***МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ***

***Государственное образовательное учреждение высшего профессионального образования***

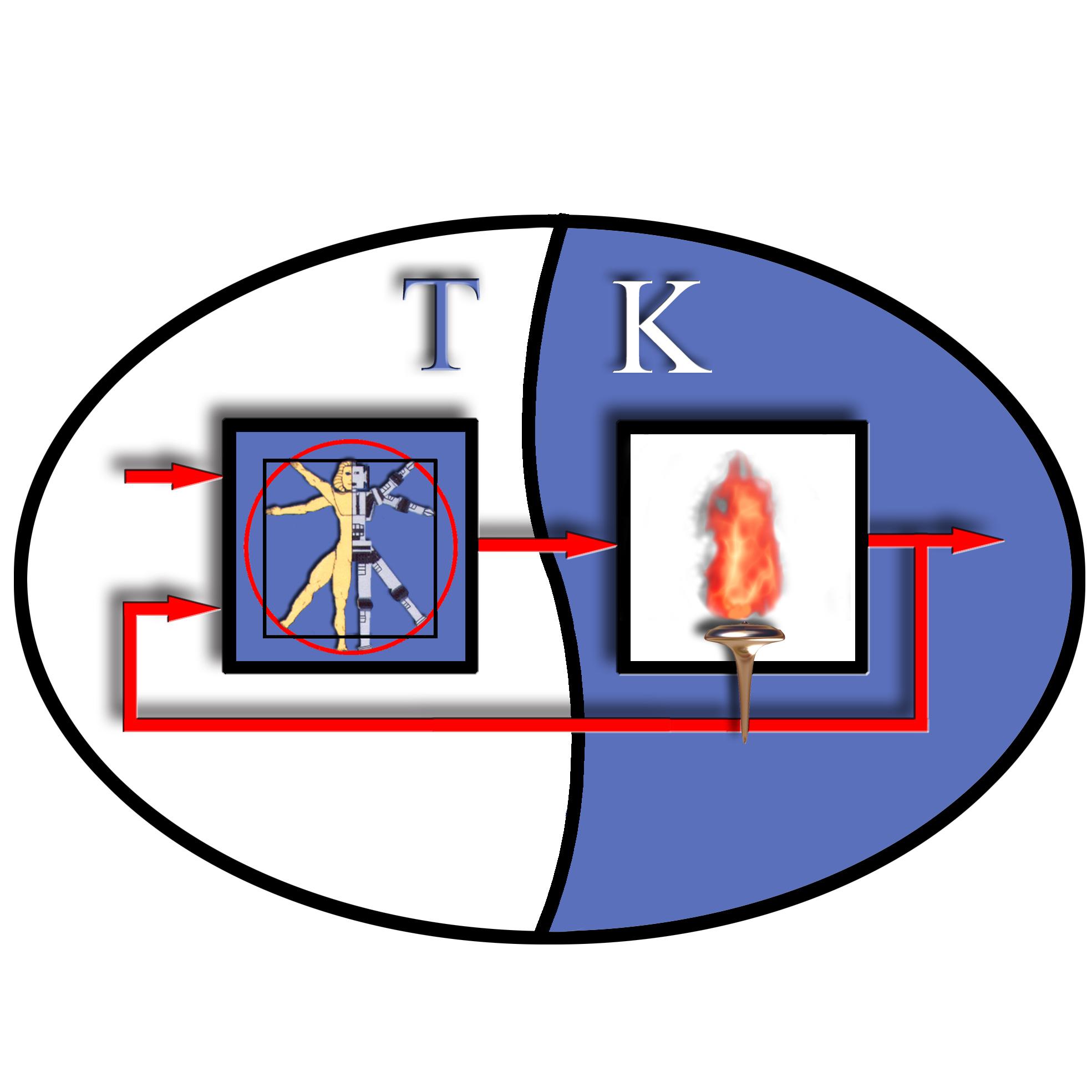
***«Белгородский государственный технологический университет***

***им. В.Г. Шухова»***

***(БГТУ им. В.Г. Шухова)***

***ИЭИТУС***

# *Кафедра «Техническая кибернетика»*



***Лабораторная работа №1***

*дисциплина:* ***«Микроконтроллеры в системах управления»***

***Выполнил:***

*студент группы УС-31*

*Сильченко С. А.*

***Принял:***

*доц. Кижук А.С.*

*Белгород 2019*

Белгород 2011

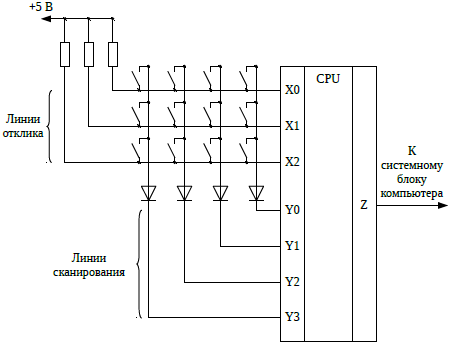
**Цель работы**

Необходимо написать программу для микроконтроллера i8051, которая позволяет считывать ASCI-код нажатой клавиши и отправлять его на другой контроллер с использованием УАПП.

**Теоретические сведения**

8-битное АЛУ может выполнять арифметические операции сложения, вычитания, умножения и деления; логические операции И, ИЛИ, исключающее ИЛИ, а также операции циклического сдвига, сброса, инвертирования и т.п. В АЛУ имеются программно недоступные регистры T1 и T2, предназначенные для временного хранения операндов, схема десятичной коррекции и схема формирования признаков.

Если рассмотреть сильно упрощенную принципиальную схему клавиатуры (для простоты представлена клавиатура с небольшим количеством клавиш) можно заметить, что все клавиши находятся в узлах матрицы.



**Рис. 1.** Упрощённая схема клавиатуры

Вертикальные линии матрицы клавиатуры подключены через диоды к выходным линиям Y0...Y3 (линии сканирования) порта Y, который является выходным портом для контроллера в том плане, что контроллер может устанавливать на линиях, связанных с портом, сигналы низкого и высокого напряжения, т. е. логические ноль «0» и единицу «1». Диоды предназначены для предотвращения коротких замыканий при одновременном нажатии нескольких клавиш.

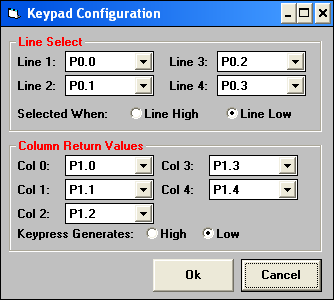
Горизонтальные линии матрицы соединены с линиями X0...X2 (линии отклика) входного порта X, значения каждой линии которого контроллер может считывать, определяя какое установлено на линии напряжение: соответствующее логическому «0» или «1». Кроме этого горизонтальные линии через подтягивающие резисторы соединены с напряжением питания +5 В, соответствующим логической «1», поэтому, когда ни одна клавиша не нажата и все контакты в узлах матрицы разомкнуты, на входах X0...X2 контроллера установлены «1».

В процессе функционирования контроллер устанавливает поочередно на каждой линии сканирования уровень напряжения логического «0», оставляя на остальных, кроме одной, уровень напряжения логической «1». Таким образом, если нажата клавиша, установка «0» на соответствующей вертикальной линии матрицы, приведет к появления сигнала «0» на соответствующей горизонтальной линии матрицы, что будет обнаружено контроллером, считывающим значения линий порта X. Зная на какой из сканирующих линий установлен в данный момент «0» и на какой линии отклика получен «0», контроллер клавиатуры определяет номер нажатой клавиши в матрицы. Также легко определяется, когда нажатая ранее клавиша отпускается.

Как только контроллер определил нажатие или отпускание клавиши, он посылает в центральный компьютер запрос на прерывание и номер клавиши в матрице, который однозначно зависит от схемы клавиатурной матрицы, но не от обозначений, нанесенных на поверхность клавиш. Этот номер называется скан-кодом (Scan Code). Слово scan «сканирование», подчеркивает тот факт, что клавиатурный компьютер сканирует клавиатуру для поиска нажатой клавиши.

Обычно программе нужен не порядковый номер нажатой клавиши, а код, соответствующий обозначению на этой клавише, то есть код ASCII. Код ASCII не связан напрямую со скан-кодом, так как одной и той же клавише могут соответствовать несколько значений кода ASCII в зависимости от состояния других клавиш. Например, клавиша <1> используется для ввода символов '1' и '!', если она была нажата вместе с клавишей <Shift>. Все преобразования скан-кода в код ASCII выполняются программно.

Преобразование начинается после того, как клавиатурный процессор отошлет центральному процессору запрос на прерывание и скан-код клавиши. При этом центральный процессор прерывает исполняемую программу и переходит на подпрограмму обработки прерывания от клавиатуры. Эта подпрограмма формирует согласно значению скан-кода двухбайтовый код с последующей засылкой его в буфер ввода данных с клавиатуры, заполнение которого происходит по мере нажатия клавиш и никак не связано с выполняемой программой.



**Рис. 2.** Настройки портов

**Код программы**

ORG 0h

;T1 M1=1 для вывода в монитор

;T0 M0=1 для

MOV TMOD,#00100001b

;передача с частотой 9600гц

MOV TH1,#0FDh

;опрос со частотой 137гц

MOV TH0,#0FFh

MOV TL0,#1Dh

;2-ой режим передачи

SETB SM1

CLR SM0

;опрос линии 0

L0:

;запуск таймера

SETB TR0

;ждём переполнения

JNB TF0,$

;остановка таймера

CLR TR0

;сброс флага переполнения

CLR TF0

;снова загружаем таймер

MOV TH0,#0FFh

MOV TL0,#1Dh

;опрашиваем линию Р0.0

CLR P0.0

;изменение на линии P1 соответсвует какому-то символу

JNB P1.0,CH1

JNB P1.1,CH2

JNB P1.2,CH3

JNB P1.3,CHC

JNB P1.4,CHD

;заканчиваем опрашивать линию Р0.0

SETB P0.0

;если нажатий не было, то переход на следующую линию

SJMP L1

;передача символа 1 на вывод

CH1:

MOV A,#31h

SETB P0.0

LJMP PR

;передача символа 2 на вывод

CH2:

MOV A,#32h

SETB P0.0

LJMP PR

;передача символа 3 на вывод

CH3:

MOV A,#33h

SETB P0.0

LJMP PR

;передача символа C на вывод

CHC:

MOV A,#43h

SETB P0.0

LJMP PR

;передача символа D на вывод

CHD:

MOV A,#44h

SETB P0.0

LJMP PR

;опрос линии 1

L1:

;запуск таймера

SETB TR0

;ждём переполнения

JNB TF0,$

;остановка таймера

CLR TR0

;сброс флага переполнения

CLR TF0

;снова загружаем таймер

MOV TH0,#0FFh

MOV TL0,#1Dh

;опрашиваем линию Р0.1

CLR P0.1

;изменение на линии P1 соответсвует какому-то символу

JNB P1.0,CH4

JNB P1.1,CH5

JNB P1.2,CH6

JNB P1.3,CHE

JNB P1.4,CHF

;заканчиваем опрашивать линию Р0.1

SETB P0.1

;если нажатий не было, то переход на следующую линию

SJMP L2

;передача символа 4 на вывод

CH4:

MOV A,#34h

SETB P0.1

LJMP PR

;передача символа 5 на вывод

CH5:

MOV A,#35h

SETB P0.1

LJMP PR

;передача символа 6 на вывод

CH6:

MOV A,#36h

SETB P0.1

LJMP PR

;передача символа E на вывод

CHE:

MOV A,#45h

SETB P0.1

LJMP PR

;передача символа F на вывод

CHF:

MOV A,#46h

SETB P0.1

LJMP PR

;опрос линии 2

L2:

;запуск таймера

SETB TR0

;ждём переполнения

JNB TF0,$

;остановка таймера

CLR TR0

;сброс флага переполнения

CLR TF0

;снова загружаем таймер

MOV TH0,#0FFh

MOV TL0,#1Dh

;опрашиваем линию Р0.2

CLR P0.2

;изменение на линии P1 соответсвует какому-то символу

JNB P1.0,CH7

JNB P1.1,CH8

JNB P1.2,CH9

JNB P1.3,CHG

JNB P1.4,CHH

;заканчиваем опрашивать линию Р0.2

SETB P0.2

;если нажатий не было, то переход на следующую линию

SJMP L3

;передача символа 7 на вывод

CH7:

MOV A,#37h

SETB P0.2

SJMP PR

;передача символа 8 на вывод

CH8:

MOV A,#38h

SETB P0.2

SJMP PR

;передача символа 9 на вывод

CH9:

MOV A,#39h

SETB P0.2

SJMP PR

;передача символа G на вывод

CHG:

MOV A,#47h

SETB P0.2

SJMP PR

;передача символа H на вывод

CHH:

MOV A,#48h

SETB P0.2

SJMP PR

;опрос линии 3

L3:

;запуск таймера

SETB TR0

;ждём переполнения

JNB TF0,$

;остановка таймера

CLR TR0

;сброс флага переполнения

CLR TF0

;снова загружаем таймер

MOV TH0,#0FFh

MOV TL0,#1Dh

;опрашиваем линию Р0.3

CLR P0.3

;изменение на линии P1 соответсвует какому-то символу

JNB P1.0,CHA

JNB P1.1,CH0

JNB P1.2,CHB

JNB P1.3,CHI

JNB P1.4,CHJ

;заканчиваем опрашивать линию Р0.3

SETB P0.3

;если нажатий не было, то переход на следующую линию

LJMP L0

;передача символа A на вывод

CHA:

MOV A,#41h

SETB P0.3

SJMP PR

;передача символа 0 на вывод

CH0:

MOV A,#30h

SETB P0.3

SJMP PR

;передача символа B на вывод

CHB:

MOV A,#42h

SETB P0.3

SJMP PR

;передача символа I на вывод

CHI:

MOV A,#49h

SETB P0.3

SJMP PR

;передача символа J на вывод

CHJ:

MOV A,#4Ah

SETB P0.3

SJMP PR

;вывод

PR:

MOV SBUF,A

;переход на новую строку

MOV A,#13

MOV SBUF,A

MOV A,#10

MOV SBUF,A

;переход к опросу линии 0

LJMP L0

END

**Листинг программы**

MCS-51 MACRO ASSEMBLER KLAVA1 04/07/;9 PAGE 1

DOS 5.0 (038-N) MCS-51 MACRO ASSEMBLER, V2.3

OBJECT MODULE PLACED IN KLAVA1.OBJ

ASSEMBLER INVOKED BY: E:\KOMPIL\ASM51.EXE KLAVA1.ASM

LOC OBJ LINE SOURCE

0000 1 ORG 0h

2 ;T1 M1=1 dk^ b{bnd` b lnmhrnp

3 ;T0 M0=1 dk^

0000 758921 4 MOV TMOD,#00100001b

5

6 ;oeped`w` q w`qrnrni 9600cv

0003 758DFD 7 MOV TH1,#0FDh

8 ;nopnq qn w`qrnrni 137cv

0006 758CFF 9 MOV TH0,#0FFh

0009 758A1D 10 MOV TL0,#1Dh

11 ;2-ni pefhl oeped`wh

000C D29E 12 SETB SM1

000E C29F 13 CLR SM0

14

15 ;nopnq khmhh 0

16 L0:

17 ;g`osqj r`ilep`

0010 D28C 18 SETB TR0

19 ;fd8l oepeonkmemh^

0012 308DFD 20 JNB TF0,$

21 ;nqr`mnbj` r`ilep`

0015 C28C 22 CLR TR0

23 ;qapnq tk`c` oepeonkmemh^

0017 C28D 24 CLR TF0

25 ;qmnb` g`cpsf`el r`ilep

0019 758CFF 26 MOV TH0,#0FFh

001C 758A1D 27 MOV TL0,#1Dh

28 ;nop`xhb`el khmh~ P0.0

001F C280 29 CLR P0.0

30 ;hglememhe m` khmhh P1 qnnrberqbser j`jnls-rn qhlbnks

0021 309010 31 JNB P1.0,CH1

0024 309114 32 JNB P1.1,CH2

0027 309218 33 JNB P1.2,CH3

002A 30931C 34 JNB P1.3,CHC

002D 309420 35 JNB P1.4,CHD

36 ;g`j`mwhb`el nop`xhb`r| khmh~ P0.0

0030 D280 37 SETB P0.0

38 ;eqkh m`f`rhi me a{kn, rn oepeund m` qkeds~ys~ khmh~

0032 8023 39 SJMP L1

40

41 ;oeped`w` qhlbnk` 1 m` b{bnd

42 CH1:

0034 7431 43 MOV A,#31h

0036 D280 44 SETB P0.0

0038 020123 45 LJMP PR

46

47 ;oeped`w` qhlbnk` 2 m` b{bnd

48 CH2:

003B 7432 49 MOV A,#32h

003D D280 50 SETB P0.0

MCS-51 MACRO ASSEMBLER KLAVA1 04/07/;9 PAGE 2

LOC OBJ LINE SOURCE

003F 020123 51 LJMP PR

52

53 ;oeped`w` qhlbnk` 3 m` b{bnd

54 CH3:

0042 7433 55 MOV A,#33h

0044 D280 56 SETB P0.0

0046 020123 57 LJMP PR

58

59 ;oeped`w` qhlbnk` C m` b{bnd

60 CHC:

0049 7443 61 MOV A,#43h

004B D280 62 SETB P0.0

004D 020123 63 LJMP PR

64

65 ;oeped`w` qhlbnk` D m` b{bnd

66 CHD:

0050 7444 67 MOV A,#44h

0052 D280 68 SETB P0.0

0054 020123 69 LJMP PR

70

71 ;nopnq khmhh 1

72 L1:

73 ;g`osqj r`ilep`

0057 D28C 74 SETB TR0

75 ;fd8l oepeonkmemh^

0059 308DFD 76 JNB TF0,$

77 ;nqr`mnbj` r`ilep`

005C C28C 78 CLR TR0

79 ;qapnq tk`c` oepeonkmemh^

005E C28D 80 CLR TF0

81 ;qmnb` g`cpsf`el r`ilep

0060 758CFF 82 MOV TH0,#0FFh

0063 758A1D 83 MOV TL0,#1Dh

84 ;nop`xhb`el khmh~ P0.1

0066 C281 85 CLR P0.1

86 ;hglememhe m` khmhh P1 qnnrberqbser j`jnls-rn qhlbnks

0068 309010 87 JNB P1.0,CH4

006B 309114 88 JNB P1.1,CH5

006E 309218 89 JNB P1.2,CH6

0071 30931C 90 JNB P1.3,CHE

0074 309420 91 JNB P1.4,CHF

92 ;g`j`mwhb`el nop`xhb`r| khmh~ P0.1

0077 D281 93 SETB P0.1

94 ;eqkh m`f`rhi me a{kn, rn oepeund m` qkeds~ys~ khmh~

0079 8023 95 SJMP L2

96

97 ;oeped`w` qhlbnk` 4 m` b{bnd

98 CH4:

007B 7434 99 MOV A,#34h

007D D281 100 SETB P0.1

007F 020123 101 LJMP PR

102

103 ;oeped`w` qhlbnk` 5 m` b{bnd

104 CH5:

0082 7435 105 MOV A,#35h

MCS-51 MACRO ASSEMBLER KLAVA1 04/07/;9 PAGE 3

LOC OBJ LINE SOURCE

0084 D281 106 SETB P0.1

0086 020123 107 LJMP PR

108

109 ;oeped`w` qhlbnk` 6 m` b{bnd

110 CH6:

0089 7436 111 MOV A,#36h

008B D281 112 SETB P0.1

008D 020123 113 LJMP PR

114

115 ;oeped`w` qhlbnk` E m` b{bnd

116 CHE:

0090 7445 117 MOV A,#45h

0092 D281 118 SETB P0.1

0094 020123 119 LJMP PR

120

121 ;oeped`w` qhlbnk` F m` b{bnd

122 CHF:

0097 7446 123 MOV A,#46h

0099 D281 124 SETB P0.1

009B 020123 125 LJMP PR

126

127 ;nopnq khmhh 2

128 L2:

129 ;g`osqj r`ilep`

009E D28C 130 SETB TR0

131 ;fd8l oepeonkmemh^

00A0 308DFD 132 JNB TF0,$

133 ;nqr`mnbj` r`ilep`

00A3 C28C 134 CLR TR0

135 ;qapnq tk`c` oepeonkmemh^

00A5 C28D 136 CLR TF0

137 ;qmnb` g`cpsf`el r`ilep

00A7 758CFF 138 MOV TH0,#0FFh

00AA 758A1D 139 MOV TL0,#1Dh

140 ;nop`xhb`el khmh~ P0.2

00AD C282 141 CLR P0.2

142 ;hglememhe m` khmhh P1 qnnrberqbser j`jnls-rn qhlbnks

00AF 309010 143 JNB P1.0,CH7

00B2 309113 144 JNB P1.1,CH8

00B5 309216 145 JNB P1.2,CH9

00B8 309319 146 JNB P1.3,CHG

00BB 30941C 147 JNB P1.4,CHH

148 ;g`j`mwhb`el nop`xhb`r| khmh~ P0.2

00BE D282 149 SETB P0.2

150 ;eqkh m`f`rhi me a{kn, rn oepeund m` qkeds~ys~ khmh~

00C0 801E 151 SJMP L3

152

153 ;oeped`w` qhlbnk` 7 m` b{bnd

154 CH7:

00C2 7437 155 MOV A,#37h

00C4 D282 156 SETB P0.2

00C6 805B 157 SJMP PR

158

159 ;oeped`w` qhlbnk` 8 m` b{bnd

160 CH8:

MCS-51 MACRO ASSEMBLER KLAVA1 04/07/;9 PAGE 4

LOC OBJ LINE SOURCE

00C8 7438 161 MOV A,#38h

00CA D282 162 SETB P0.2

00CC 8055 163 SJMP PR

164

165 ;oeped`w` qhlbnk` 9 m` b{bnd

166 CH9:

00CE 7439 167 MOV A,#39h

00D0 D282 168 SETB P0.2

00D2 804F 169 SJMP PR

170

171 ;oeped`w` qhlbnk` G m` b{bnd

172 CHG:

00D4 7447 173 MOV A,#47h

00D6 D282 174 SETB P0.2

00D8 8049 175 SJMP PR

176

177 ;oeped`w` qhlbnk` H m` b{bnd

178 CHH:

00DA 7448 179 MOV A,#48h

00DC D282 180 SETB P0.2

00DE 8043 181 SJMP PR

182

183 ;nopnq khmhh 3

184 L3:

185 ;g`osqj r`ilep`

00E0 D28C 186 SETB TR0

187 ;fd8l oepeonkmemh^

00E2 308DFD 188 JNB TF0,$

189 ;nqr`mnbj` r`ilep`

00E5 C28C 190 CLR TR0

191 ;qapnq tk`c` oepeonkmemh^

00E7 C28D 192 CLR TF0

193 ;qmnb` g`cpsf`el r`ilep

00E9 758CFF 194 MOV TH0,#0FFh

00EC 758A1D 195 MOV TL0,#1Dh

196 ;nop`xhb`el khmh~ P0.3

00EF C283 197 CLR P0.3

198 ;hglememhe m` khmhh P1 qnnrberqbser j`jnls-rn qhlbnks

00F1 309011 199 JNB P1.0,CHA

00F4 309114 200 JNB P1.1,CH0

00F7 309217 201 JNB P1.2,CHB

00FA 30931A 202 JNB P1.3,CHI

00FD 30941D 203 JNB P1.4,CHJ

204 ;g`j`mwhb`el nop`xhb`r| khmh~ P0.3

0100 D283 205 SETB P0.3

206 ;eqkh m`f`rhi me a{kn, rn oepeund m` qkeds~ys~ khmh~

0102 020010 207 LJMP L0

208

209 ;oeped`w` qhlbnk` A m` b{bnd

210 CHA:

0105 7441 211 MOV A,#41h

0107 D283 212 SETB P0.3

0109 8018 213 SJMP PR

214

215 ;oeped`w` qhlbnk` 0 m` b{bnd

MCS-51 MACRO ASSEMBLER KLAVA1 04/07/;9 PAGE 5

LOC OBJ LINE SOURCE

216 CH0:

010B 7430 217 MOV A,#30h

010D D283 218 SETB P0.3

010F 8012 219 SJMP PR

220

221 ;oeped`w` qhlbnk` B m` b{bnd

222 CHB:

0111 7442 223 MOV A,#42h

0113 D283 224 SETB P0.3

0115 800C 225 SJMP PR

226

227 ;oeped`w` qhlbnk` I m` b{bnd

228 CHI:

0117 7449 229 MOV A,#49h

0119 D283 230 SETB P0.3

011B 8006 231 SJMP PR

232

233 ;oeped`w` qhlbnk` J m` b{bnd

234 CHJ:

011D 744A 235 MOV A,#4Ah

011F D283 236 SETB P0.3

0121 8000 237 SJMP PR

238

239 ;b{bnd

240 PR:

0123 F599 241 MOV SBUF,A

242 ;oepeund m` mnbs~ qrpnjs

0125 740D 243 MOV A,#13

0127 F599 244 MOV SBUF,A

0129 740A 245 MOV A,#10

012B F599 246 MOV SBUF,A

247 ;oepeund j nopnqs khmhh 0

012D 020010 248 LJMP L0

249

250 END

MCS-51 MACRO ASSEMBLER KLAVA1 04/07/;9 PAGE 6

SYMBOL TABLE LISTING

------ ----- -------

N A M E T Y P E V A L U E A T T R I B U T E S

CH0. . . . C ADDR 010BH A

CH1. . . . C ADDR 0034H A

CH2. . . . C ADDR 003BH A

CH3. . . . C ADDR 0042H A

CH4. . . . C ADDR 007BH A

CH5. . . . C ADDR 0082H A

CH6. . . . C ADDR 0089H A

CH7. . . . C ADDR 00C2H A

CH8. . . . C ADDR 00C8H A

CH9. . . . C ADDR 00CEH A

CHA. . . . C ADDR 0105H A

CHB. . . . C ADDR 0111H A

CHC. . . . C ADDR 0049H A

CHD. . . . C ADDR 0050H A

CHE. . . . C ADDR 0090H A

CHF. . . . C ADDR 0097H A

CHG. . . . C ADDR 00D4H A

CHH. . . . C ADDR 00DAH A

CHI. . . . C ADDR 0117H A

CHJ. . . . C ADDR 011DH A

L0 . . . . C ADDR 0010H A

L1 . . . . C ADDR 0057H A

L2 . . . . C ADDR 009EH A

L3 . . . . C ADDR 00E0H A

P0 . . . . D ADDR 0080H A

P1 . . . . D ADDR 0090H A

PR . . . . C ADDR 0123H A

SBUF . . . D ADDR 0099H A

SM0. . . . B ADDR 0098H.7 A

SM1. . . . B ADDR 0098H.6 A

TF0. . . . B ADDR 0088H.5 A

TH0. . . . D ADDR 008CH A

TH1. . . . D ADDR 008DH A

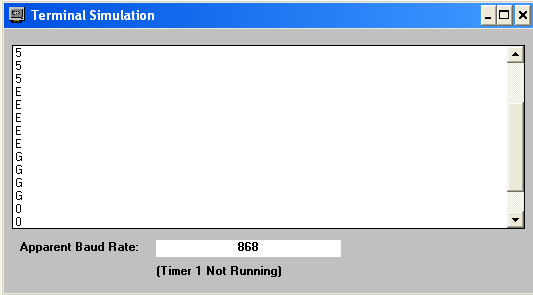
TL0. . . . D ADDR 008AH A

TMOD . . . D ADDR 0089H A

TR0. . . . B ADDR 0088H.4 A

REGISTER BANK(S) USED: 0

ASSEMBLY COMPLETE, NO ERRORS FOUND



**Рис. 3.** Пример работы

**Вывод**

В ходе выполнения данной лабораторной работы была написана программа для микроконтроллера i8051, которая позволяет считывать ASCI-код нажатой клавиши и отправлять его на другой контроллер с использованием УАПП.