

Описание:

ATmega168 приобретать специально не обязательно

ATmega88 есть две причины для ее использования: 1- у неё еепром 512байт (в ATmega 328 1k) значит войдут более длинные мелодии, 2- если я когда ни будь сделаю обновление которое не влезет в 48-ю

ATmega8, вариант на всякий случай, если он уже есть или нет возможности приобрести ATmega48/88/168.

У меня ATmega48 потому что они были в некотором количестве

```
*****  
;  
; !!! FUSES !!!
```

```
*****  
;  
ATmega48/88/168
```

```
; SUT_CKSEL = (Int. RC Osc. 8 MHz; Start-up time PWRDWN/RESET: 6 CK/14 CK + 65 ms;  
default value) то есть оставить заводские
```

Убрать Внутренний делитель на 8: [CKDIV8=1]

~~;- BODLEVEL = (Brown-out detection at VCC=1,8V)~~ при использовании ионистора и при частом отключении можно включить, чтоб избежать глюков, но это уменьшит время работы в три-четыре раза!

В ATmega48P/88P/168P вроде как есть возможность отключать программно, я еще не разобрался что да как, но будем над этим работать!

все возможности экономии для простых 48/88/168 задействованы

```
*****  
;  
ATmega8; SUT_CKSEL = (Int. RC Osc. 8 MHz; Start-up time: 14 CK + 65 ms) [CKSEL=0100  
SUT=10]
```

```
;- BODLEVEL = (Brown-out detection at VCC=2,7V) [BODLEVEL=1] и
```

~~Brown-out detection-enabled; [BODEN=0]~~ лучше про это забыть, он и так много потребляет

Управление.

Кнопки:

S1 – (-);

S2 – (+);

S3 – (*) если не перепутал :)

Ради эксперимента пробовал подключать энкодер от колесика «сдохшей» мыши, результат можно попробовать, но только как дополнительная опция

Длинное нажатие кнопки

- * - установка часов, из любого режима
- + - установка будильника, из любого режима
- - изменение настроек

разовое нажатие кнопки

- * - переключение на дату, дату2, температуру1 и 2 и обратно, также обратно по таймеру
- + - просмотр будильника

в режиме любых установок

- * - выбор/сохранение
- + - плюс , для дня недели будильника выбор дня, 99=общее запрещение

- - минус, для дня недели будильника снятие-установка флага

установка часов: год>мес>день>Час>минута,
день недели рассчитывается автоматом перед записью в RTC + сек=0
если настройки не менялись, то сохранение не производится

установка будильника: Час > мин > (ON/OFF+дни недели) > 8X=мелодия(X=0...2) > число повторов
тоже и для второго

изменение настроек:

1. Уровень понижения яркости, в режиме “8” вместо изменения уровня происходит отключение генератора.
2. время вкл. ночной режим
3. время выкл. ночной режим вкл=выкл то раб. постоянно
4. 1=показ даты в начале минуты 2= показ температуры после 20й и 40й секунды, 3= и то и другое
5. 1=звук кнопок 2=ежечасный сигнал 3= и то и другое
6. установка коррекции $\pm 12,7$ сек
7. ХЗ на чё она влияет, пока не использовал ☺

Мелодии будильника записываются в EEPROM память контроллера

по этому для их замены достаточно прошить только EEPROM

Подробно о структуре EEPROM и о том, как запрограммировать мелодию, показано в файле «EEPROM.asm», его я вам предлагаю подготовить для прошивки самостоятельно.

Формат кодирования похож на тот, что описан в книге (Белов А.В. Микроконтроллеры AVR в радиолюбительской практике. – СПб.: Наука и Техника, 2007. – 352с.: ил.).

Для кодирования мелодии, можете воспользоваться её нотной записью или записью мелодии в формате текстового описания, что-то типа: «...Fis2(1/16) G2(1/16) A2(1/4.)...» только имейте в виду, что полутонные длительности «(1/4.)» с точкой, здесь не поддерживаются. Этот формат был в свое время популярен, среди пользователей мобильных телефонов Siemens по этому думаю, Интернет поможет вам подобрать подходящую мелодию. Только смотрите чтобы размер EEPROM, соответствовал ATmega48 -256 байт или ATmega8/88/168 -512 байт.

Его после изменения можно компилировать совершенно отдельно, только включить описание соответствующего контроллера или самостоятельно указать константу «EEPROMEND», если EEPROM не шить будет работать сирена

О настройке “ЖЕЛЕЗА”:

Для преобразователя использован аппаратный ШИМ Timer0, частота, фиксированная 31250 Гц, мощность обеспечивается подбором индуктивности дросселя, файл с расчетом где-то затерялся.

Приведу наугад несколько примеров: для ИН17, ИН 16 надо использовать где-то 820-1000 мкГн, я поставил 820х0,6А самую большую, что у меня была, для ИН12, ИН 14 560х1А, 560х0.7 у меня тянула (4-е ин18 + ин9 на всю шкалу) ток больше 20 мА, это предел, но мосфет был едва теплый, для ИН18 470х1,5А.

Угол открытия транзистора, при изменении нагрузки и напряжения питания, видно на 12-й ноге осциллографом, 0,6-0,7 = нормально 0,9 = max, регулятор использует (внутренний ИОН на 1,1В), выходное напряжение задается делителем R6, R9.
Программной настройки не предусмотрено!

R3,R5 и стабилитрон D1 – сенсор питающего напряжения, для перевода в режим пониженного потребления, необходимо чтобы при снижении питающего напряжения до 7-8В на входе INT0 было $<VCC/2$ (если $VCC=V_{bat\ min}$), но и не превышало $VCC=5V$ в нормальном режиме работы, еще можно по пробовать поставить электролит 5-10 мкФ параллельно C7.

Резистор R2 нужен для защиты от КЗ (используется прерывание) при паразитном питании датчиков. Датчики ds1820, 18s20, 18b20.