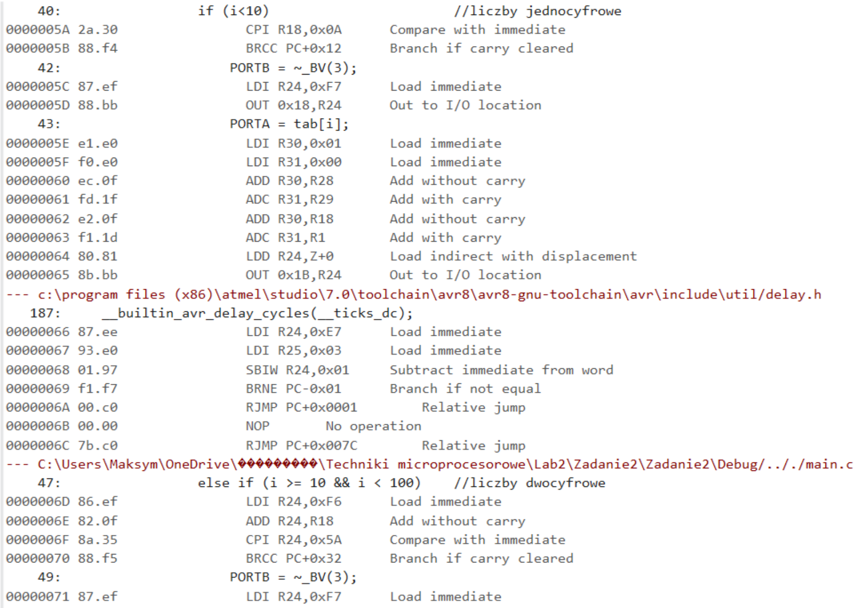
```
                PORTA = tab[0];     // ustawianie liczby 0
                _delay_ms(2);

                PORTB = ~_BV(2);    // ustawienie trzeciej kolumny
                PORTA = tab[0];     // ustawianie liczby 0
                _delay_ms(2);

                PORTB = ~_BV(3);    // ustawienie czwartej kolumny
                PORTA = tab[3];     // ustawianie liczby 3
                _delay_ms(2);
        }
            return 0;
    }
```

2. Zadanie 2

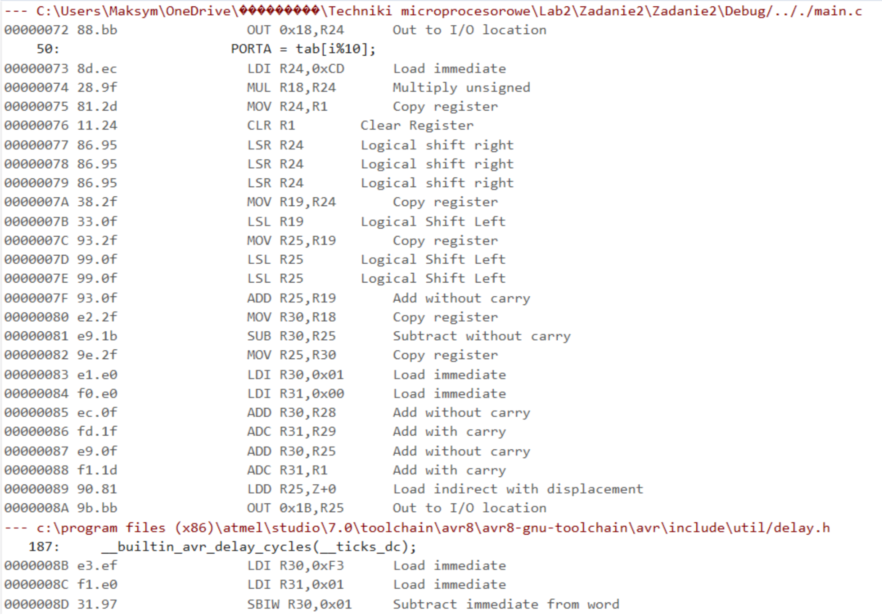
   Disassembly:



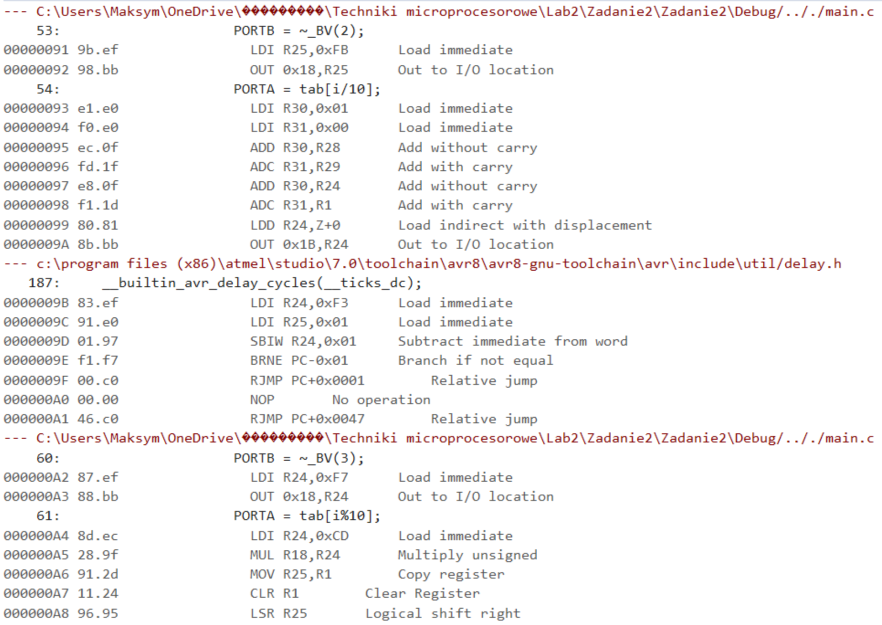
```
--- C:\Users\Maksym\OneDrive\���������\Techniki microprocesorowe\Lab2\Zadanie2\Zadanie2\Debug/.././main.c
    14: {
00000041 cf.93                PUSH R28      Push register on stack
00000042 df.93                PUSH R29      Push register on stack
00000043 cd.b7                IN R28,0x3D       In from I/O location
00000044 de.b7                IN R29,0x3E       In from I/O location
00000045 2a.97                SBIW R28,0x0A     Subtract immediate from word
00000046 0f.b6                IN R0,0x3F        In from I/O location
00000047 f8.94                CLI       Global Interrupt Disable
00000048 de.bf                OUT 0x3E,R29      Out to I/O location
00000049 0f.be                OUT 0x3F,R0       Out to I/O location
0000004A cd.bf                OUT 0x3D,R28      Out to I/O location
    15:     DDRA=0xFF;
0000004B 8f.ef                SER R24       Set Register
0000004C 8a.bb                OUT 0x1A,R24      Out to I/O location
    16:     DDRB=0x0F;
0000004D 8f.e0                LDI R24,0x0F      Load immediate
0000004E 87.bb                OUT 0x17,R24      Out to I/O location
    18:     unsigned char tab[10] =
0000004F 8a.e0                LDI R24,0x0A      Load immediate
00000050 e0.e6                LDI R30,0x60      Load immediate
00000051 f0.e0                LDI R31,0x00      Load immediate
00000052 de.01                MOVW R26,R28      Copy register pair
00000053 11.96                ADIW R26,0x01     Add immediate to word
--- C:\Users\Maksym\OneDrive\���������\Techniki microprocesorowe\Lab2\Zadanie2\Zadanie2\Debug/.././main.c
00000054 01.90                LD R0,Z+      Load indirect and postincrement
00000055 0d.92                ST X+,R0      Store indirect and postincrement
00000056 8a.95                DEC R24       Decrement
00000057 e1.f7                BRNE PC-0x03      Branch if not equal
    36:         for (uint8_t i = 2; i<=142; i+=7)       //wyswietlanie znakow od 2 do 142
00000058 22.e0                LDI R18,0x02      Load immediate
00000059 95.c0                RJMP PC+0x0096        Relative jump
```



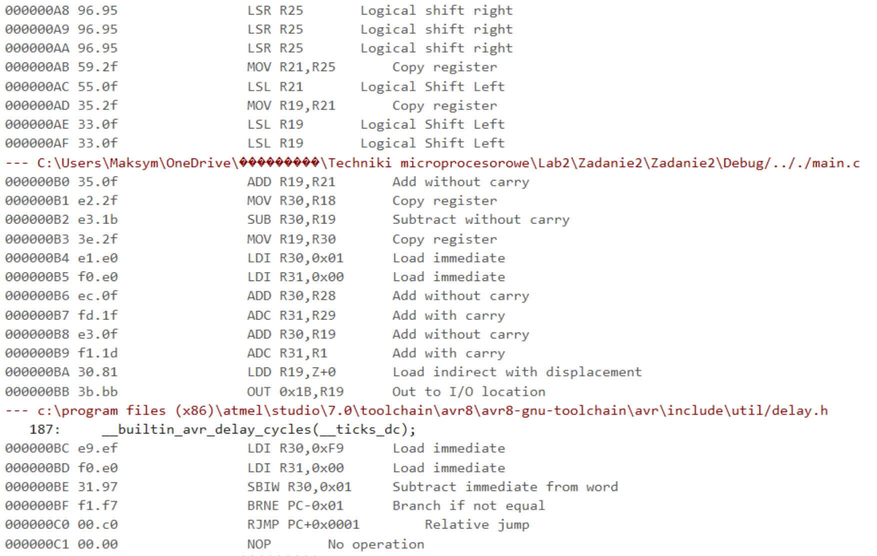
```
    40:                 if (i<10)                           //liczby jednocyfrowe
0000005A 2a.30                CPI R18,0x0A      Compare with immediate
0000005B 88.f4                BRCC PC+0x12      Branch if carry cleared
    42:                     PORTB = ~_BV(3);
0000005C 87.ef                LDI R24,0xF7      Load immediate
0000005D 88.bb                OUT 0x18,R24      Out to I/O location
    43:                     PORTA = tab[i];
0000005E e1.e0                LDI R30,0x01      Load immediate
0000005F f0.e0                LDI R31,0x00      Load immediate
00000060 ec.0f                ADD R30,R28       Add without carry
00000061 fd.1f                ADC R31,R29       Add with carry
00000062 e2.0f                ADD R30,R18       Add without carry
00000063 f1.1d                ADC R31,R1        Add with carry
00000064 80.81                LDD R24,Z+0       Load indirect with displacement
00000065 8b.bb                OUT 0x1B,R24      Out to I/O location
--- c:\program files (x86)\atmel\studio\7.0\toolchain\avr8\avr8-gnu-toolchain\avr\include\util/delay.h
   187:     __builtin_avr_delay_cycles(__ticks_dc);
00000066 87.ee                LDI R24,0xE7      Load immediate
00000067 93.e0                LDI R25,0x03      Load immediate
00000068 01.97                SBIW R24,0x01     Subtract immediate from word
00000069 f1.f7                BRNE PC-0x01      Branch if not equal
0000006A 00.c0                RJMP PC+0x0001        Relative jump
0000006B 00.00                NOP       No operation
0000006C 7b.c0                RJMP PC+0x007C        Relative jump
--- C:\Users\Maksym\OneDrive\���������\Techniki microprocesorowe\Lab2\Zadanie2\Zadanie2\Debug/.././main.c
    47:                 else if (i >= 10 && i < 100)    //liczby dwocyfrowe
0000006D 86.ef                LDI R24,0xF6      Load immediate
0000006E 82.0f                ADD R24,R18       Add without carry
0000006F 8a.35                CPI R24,0x5A      Compare with immediate
00000070 88.f5                BRCC PC+0x32      Branch if carry cleared
    49:                     PORTB = ~_BV(3);
00000071 87.ef                LDI R24,0xF7      Load immediate
```



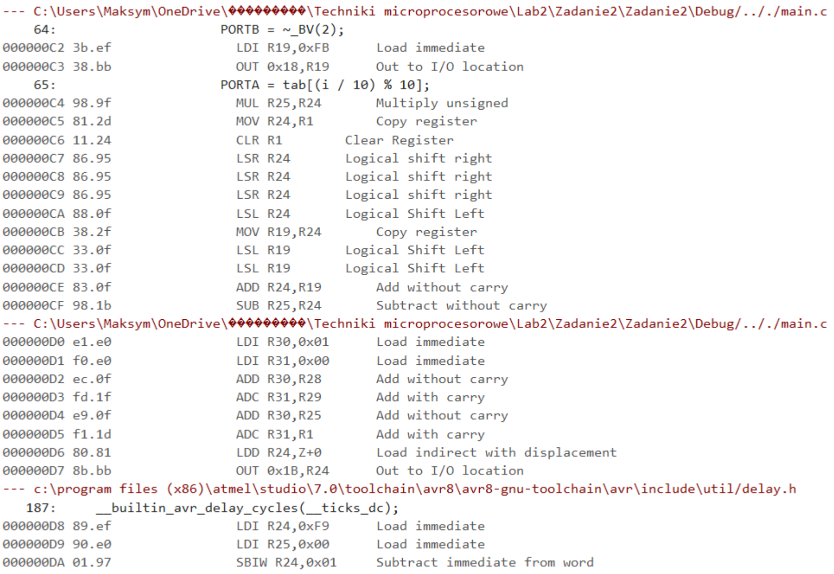
```
--- C:\Users\Maksym\OneDrive\���������\Techniki microprocesorowe\Lab2\Zadanie2\Zadanie2\Debug/.././main.c
00000072 88.bb                OUT 0x18,R24      Out to I/O location
    50:                     PORTA = tab[i%10];
00000073 8d.ec                LDI R24,0xCD      Load immediate
00000074 28.9f                MUL R18,R24       Multiply unsigned
00000075 81.2d                MOV R24,R1        Copy register
00000076 11.24                CLR R1        Clear Register
00000077 86.95                LSR R24       Logical shift right
00000078 86.95                LSR R24       Logical shift right
00000079 86.95                LSR R24       Logical shift right
0000007A 38.2f                MOV R19,R24       Copy register
0000007B 33.0f                LSL R19       Logical Shift Left
0000007C 93.2f                MOV R25,R19       Copy register
0000007D 99.0f                LSL R25       Logical Shift Left
0000007E 99.0f                LSL R25       Logical Shift Left
0000007F 93.0f                ADD R25,R19       Add without carry
00000080 e2.2f                MOV R30,R18       Copy register
00000081 e9.1b                SUB R30,R25       Subtract without carry
00000082 9e.2f                MOV R25,R30       Copy register
00000083 e1.e0                LDI R30,0x01      Load immediate
00000084 f0.e0                LDI R31,0x00      Load immediate
00000085 ec.0f                ADD R30,R28       Add without carry
00000086 fd.1f                ADC R31,R29       Add with carry
00000087 e9.0f                ADD R30,R25       Add without carry
00000088 f1.1d                ADC R31,R1        Add with carry
00000089 90.81                LDD R25,Z+0       Load indirect with displacement
0000008A 9b.bb                OUT 0x1B,R25      Out to I/O location
--- c:\program files (x86)\atmel\studio\7.0\toolchain\avr8\avr8-gnu-toolchain\avr\include\util/delay.h
   187:      __builtin_avr_delay_cycles(__ticks_dc);
0000008B e3.ef                LDI R30,0xF3      Load immediate
0000008C f1.e0                LDI R31,0x01      Load immediate
0000008D 31.97                SBIW R30,0x01     Subtract immediate from word
```



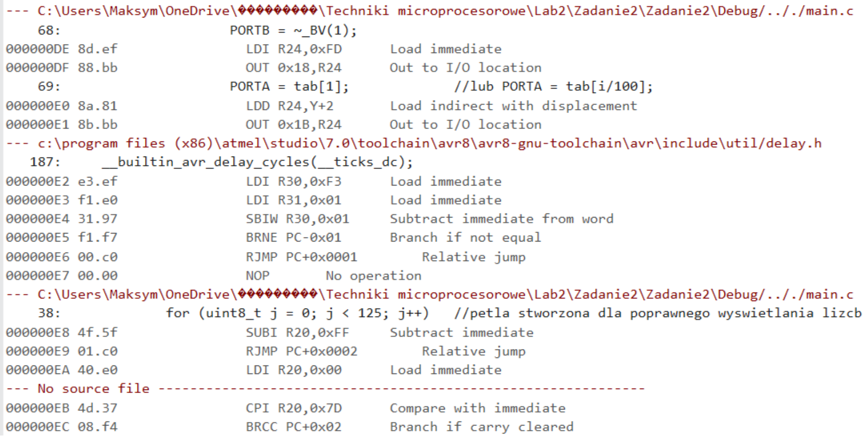
```
--- C:\Users\Maksym\OneDrive\���������\Techniki microprocesorowe\Lab2\Zadanie2\Zadanie2\Debug/.././main.c
    53:                     PORTB = ~_BV(2);
00000091 9b.ef                LDI R25,0xFB      Load immediate
00000092 98.bb                OUT 0x18,R25      Out to I/O location
    54:                     PORTA = tab[i/10];
00000093 e1.e0                LDI R30,0x01      Load immediate
00000094 f0.e0                LDI R31,0x00      Load immediate
00000095 ec.0f                ADD R30,R28       Add without carry
00000096 fd.1f                ADC R31,R29       Add with carry
00000097 e8.0f                ADD R30,R24       Add without carry
00000098 f1.1d                ADC R31,R1        Add with carry
00000099 80.81                LDD R24,Z+0       Load indirect with displacement
0000009A 8b.bb                OUT 0x1B,R24      Out to I/O location
--- c:\program files (x86)\atmel\studio\7.0\toolchain\avr8\avr8-gnu-toolchain\avr\include\util/delay.h
   187:      __builtin_avr_delay_cycles(__ticks_dc);
0000009B 83.ef                LDI R24,0xF3      Load immediate
0000009C 91.e0                LDI R25,0x01      Load immediate
0000009D 01.97                SBIW R24,0x01     Subtract immediate from word
0000009E f1.f7                BRNE PC-0x01      Branch if not equal
0000009F 00.c0                RJMP PC+0x0001        Relative jump
000000A0 00.00                NOP       No operation
000000A1 46.c0                RJMP PC+0x0047        Relative jump
--- C:\Users\Maksym\OneDrive\���������\Techniki microprocesorowe\Lab2\Zadanie2\Zadanie2\Debug/.././main.c
    60:                     PORTB = ~_BV(3);
000000A2 87.ef                LDI R24,0xF7      Load immediate
000000A3 88.bb                OUT 0x18,R24      Out to I/O location
    61:                     PORTA = tab[i%10];
000000A4 8d.ec                LDI R24,0xCD      Load immediate
000000A5 28.9f                MUL R18,R24       Multiply unsigned
000000A6 91.2d                MOV R25,R1        Copy register
000000A7 11.24                CLR R1        Clear Register
000000A8 96.95                LSR R25       Logical shift right
```



```
000000A8 96.95                LSR R25       Logical shift right
000000A9 96.95                LSR R25       Logical shift right
000000AA 96.95                LSR R25       Logical shift right
000000AB 59.2f                MOV R21,R25       Copy register
000000AC 55.0f                LSL R21       Logical Shift Left
000000AD 35.2f                MOV R19,R21       Copy register
000000AE 33.0f                LSL R19       Logical Shift Left
000000AF 33.0f                LSL R19       Logical Shift Left
--- C:\Users\Maksym\OneDrive\���������\Techniki microprocesorowe\Lab2\Zadanie2\Zadanie2\Debug/.././main.c
000000B0 35.0f                ADD R19,R21       Add without carry
000000B1 e2.2f                MOV R30,R18       Copy register
000000B2 e3.1b                SUB R30,R19       Subtract without carry
000000B3 3e.2f                MOV R19,R30       Copy register
000000B4 e1.e0                LDI R30,0x01      Load immediate
000000B5 f0.e0                LDI R31,0x00      Load immediate
000000B6 ec.0f                ADD R30,R28       Add without carry
000000B7 fd.1f                ADC R31,R29       Add with carry
000000B8 e3.0f                ADD R30,R19       Add without carry
000000B9 f1.1d                ADC R31,R1        Add with carry
000000BA 30.81                LDD R19,Z+0       Load indirect with displacement
000000BB 3b.bb                OUT 0x1B,R19      Out to I/O location
--- c:\program files (x86)\atmel\studio\7.0\toolchain\avr8\avr8-gnu-toolchain\avr\include\util/delay.h
   187:      __builtin_avr_delay_cycles(__ticks_dc);
000000BC e9.ef                LDI R30,0xF9      Load immediate
000000BD f0.e0                LDI R31,0x00      Load immediate
000000BE 31.97                SBIW R30,0x01     Subtract immediate from word
000000BF f1.f7                BRNE PC-0x01      Branch if not equal
000000C0 00.c0                RJMP PC+0x0001        Relative jump
000000C1 00.00                NOP       No operation
```



```
--- C:\Users\Maksym\OneDrive\���������\Techniki microprocesorowe\Lab2\Zadanie2\Zadanie2\Debug/../.\/main.c
    64:                         PORTB = ~_BV(2);
000000C2 3b.ef                    LDI R19,0xFB      Load immediate
000000C3 38.bb                    OUT 0x18,R19      Out to I/O location
    65:                         PORTA = tab[(i / 10) % 10];
000000C4 98.9f                    MUL R25,R24       Multiply unsigned
000000C5 81.2d                    MOV R24,R1        Copy register
000000C6 11.24                    CLR R1        Clear Register
000000C7 86.95                    LSR R24       Logical shift right
000000C8 86.95                    LSR R24       Logical shift right
000000C9 86.95                    LSR R24       Logical shift right
000000CA 88.0f                    LSL R24       Logical Shift Left
000000CB 38.2f                    MOV R19,R24       Copy register
000000CC 33.0f                    LSL R19       Logical Shift Left
000000CD 33.0f                    LSL R19       Logical Shift Left
000000CE 83.0f                    ADD R24,R19       Add without carry
000000CF 98.1b                    SUB R25,R24       Subtract without carry
--- C:\Users\Maksym\OneDrive\���������\Techniki microprocesorowe\Lab2\Zadanie2\Zadanie2\Debug/.././main.c
000000D0 e1.e0                    LDI R30,0x01      Load immediate
000000D1 f0.e0                    LDI R31,0x00      Load immediate
000000D2 ec.0f                    ADD R30,R28       Add without carry
000000D3 fd.1f                    ADC R31,R29       Add with carry
000000D4 e9.0f                    ADD R30,R25       Add without carry
000000D5 f1.1d                    ADC R31,R1        Add with carry
000000D6 80.81                    LDD R24,Z+0       Load indirect with displacement
000000D7 8b.bb                    OUT 0x1B,R24      Out to I/O location
--- c:\program files (x86)\atmel\studio\7.0\toolchain\avr8\avr8-gnu-toolchain\avr\include\util/delay.h
   187:      __builtin_avr_delay_cycles(__ticks_dc);
000000D8 89.ef                    LDI R24,0xF9      Load immediate
000000D9 90.e0                    LDI R25,0x00      Load immediate
000000DA 01.97                    SBIW R24,0x01     Subtract immediate from word
```



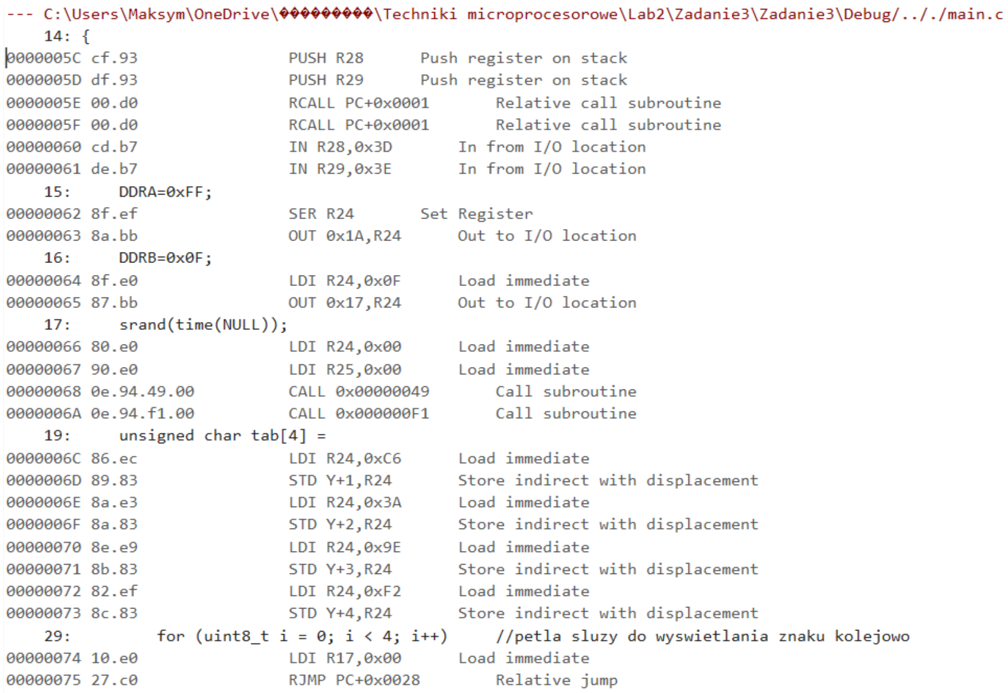
```
--- C:\Users\Maksym\OneDrive\���������\Techniki microprocesorowe\Lab2\Zadanie2\Zadanie2\Debug/.././main.c
    68:                         PORTB = ~_BV(1);
000000DE 8d.ef                    LDI R24,0xFD      Load immediate
000000DF 88.bb                    OUT 0x18,R24      Out to I/O location
    69:                         PORTA = tab[1];             //lub PORTA = tab[i/100];
000000E0 8a.81                    LDD R24,Y+2       Load indirect with displacement
000000E1 8b.bb                    OUT 0x1B,R24      Out to I/O location
--- c:\program files (x86)\atmel\studio\7.0\toolchain\avr8\avr8-gnu-toolchain\avr\include\util/delay.h
   187:      __builtin_avr_delay_cycles(__ticks_dc);
000000E2 e3.ef                    LDI R30,0xF3      Load immediate
000000E3 f1.e0                    LDI R31,0x01      Load immediate
000000E4 31.97                    SBIW R30,0x01     Subtract immediate from word
000000E5 f1.f7                    BRNE PC-0x01      Branch if not equal
000000E6 00.c0                    RJMP PC+0x0001        Relative jump
000000E7 00.00                    NOP       No operation
--- C:\Users\Maksym\OneDrive\���������\Techniki microprocesorowe\Lab2\Zadanie2\Zadanie2\Debug/.././main.c
    38:                 for (uint8_t j = 0; j < 125; j++)   //petla stworzona dla poprawnego wyswietlania lizcb
000000E8 4f.5f                    SUBI R20,0xFF     Subtract immediate
000000E9 01.c0                    RJMP PC+0x0002        Relative jump
000000EA 40.e0                    LDI R20,0x00      Load immediate
--- No source file -------------------------------------------------------------
000000EB 4d.37                    CPI R20,0x7D      Compare with immediate
000000EC 08.f4                    BRCC PC+0x02      Branch if carry cleared
```

Source code:

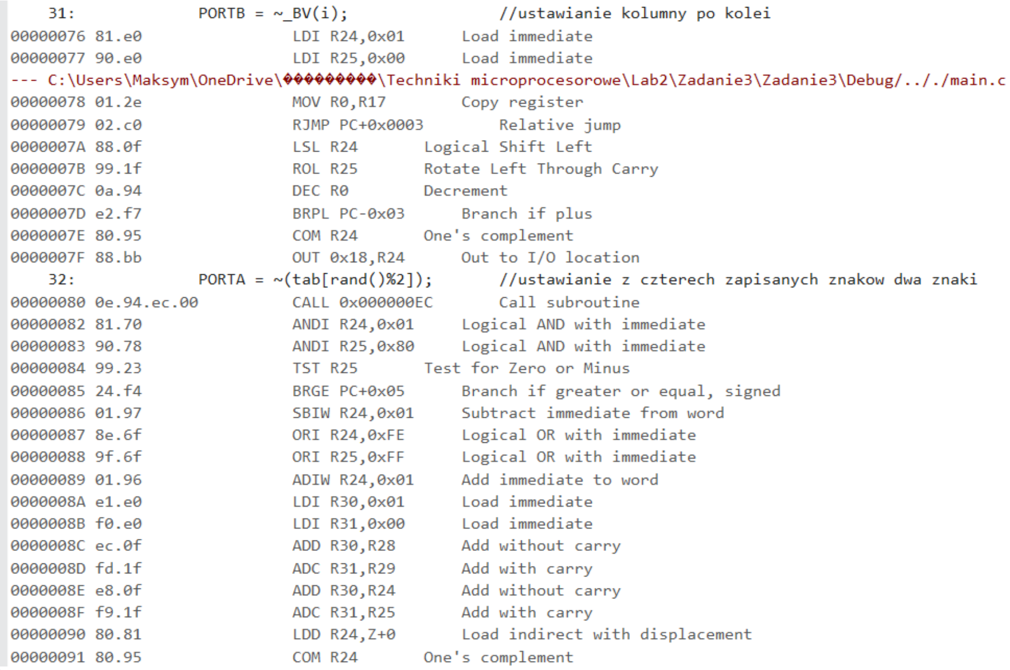
```
/*
 * Zadanie2.c
 *
 * Created: 28.04.2022 12:58:22
 * Author : Student_PL
 */
#define F_CPU 1000000L
#include <avr/io.h>
#include <util/delay.h>
#include <stdlib.h>


int main(void)
{
    DDRA=0xFF;
       DDRB=0x0F;

       unsigned char tab[10] =
       {
              0b00000011,                //wyswietlanie cyfry 0
              0b10011111,                //wyswietlanie cyfry 1
              0b00100101,                //wyswietlanie cyfry 2
              0b00001101,                //wyswietlanie cyfry 3
              0b10011001,                //wyswietlanie cyfry 4
              0b01001001,                //wyswietlanie cyfry 5
              0b01000001,                //wyswietlanie cyfry 6
              0b00011111,                //wyswietlanie cyfry 7
              0b00000001,                //wyswietlanie cyfry 8
              0b00001001                   //wyswietlanie cyfry 9
       };

       //int czas;

    while (1)
    {
              for (uint8_t i = 2; i<=142; i+=7)       //wyswietlanie znakow od 2 do 142
              {
                     for (uint8_t j = 0; j < 125; j++)       //petla stworzona dla
                                                              poprawnego wyswietlania
                                                              lizcb
                     {
                            if (i<10)                         //liczby jednocyfrowe
                            {
                                   PORTB = ~_BV(3);
                                   PORTA = tab[i];
                                   _delay_ms(4);
                            }

                            else if (i >= 10 && i < 100)      //liczby dwocyfrowe
                            {
                                   PORTB = ~_BV(3);
                                   PORTA = tab[i%10];
                                   _delay_ms(2);

                                   PORTB = ~_BV(2);
                                   PORTA = tab[i/10];
                                   _delay_ms(2);
                            }

                            else                              //liczby trzycyfrowe
                            {
                                   PORTB = ~_BV(3);
                                   PORTA = tab[i%10];
                                   _delay_ms(1);

                                   PORTB = ~_BV(2);
                                   PORTA = tab[(i / 10) % 10];
                                   _delay_ms(1);

                                   PORTB = ~_BV(1);
                                   PORTA = tab[1];            //lub PORTA = tab[i/100];
                                   _delay_ms(2);
                            }
                     }
              }
    }
       return 0;
}
```

3. Zadanie 3

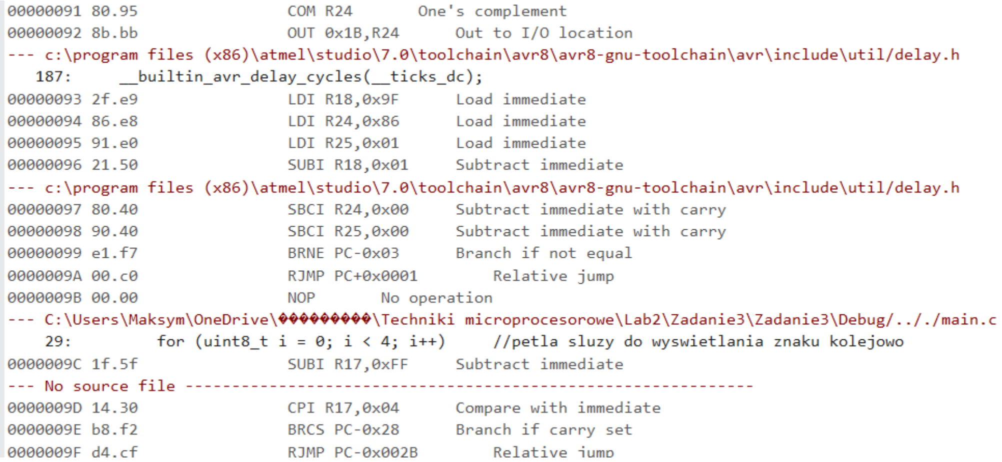
   Disassembly:



```
--- C:\Users\Maksym\OneDrive\���������\Techniki microprocesorowe\Lab2\Zadanie3\Zadanie3\Debug/.././main.c
    14: {
0000005C cf.93                PUSH R28       Push register on stack
0000005D df.93                PUSH R29       Push register on stack
0000005E 00.d0                RCALL PC+0x0001       Relative call subroutine
0000005F 00.d0                RCALL PC+0x0001       Relative call subroutine
00000060 cd.b7                IN R28,0x3D       In from I/O location
00000061 de.b7                IN R29,0x3E       In from I/O location
    15:      DDRA=0xFF;
00000062 8f.ef                SER R24        Set Register
00000063 8a.bb                OUT 0x1A,R24      Out to I/O location
    16:      DDRB=0x0F;
00000064 8f.e0                LDI R24,0x0F      Load immediate
00000065 87.bb                OUT 0x17,R24      Out to I/O location
    17:      srand(time(NULL));
00000066 80.e0                LDI R24,0x00      Load immediate
00000067 90.e0                LDI R25,0x00      Load immediate
00000068 0e.94.49.00          CALL 0x00000049       Call subroutine
0000006A 0e.94.f1.00          CALL 0x000000F1       Call subroutine
    19:      unsigned char tab[4] =
0000006C 86.ec                LDI R24,0xC6      Load immediate
0000006D 89.83                STD Y+1,R24       Store indirect with displacement
0000006E 8a.e3                LDI R24,0x3A      Load immediate
0000006F 8a.83                STD Y+2,R24       Store indirect with displacement
00000070 8e.e9                LDI R24,0x9E      Load immediate
00000071 8b.83                STD Y+3,R24       Store indirect with displacement
00000072 82.ef                LDI R24,0xF2      Load immediate
00000073 8c.83                STD Y+4,R24       Store indirect with displacement
    29:         for (uint8_t i = 0; i < 4; i++)      //petla sluzy do wyswietlania znaku kolejowo
00000074 10.e0                LDI R17,0x00      Load immediate
00000075 27.c0                RJMP PC+0x0028        Relative jump
```



```
    31:            PORTB = ~_BV(i);            //ustawianie kolumny po kolei
00000076 81.e0                LDI R24,0x01      Load immediate
00000077 90.e0                LDI R25,0x00      Load immediate
--- C:\Users\Maksym\OneDrive\���������\Techniki microprocesorowe\Lab2\Zadanie3\Zadanie3\Debug/.././main.c
00000078 01.2e                MOV R0,R17        Copy register
00000079 02.c0                RJMP PC+0x0003        Relative jump
0000007A 88.0f                LSL R24        Logical Shift Left
0000007B 99.1f                ROL R25        Rotate Left Through Carry
0000007C 0a.94                DEC R0         Decrement
0000007D e2.f7                BRPL PC-0x03      Branch if plus
0000007E 80.95                COM R24        One's complement
0000007F 88.bb                OUT 0x18,R24      Out to I/O location
    32:            PORTA = ~(tab[rand()%2]);       //ustawianie z czterech zapisanych znakow dwa znaki
00000080 0e.94.ec.00          CALL 0x000000EC       Call subroutine
00000082 81.70                ANDI R24,0x01     Logical AND with immediate
00000083 90.78                ANDI R25,0x80     Logical AND with immediate
00000084 99.23                TST R25        Test for Zero or Minus
00000085 24.f4                BRGE PC+0x05      Branch if greater or equal, signed
00000086 01.97                SBIW R24,0x01     Subtract immediate from word
00000087 8e.6f                ORI R24,0xFE      Logical OR with immediate
00000088 9f.6f                ORI R25,0xFF      Logical OR with immediate
00000089 01.96                ADIW R24,0x01     Add immediate to word
0000008A e1.e0                LDI R30,0x01      Load immediate
0000008B f0.e0                LDI R31,0x00      Load immediate
0000008C ec.0f                ADD R30,R28       Add without carry
0000008D fd.1f                ADC R31,R29       Add with carry
0000008E e8.0f                ADD R30,R24       Add without carry
0000008F f9.1f                ADC R31,R25       Add with carry
00000090 80.81                LDD R24,Z+0       Load indirect with displacement
00000091 80.95                COM R24        One's complement
```



```
00000091 80.95                COM R24        One's complement
00000092 8b.bb                OUT 0x1B,R24      Out to I/O location
--- c:\program files (x86)\atmel\studio\7.0\toolchain\avr8\avr8-gnu-toolchain\avr\include\util/delay.h
   187:      __builtin_avr_delay_cycles(__ticks_dc);
00000093 2f.e9                LDI R18,0x9F      Load immediate
00000094 86.e8                LDI R24,0x86      Load immediate
00000095 91.e0                LDI R25,0x01      Load immediate
00000096 21.50                SUBI R18,0x01     Subtract immediate
--- c:\program files (x86)\atmel\studio\7.0\toolchain\avr8\avr8-gnu-toolchain\avr\include\util/delay.h
00000097 80.40                SBCI R24,0x00     Subtract immediate with carry
00000098 90.40                SBCI R25,0x00     Subtract immediate with carry
00000099 e1.f7                BRNE PC-0x03      Branch if not equal
0000009A 00.c0                RJMP PC+0x0001        Relative jump
0000009B 00.00                NOP        No operation
--- C:\Users\Maksym\OneDrive\���������\Techniki microprocesorowe\Lab2\Zadanie3\Zadanie3\Debug/.././main.c
    29:         for (uint8_t i = 0; i < 4; i++)      //petla sluzy do wyswietlania znaku kolejowo
0000009C 1f.5f                SUBI R17,0xFF     Subtract immediate
--- No source file -------------------------------------------------------------
0000009D 14.30                CPI R17,0x04      Compare with immediate
0000009E b8.f2                BRCS PC-0x28      Branch if carry set
0000009F d4.cf                RJMP PC-0x002B        Relative jump
```

Source code:

```
/*
 * Zadanie3.c
 *
 * Created: 28.04.2022 13:49:08
 * Author : Student_PL
 */
#define F_CPU 1000000L
#include <avr/io.h>
#include <util/delay.h>
#include <stdlib.h>


int main(void)
{
    DDRA=0xFF;
       DDRB=0x0F;
       srand(time(NULL));

       unsigned char tab[4] =
       {
              0b11000110,                 //wyswietlanie kwadratu na gorze
              0b00111010,                 //wyswietlanie kwadratu na dolu
              0b10011110,                 //wyswietlanie litery E
              0b11110010                  //wyswietlanie cyfry 3
       };

    while (1)
    {
              for (uint8_t i = 0; i < 4; i++)          //petla sluzy do wyswietlania znaku kolejowo
              {
                     PORTB = ~_BV(i);                  //ustawianie kolumny po kolei
                     PORTA = ~(tab[rand()%2]);         //ustawianie z czterech zapisanych znakow dwa znaki

                     _delay_ms(500);
              }
    }
       return 0;
}
```