

1. Цель работы

Освоить основные приемы табличного преобразования форматов представления чисел и приемы работы с семисегментными индикаторами.

1. Исходные данные

Таблица 1. Входные данные.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Индикация | АЦП, бит | Кнопка 1 | Кнопка 2 |
| 17 | 16x | 6 | Y = X and (-Y) | Ввод Х |

Три семисегментных индикатора → P0, P1, P2 соответственно;

Кнопки и выходные линии АЦП → P3.

1. Результаты работы программы

На рисунке 3.1. представлен листинг кода программы:

|  |
| --- |
| yl equ 30h  yh equ 31h  xl equ 32h  xh equ 33h  org 0  jmp main  org 30h  main:  mov SP, #100  mov yl, #low(27) ;yl -> 1B  mov yh, #high(27) ;yh -> 00  ;кнопки будут висеть на портах P3.6 и P3.7  ;activation  mov p0, #0  mov p1, #0  mov p2, #0  mov p3, #0ffh  ;central loop  ;============  loop:  ;формула  mov a, p3  anl a, #64  add a, #0  jz vvod  call form  jmp lo  vvod:  ;ввод X  mov a, p3  anl a, #128  add a, #0  jz wheel  call input  jmp lo  wheel:  mov a, xl  jz nl  jmp lo  nl:  mov a, p3  anl a, #63  add a, #0  jz lo  call ferrisWheel  jmp lo  lo: jmp loop  ;============  ;SubFunction  form:  loopForm:  jb p3.6, loopForm  clr c  mov a, #0 ;вычитание из нуля  subb a, yl ;  mov r3, a ;  mov a, #0 ;  subb a, yh ;  mov r4, a ;  mov a, r3 ;операция "И" и запись  anl a, xl ;  mov yl, a ;  ;  mov a, r4 ;  anl a, xh ;  mov yh, a ;  mov a, yl ;разделение переменных для их последующего вывода  anl a, #240 ;  swap a ;  mov r6, a ;  mov a, yl ;  anl a, #15 ;  mov r5, a ;  mov p2, #63  mov p1, #63  mov p0, #63  loopView:  mov dptr, #tab  mov a, r5  movc a, @a+dptr  mov p2, a  mov a, r6  movc a, @a+dptr  mov p1, a  jb p3.6, loopView  ret  input:  loopInput:  jb p3.7, loopInput  ;сохранение реальных чисел  mov a, xl  jnz null  mov xl, r2  jmp nil  null:  mov xl, #0  nil:  ret  ferrisWheel:  mov dptr, #tab  mov a, p3  clr acc.6  clr acc.7  loopTr16: ;цикл перевода в 16-ную  mov b, #16  div ab  mov r1, a ;второй разряд  mov r0, b ;первый разряд  cjne r0, #16, d16  d16: jc write  jmp loopTr16  write:  swap a  orl a, r0  mov r2, a  mov a, r0 ;вывод первого разряда (единицы)  movc a, @a+dptr ;  mov p2, a ;  mov a, r1 ;вывод второго разряда (десятки)  movc a, @a+dptr ;  mov p1, a ;  mov a, #0 ;вывод третьего разряда (сотни)  movc a, @a+dptr ;  mov p0, a ;  ret  stop: jmp stop  tab: db 3fh,06h, 5bh, 4fh, 66h, 6dh, 7dh, 07h, 7fh, 6fh, 77h, 7ch, 39h, 5eh, 79h, 71h  ; 0 1 2 3 4 5 6 7 8 9 A B C D E F  end |

Рисунок 3.1. – Листинг кода программы

1. Результаты работы программы



Рисунок 4.1. – Окружение микроконтроллера после запуска моделирования

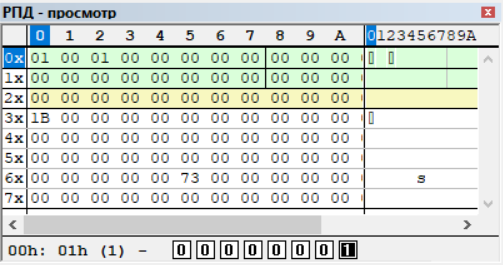


Рисунок 4.2. – РПД микроконтроллера после запуска моделирования (в ячейке 02h (**r2**) хранится число введенное с помощью АЦП; в ячейке 30h (**yl**) хранится первоначальное число Y в данном случае оно равно 2710 или 1B16)

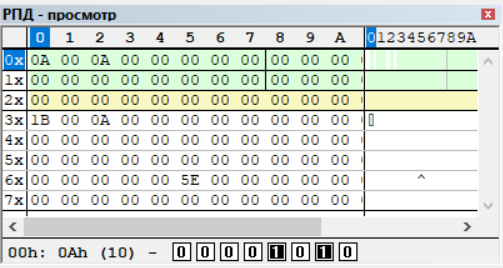


Рисунок 4.3. – РПД после нажатия на кнопку «ввод X» число 0Ah записалось в ячейку 32h (**xl**)

После ввода **xl** блокируется доступ к кнопке «ввод X», так как в **xl** уже есть данные, но после повторного нажатия на данную кнопку данные очищаются и есть возможность задать данные заново.

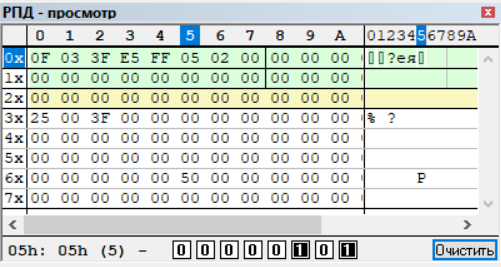


Рисунок 4.4. – РПД после задания ввода числа 63 (3fh) ← ячейка 32h; после нажатия на кнопку «Формула» в ячейках 03h и 04h (r3 и r4) появляется число -Y; в ячейках 05h и 06h находится разложенное на разряды число Y после применения формулы; в ячейке 30h (**yl**) результат после нажатия кнопки «Формула»



Рисунок 4.6. – Ввод числа X



Рисунок 4.5. – Результат работы формулы

Таблица 2. Результаты работы программы.

|  |  |  |
| --- | --- | --- |
| Y, hex | -Y, hex | X, hex |
| 1B | FFE5 | 3F |
| 25 | FFDB | 37 |
| 13 | FFED | 32 |
| 20 | … | … |

Результаты работы программы верны, в этом можно убедиться, если проверить вычисления на калькуляторе, например для числа **13h**:



Рисунок 4.6. – Отрицательное число Y

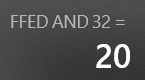


Рисунок 4.7. – Операция «И»

1. Вывод

В результате выполнения данной лабораторной работы, основные приемы табличного преобразования форматов представления чисел и приемы работы с семисегментными индикаторами, также получили навыки перевода в другие системы счисления с помощью ассемблера 8051.