

1. Цель работы

Освоить основные приемы использования таймера для

реализации требуемого интервала квантования времени и формирования

импульсов заданной длительности.

1. Исходные данные

Вариант №17

Управление линейкой светодиодов. На линейке из 16 светодиодов

отображать число из ячеек ОЗУ, которое модифицировать периодически

по алгоритму умножения начального числа (тройка) на 4. Период

изменения числа задать посредством АЦП от 0 до 1.5 с. с разрешением 0.1 с.

Заданный период отображать на цифровом индикаторе. Кнопки управления:

Пуск/Стоп (1) и Начальное состояние (2).

1. Ход работы

На рисунке 3.1 представлен листинг кода программы:

|  |
| --- |
| xl equ 30h  xh equ 31h  yl equ 32h  yh equ 33h  N1 equ 34h  N2 equ 35h  N0 equ 36h  org 0  **jmp** main  ;прерывание таймера TF0  org 0bh  **jmp** timeOut  org 30h  main:  **mov** sp, #100  ;начальные значения  **mov** xl, #3  **mov** xh, #0  ;активация окружения  **mov** p0, #0  **mov** p1, #0  **mov** p2, #0  **mov** p3, #0cfh  ;активация прерываний  **mov** tl0, #low(7628h)  **mov** th0, #high(7628h)  **mov** tmod, #01h  **mov** ie, #82  ;=================================  LoopMain:  **jnb** tr0, skip  **jnb** tf0, $  **clr** tf0  **djnz** r0, skip  **mov** r0, N0  call timeOut  **jmp** LoopMain  ;=================================  ;проверка периферии  skip:  **mov** 24h, tcon  **jb** p3.6, start  **jb** p3.7, firstValue  **mov** a, p3  **anl** a, #15  **add** a, #0  **jnz** ferrisWheel  **jmp** LoopMain  ;начальное состояние  firstValue:  **jb** p3.7, $  **mov** p0, #3  **mov** p1, #0  **mov** xl, #3  **mov** xh, #0  **mov** yl, #0  **mov** yh, #0  **jmp** loopMain  ;обработка прервания TF0  timeOut:  **mov** tl0, #low(7628h)  **mov** th0, #high(7628h)  ;формула  **mov** a, xl  **mov** b, #4  **mul** ab  **mov** yl, a  **mov** xl, yl  **mov** p0, yl  **mov** a, b  **anl** a, #3  **add** a, #0  **jnz** levelOne  **jmp** count  levelOne: ;процедура обработки ошибки при переходе из первой линейки светодиодов во вторую  **mov** yh, b  **mov** xh, yh  **mov** p1, yh  **jmp** endl  count:  **mov** a, xh  **mov** b, #4  **mul** ab  **mov** yh, a  **mov** xh, yh  **mov** p1, yh  endl:  **reti**  ;пуск/стоп  start:  **jb** p3.6, $  **cpl** tr0  **jmp** loopMain  ferrisWheel:  **mov** dptr, #tab ;ацп изменяет значение в r0 тем самым меняет период изменения числа на индикаторе  **mov** a, p3  **anl** a, #15  **mov** b, #5 ;умножаем на 5 для масштабирования коэффициента  **mul** ab ;  **mov** N0, a ;  **mov** a, p3  **anl** a, #15 ;возвращаемся к шкале ацп  **mov** b, #10 ;сохраняем разделенные числа в регистрах r3 и r2  **div** ab ;  **mov** r3, a ;старший разряд  **jnz** RazN1 ;  **mov** N1, #0 ;  **jmp** next1 ;  RazN1: **mov** N1, #1 ;  next1: ;  **mov** a, b ;  **mov** r2, a ;младший разряд  **jnz** RazN2 ;  **mov** N2, #0 ;  **jmp** next2 ;  RazN2: **mov** N2, #2 ;  next2: ;  **jmp** indic ;  **jmp** loopMain ;    indic:  **mov** a, r3  **jz** L1  **sjmp** L2  L2: **setb** p3.4  **sjmp** L3  L1: **clr** p3.4  L3:  **setb** p3.5  **mov** a, r2  **movc** a, @a+dptr  **mov** p2, a  **ljmp** LoopMain  stop: **jmp** stop  tab: db 3fh, 06h, 5bh, 4fh, 66h, 6dh, 7dh, 07h, 7fh, 6fh, 80h  ; 0 1 2 3 4 5 6 7 8 9 d  end |

Рисунок 3.1 – Листинг кода программы

1. Результаты работы программы

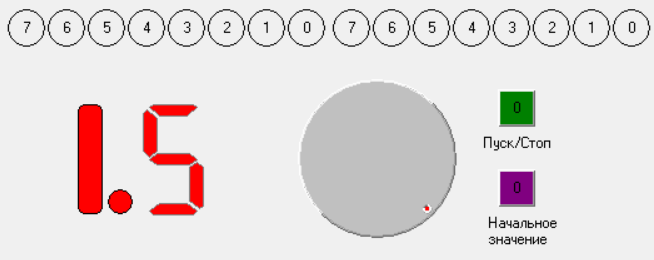


Рисунок 4.1 – Окружение микроконтроллера после запуска моделирования

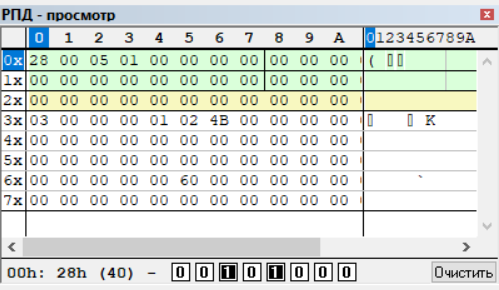


Рисунок 4.2 – РПД после запуска моделирования

На рисунке 4.2:

* в ячейках 02h (**r2**) и 03h (**r3**) находится разложенное на множители число с АЦП;
* в ячейках 34h (**N1**) и 35h (**N2**) находятся данные о том, выводить ли десятки или только единицы, или и то, и другое;
* в ячейке 36h (**N0**) число, переведенное в другой масштаб (оно пропорционально данным с АЦП);
* в ячейке 30h (**xl**) хранится первоначальное число, оно будет изменяться в результате работы формулы и выводиться на индикатор.

После нажатия на кнопку «Пуск/Стоп» запускается работа таймера, (при выставлении на АЦП значения 0,1с) по истечении 0,1с которого вызывается прерывание и срабатывает формула над данными, которые хранятся в ОЗУ.

Вид индикатора данных обусловлен наличием всего одного свободного порта.

После завершения индикации работы формулы, требуется нажать на кнопку «Начальное значение», она вернет все первоначальные данные на индикатор и в ОЗУ.

Если во время выполнения нажать на кнопку «Пуск/Стоп» процесс остановится, при повторном нажатии на данную кнопку процесс возобновится.

Во время работы также можно изменять период индикации работы формулы, измененный период будет отображаться на индикаторе данных, а его изменения будут применены на следующей итерации работы формулы.

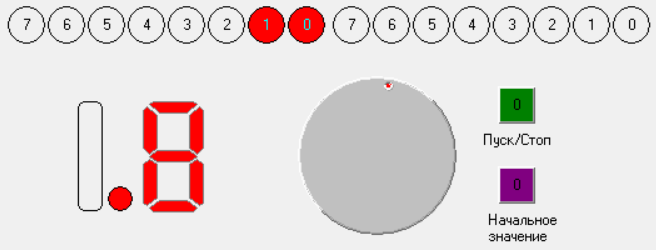


Рисунок 4.3 – Окружение микроконтроллера при работе с периодом 0,8с

Чтобы добиться интервала 0,1с в качестве цены деления на АЦП воспользовались счетчиком перезагрузок для этого использовали регистр r0 из-за которого прерывание вызывалось, только после нужного нам интервала времени. Настройка счетчика перезагрузок производилась следующим образом:

1. Посчитаем время перезагрузки таймера

При этом подразумевается, что регистры TL0 и TH0 будут всегда сбрасываться в 0.

1. Однако, в данном случае требуется установить другой интервал сброса, поэтому воспользуемся следующей формулой:

таким образом мы получили значение смещение, которое требуется поместить в регистры TL0 и TH0.

При таком интервале сброса была составлена таблица:

Таблица 4.1. Отношение численных значений со временем.

|  |  |
| --- | --- |
| Время t, c | Число на АЦП |
| 0,1 | 5 |
| 0,2 | 10 |
| 0,3 | 15 |
| 0,4 | 20 |
| 0,5 | 25 |

Из таблицы 4.1. видно что число на АЦП должно быть кратно 5, однако в данном случае использовался 4-х разрядный АЦП, максимально возможное число в котором 15, следовательно для реализации изменений периода отображения значений на индикаторе от 0,1с до 1,5с требуется умножать каждое число на 5, тогда шкала чисел на АЦП будет от 5 до 75.

1. Вывод

В результате выполнения данной лабораторной работы, освоили основные приемы использования таймера для реализации требуемого интервала квантования времени с помощью счетчика перезагрузок, за счет которого реализовали регулируемое изменение интервалов времени при отображении изменяющихся значений в озу на индикаторе из 16 светодиодов.