

Факультет программной инженерии и компьютерной техники

Архитектура компьютера

Лабораторная работа №7

Вариант 4

Преподаватель: Тропченко Андрей Александрович

Выполнили: Кульбако Артемий Юрьевич Р33112

# Задание

Построить таблицу кодов сканирования.

Разработать программу ввода чисел для заданной раскладки.

**1 2 3 4 5**

**6 7 8 9 0**

**-, esc**

# Исходные текста программ

## С51

# include <reg51.h>

void Int00(void) interrupt 0 {

    int i = 0;

    float n;

    char digit;

    char xdata str[8];

    char m = 1;

    int mm;

    char x = P3;

    switch (x) {

        case 0xfe: digit = '1'; break;

        case 0x02: digit = '2'; break;

        case 0xfd: digit = '3'; break;

        case 0x79: digit = '4'; break;

        case 0xfb: digit = '5'; break;

        case 0x08: digit = '6'; break;

        case 0xf7: digit = '7'; break;

        case 0x10: digit = '8'; break;

        case 0xef: digit = '9'; break;

        case 0x01: digit = '0'; break;

        case 0xfa: digit = '-'; break;

        case 0xbe: digit = ','; break;

        case 0xfc: digit = 'e'; break;

        default: digit = 0xff;

    }

    if (digit == ',' || digit == '-') {

        mm = m;

        str[i++] = digit;

    }

    if (digit == 'e') {

        digit = 0xff;

        n /= mm;

    }

    if (digit != 0xff) {

        str[i++] = digit;

        m \*= 10;

        n = n \* 10 + (digit & 0xff);

    }

    while (~INT0);

}

int main() {

    EX0 = 1;

    IT0 = 1;

    EA = 1;

    while(1);

    return 0;

}

## A51

PUSH ACC

PUSH B

PUSH DPH

PUSH DPL

PUSH PSW

MOV PSW,#00H

PUSH AR0

PUSH AR1

PUSH AR2

PUSH AR3

PUSH AR4

PUSH AR5

PUSH AR6

PUSH AR7

MOV i,#00H

MOV i+01H,#00H

MOV m,#01H

MOV R7,P3

MOV A,R7

LCALL ?C?CCASE

DW ?C0011

DB 01H

DW ?C0003

DB 02H

DW ?C0007

DB 08H

DW ?C0009

DB 010H

DW ?C0005

DB 079H

DW ?C0013

DB 0BEH

DW ?C0010

DB 0EFH

DW ?C0008

DB 0F7H

DW ?C0012

DB 0FAH

DW ?C0006

DB 0FBH

DW ?C0014

DB 0FCH

DW ?C0004

DB 0FDH

DW ?C0002

DB 0FEH

DW 00H

DW ?C0015

?C0002:

MOV digit,#031H

SJMP ?C0001

?C0003:

MOV digit,#032H

SJMP ?C0001

?C0004:

MOV digit,#033H

SJMP ?C0001

?C0005:

MOV digit,#034H

SJMP ?C0001

?C0006:

MOV digit,#035H

SJMP ?C0001

?C0007:

MOV digit,#036H

SJMP ?C0001

?C0008:

MOV digit,#037H

SJMP ?C0001

?C0009:

MOV digit,#038H

SJMP ?C0001

?C0010:

MOV digit,#039H

SJMP ?C0001

?C0011:

MOV digit,#030H

SJMP ?C0001

?C0012:

MOV digit,#02DH

SJMP ?C0001

?C0013:

MOV digit,#02CH

SJMP ?C0001

?C0014:

MOV digit,#065H

SJMP ?C0001

?C0015:

MOV digit,#0FFH

?C0001:

MOV A,digit

XRL A,#02CH

JZ ?C0017

MOV A,digit

CJNE A,#02DH,?C0016

?C0017:

MOV R7,m

MOV A,R7

RLC A

SUBB A,ACC

MOV mm,A

MOV mm+01H,R7

INC i+01H

MOV A,i+01H

MOV R6,i

JNZ ?C0026

INC i

?C0026:

DEC A

ADD A,#LOW str

MOV DPL,A

MOV A,#HIGH str

ADDC A,R6

MOV DPH,A

MOV A,digit

MOVX @DPTR,A

?C0016:

MOV A,digit

CJNE A,#065H,?C0018

MOV digit,#0FFH

MOV R7,n+03H

MOV R6,n+02H

MOV R5,n+01H

MOV R4,n

PUSH AR4

PUSH AR5

PUSH AR6

PUSH AR7

MOV R4,mm

MOV R5,mm+01H

MOV A,R4

LCALL ?C?FCASTI

MOV R0,AR4

MOV R1,AR5

MOV R2,AR6

MOV R3,AR7

POP AR7

POP AR6

POP AR5

POP AR4

LCALL ?C?FPDIV

MOV n+03H,R7

MOV n+02H,R6

MOV n+01H,R5

MOV n,R4

?C0018:

MOV R7,digit

MOV A,R7

RLC A

SUBB A,ACC

MOV R6,A

SMOV A,R7

CPL A

ORL A,R6

JZ ?C0020

INC i+01H

MOV A,i+01H

MOV R6,i

JNZ ?C0027

INC i

C0027:

DEC A

ADD A,#LOW str

MOV DPL,A

MOV A,#HIGH str

ADDC A,R6

MOV DPH,A

MOV A,digit

MOVX @DPTR,A

MOV A,m

MOV B,#0AH

MUL AB

MOV m,A

CLR A

MOV R7,A

MOV R6,A

MOV R5,#020H

MOV R4,#041H

MOV R3,n+03H

MOV R2,n+02H

MOV R1,n+01H

MOV R0,n

LCALL ?C?FPMUL

PUSH AR4

PUSH AR5

PUSH AR6

PUSH AR7

MOV R4,digit

CLR A

LCALL ?C?FCASTC

POP AR3

POP AR2

POP AR1

POP AR0

LCALL ?C?FPADD

MOV n+03H,R7

MOV n+02H,R6

MOV n+01H,R5

MOV n,R4

?C0020:

MOV C,INT0

CPL C

JC ?C0020

POP AR7

POP AR6

POP AR5

POP AR4

POP AR3

POP AR2

POP AR1

POP AR0

POP PSW

POP DPL

POP DPH

POP B

POP ACC

RETI

SETB EX0

SETB IT0

SETB EA

?C0023:

SJMP ?C0023

# HEX

:03000300020B5A93

:100B5A00C0E0C0F0C083C082C0D075D000C000C061

:0D0B6A0001C002C003C004C005C006C007E2

:100B7700750800750900750F01AFB0EF120B340B44

:100B8700DE010BB6020BCA080BD4100BC0790BE8B9

:100B9700BE0BD9EF0BCFF70BE3FA0BC5FB0BEDFC45

:100BA7000BBBFD0BB1FE00000BF2750E31803F75DC

:100BB7000E32803A750E338035750E348030750EDF

:100BC70035802B750E368026750E378021750E38C9

:100BD700801C750E398017750E308012750E2D80AA

:100BE7000D750E2C8008750E658003750EFFE50EDA

:100BF700642C6005E50EB42D21AF0FEF3395E0F5BA

:100C0700108F110509E509AE0870020508142400C4

:100C1700F58274003EF583E50EF0E50EB465367592

:100C27000EFFAF0DAE0CAD0BAC0AC004C005C0067D

:100C3700C007AC10AD11EC120AA3A804A905AA06B7

:100C4700AB07D007D006D005D004120A018F0D8E4E

:100C57000C8D0B8C0AAF0EEF3395E0FEEFF44E6070

:100C6700520509E509AE0870020508142400F5824B

:100C770074003EF583E50EF0E50F75F00AA4F50F55

:100C8700E4FFFE7D207C41AB0DAA0CA90BA80A123C

:100C970008F8C004C005C006C007AC0EE4120AA8D5

:100CA700D003D002D001D0001208078F0D8E0C8D13

:080CB7000B8C0AA2B2B340FB52

:100CBF00D007D006D005D004D003D002D001D00089

:0B0CCF00D0D0D082D083D0F0D0E03233

:080CE600D2A8D288D2AF80FE33

:10080000020B11E86480F8E933E83360110460F00A

:10081000ED33EC337009E8FCE9FDEAFEEBFF22045E

:1008200060DED3EB9FEA9EE99DE8C2E78CF0C2F759

:1008300095F0400CE8CCF8E9CDF9EACEFAEBCFFB25

:10084000120ADC85D0F05804700320D5B3E8047098

:10085000075002B2D5020B1B92D5EC0460F7E4CC32

:10086000C0E0C398F8603B94186008400DD0E0FBEE

:10087000020AF3E4FBFAC9FC8028E830E406E4C984

:10088000FBE4CAFCE830E305E4C9CACBFCE8540742

:100890006010F8C3E913F9EA13FAEB13FBEC13FC4D

:1008A000D8F130F52FC3E49CFCEF9BFFEE9AFEEDF0

:1008B00099FDD0E0FBEF4E4D4C701222DB03020B92

:1008C00018EC2CFCEF33FFEE33FEED33FDED30E79B

:1008D000EB020AF3EF2BFFEE3AFEED39FDD0E0FB21

:1008E00050130BBB0003020B1BED13FDEE13FEEFC9

:0808F00013FFEC13FC020AF3F4

:1008F800EC4D6011E8497017ED33EC3304600DE4FA

:10090800FCFFFEFD22E933E8330470F8020B1112F4

:100918000ADC58046009E4CC2481500628500902F6

:100928000B1B284003020B18C0E0EB4A7044B98047

:1009380006D0E0FB020B07EF4E701CBD8008EBFFF2

:10094800EAFEE9FD80EBE98DF0A4FEE5F00209E19D

:10095800E9CDF9EAFEEBFFEF89F0A4FCE5F0CE89DA

:10096800F0A42EFFE435F0CD89F0A42DFEE435F097

:100978008067EF4E7005BD80D780C3EF8BF0A4ACC5

:10098800F0EE8BF0A42CFCE435F0F8EF8AF0A42C00

:10099800E5F038FCE433CB8DF0A42CFCE5F03BF813

:1009A800EE8AF0A42CFCE5F038F8E433CF89F0A403

:1009B8002CFCE5F038CF3400CE89F0A42FFFE5F009

:1009C8003EFEE433C98DF0A42EFEE5F039CD8AF061

:1009D800A42FFFE5F03EFEE43DFD33D0E0FB5007D9

:1009E8000BBB000F020B1BEC2CFCEF33FFEE33FEAE

:0609F800ED33FD020AF3DD

:1009FE00020B1BEC5D046005E859047003020B1139

:100A0E00120ADC580460F6EC4860F2EC7004FDFE4D

:100A1E00FF22C860DB2481C85009C39860025006CB

:100A2E00020B189850CAF582E9294B4A7005AB8221

:100A3E00020B0775F0007C1A7880C3EF9BEE9AEDDF

:100A4E0099400DC3EF9BFFEE9AFEED99FDE842F043

:100A5E00DC23ACF0D0E0FFD0E0FED0E0FDAB822096

:100A6E00E7101BEB60BAEC2CFCEF33FFEE33FEED20

:100A7E0033FD020AF3E803F830E705C0F075F00025

:100A8E00EF2FFFEE33FEED33FD40B830E7C280AA04

:100A9E0075F020800E75F010800575F0087D007ED3

:100AAE00007F003392D530D503120B26EC33401065

:100ABE00EF33FFEE33FEED33FDEC33FCD5F0ED22DC

:0E0ACE00E5F0247EA2D513CC92E7CDCEFF2218

:100ADC00E9D2E7C933E833F892D5EDD2E7CD33EC60

:070AEC0033FC5002B2D522D9

:100AF300EC30E7100FBF000C0EBE00080DBD000464

:100B03000BEB6014A2D5EB13FCED92E7FD2274FF0F

:100B1300FCFDFEFF22E480F8A2D574FF13FC7D8068

:030B2300E480EF7C

:03000000020CDA15

:0C0CDA00787FE4F6D8FD758111020CE66D

:0E0B2600C3E49FFFE49EFEE49DFDE49CFC22E0

:100B3400D083D082F8E4937012740193700DA3A350

:100B440093F8740193F5828883E473740293686064

:060B5400EFA3A3A380DF64

:00000001FF

# Выводы

Лабораторная на практическом примере показала мне как микроконтроллеры обрабатывают и интерпретируют сигналы прерывания с устройства ввода-вывода - клавиатура.