



(/)

Разделы

Новости (/index.php/novosti.html)
Встраиваемые системы (/index.php/embedded-programming.html)
Программирование AVR (/index.php/programming-avr.html)
Программирование ARM (/index.php/programmirovanie-arm.html)
Инструменты/технологии (/index.php/instruments-technologies.html)
Как подключить (/index.php/how-connection.html)
Компоненты (/index.php/electronic-components.html)
RTOS (/index.php/rtos.html)
Софт (/index.php/iar-embedded-workbench.html)
Проекты (/index.php/projects-avr.html)
Ссылки (/index.php/links.html)

1wire (/index.php/programming-avr/itemlist/tag/1wire.html) arm (/index.php/programming-avr/itemlist/tag/arm.html)

avr (/index.php/programming-avr/itemlist/tag/avr.html)

avr программатор (/index.php/programming-avr/itemlist/tag/avr-программатор.html) ds18b20 (/index.php/programming-avr/itemlist/tag/ds18b20.html)

eeprom (/index.php/programming-avr/itemlist/tag/eeprom.html) i2c (/index.php/programming-avr/itemlist/tag/i2c.html)

IAR (/index.php/programming-avr/itemlist/tag/IAR.html) lcd (/index.php/programming-avr/itemlist/tag/lcd.html)

tsop (/index.php/programming-avr/itemlist/tag/tsop.html) twi (/index.php/programming-avr/itemlist/tag/twi.html)

алгоритмы (/index.php/programming-avr/itemlist/tag/алгоритмы.html)

библиотеки (/index.php/programming-avr/itemlist/tag/библиотеки.html) датчик (/index.php/programming-avr/itemlist/tag/датчик.html)

драйвер (/index.php/programming-avr/itemlist/tag/драйвер.html) интерфейс (/index.php/programming-avr/itemlist/tag/интерфейс.html)

компоненты (/index.php/programming-avr/itemlist/tag/компоненты.html) макросы (/index.php/programming-avr/itemlist/tag/макросы.html)

oy (/index.php/programming-avr/itemlist/tag/oy.html)

программирование микроконтроллеров (/index.php/programming-avr/itemlist/tag/программирование-микроконтроллеров.html)

расчет (/index.php/programming-avr/itemlist/tag/расчет.html)

семисегментный индикатор (/index.php/programming-avr/itemlist/tag/семисегментный-индикатор.html) си (/index.php/programming-avr/itemlist/tag/си.html)

событийная система (/index.php/programming-avr/itemlist/tag/событийная-система.html)

схемотехника (/index.php/programming-avr/itemlist/tag/схемотехника.html) таймер (/index.php/programming-avr/itemlist/tag/таймер.html)

управление (/index.php/programming-avr/itemlist/tag/управление.html) устройства (/index.php/programming-avr/itemlist/tag/устройства.html)

учебный курс avr (/index.php/programming-avr/itemlist/tag/учебный-курс-avr.html)

шим (/index.php/programming-avr/itemlist/tag/шим.html)

Библиотека для опроса кнопок



29/04/2014 - 06:02 | Pavel Bobkov

Введение

Написал библиотеку для опроса кнопок.

Особенности библиотеки:

- предназначена для AVR
- работает с компиляторами GCC, IAR, CodeVision
- позволяет опрашивать до 32 цифровых входов
- имеет программную защиту от дребезга контактов
- распознает нажатие, удержание, отпускание и двойное нажатие на кнопку
- позволяет индивидуально задавать активный уровень и обрабатываемые события
- зафиксированные события сохраняются в кольцевом буфере
- позволяет отключать неиспользуемые функции
- легко интегрируется в готовый проект

Состав библиотеки

buttons.h - заголовочный файл библиотеки, в котором задаются настройки
buttons.c - файл реализация функций

Подключение к проекту

1. Копируем все файлы в папку проекта.
2. Подключаем buttons.c к проекту.
3. Инcludим заголовочный файл buttons.h к сишному файлу, в котором будут использоваться функции библиотеки.
4. Настраиваем конфигурацию и подключения в файле buttons.h
5. Прописываем в код вызов функций библиотеки.

Настройка библиотеки

Настройка конфигурации состоит из следующих шагов.

1. **Задаем количество кнопок.** Эта настройка может принимать значения от 1 до 32.

```
#define BUT_AMOUNT 4
```

2. **Задаем сколько циклов опроса должна удерживаться кнопка, чтобы она считалась нажатой.** Число должно быть меньше BUT_COUNT_HELD (смотри ниже).

```
#define BUT_COUNT_THR 10
```

3. **Задаем количество циклов опроса, в течении которых должно произойти второе нажатие на кнопку, чтобы было зафиксировано событие "двойной клик".** Желательно задавать это значение до 250, если задать выше, код будет занимать больше памяти. Если эта функция не используется, настройку можно пропустить.

```
#define BUT_COUNT_THR_2 100
```

4. **Задаем количество циклов опроса, в течении которых должна удерживаться кнопка, чтобы было зафиксировано событие "длительное удержание".** По сути - это время удержания кнопки. Желательно задавать это значение до 250, если задать выше, код будет требовать больше RAM и Flash памяти. Значение должно быть больше BUT_COUNT_THR (смотри выше).

```
#define BUT_COUNT_HELD 250
```

5. **Задаем размер буфера событий.** Его значение должно быть кратно степени двойки (2, 4, 8, 16...).

```
#define BUT_SIZE_BUF 8
```

6. **Задаем тип опроса.**

0 - полный опрос. За один вызов функции BUT_Poll() выполняется один цикл опроса всех кнопок. 1 - циклический опрос. За один вызов BUT_Poll() выполняется один цикл опроса одной кнопки. Если кнопок мало, задавайте 0.

```
#define BUT_POLL_ROTATION 0
```

7. **Задаем события, которые будут регистрироваться.** 0 - событие не регистрируется, 1 - событие будет регистрироваться, если задана индивидуальная настройка кнопки. Библиотека может регистрировать следующие события:

- нажатие на кнопку
- удержание кнопки
- отпускание кнопки после короткого удержания
- отпускание кнопки после длительного удержания
- быстрое двойное нажатие на кнопку

Исключив ненужные события, можно сэкономить немного памяти и времени микроконтроллера.

```
#define BUT_PRESSED_EN 1
#define BUT_HELD_EN 1
#define BUT_RELEASED_EN 1
#define BUT_RELEASE_LONG_EN 1
#define BUT_DOUBLE_CLICK_EN 1
```

8. Если нужно, задаем коды кнопок. Эти коды будут сохраняться в буфере событий, когда зафиксировано соответствующее событие. Коды могут принимать значения от 1 до 255. Можно задать всем событиям одинаковый код.

```
#define BUT_PRESSED_CODE 1
#define BUT_HELD_CODE 2
#define BUT_RELEASED_CODE 3
#define BUT_RELEASED_LONG_CODE 4
#define BUT_DOUBLE_CLICK_CODE 5
```

9. Задаем конфигурацию кнопок. Конфигурация кнопки включает в себя:

- Идентификационный номер - BUT_N_ID. Где N - это число от 1 до 32. Задается по порядку - 1, 2, 3 ... 32. Без пропусков!
- Порт направления - BUT_N_DDRX
- Порт подтягивающих резисторов - BUT_N_PORTX
- Порт состояния пина микроконтроллера - BUT_N_PINX
- Номер пина микроконтроллера - BUT_N_PIN
- Активный уровень кнопки - BUT_N_LEV. Если кнопка замыкается на землю - активный уровень 0. Если она замыкается на плюс питания - активный уровень 1.
- Активация pull-up резистора - BUT_N_PULL. 0 - не включать, 1 - включать подтягивающий резистор.
- Список регистрируемых событий - BUT_N_EVENT. Это события, которые будут заноситься в буфер, если они разрешены глобальными настройками. Задаются в скобках через оператор побитового ИЛИ. Список событий такой - BUT_EV_PRESSED, BUT_EV_HELD, BUT_EV_RELEASED, BUT_EV_RELEASED_LONG, BUT_EV_DOUBLE_CLICK, BUT_EV_ALL.

Пример настройки кнопки, подключенной к 0 выводу порта C. Кнопка замыкает вывод на землю, включен подтягивающий резистор и регистрируются события "отпускание кнопки" и "длительное удержание".

```
#define BUT_1_ID 1
#define BUT_1_DDRX DDRC
#define BUT_1_PORTX PORTC
#define BUT_1_PINX PINC
#define BUT_1_PIN 0
#define BUT_1_LEV 0
#define BUT_1_PULL 1
#define BUT_1_EVENT (BUT_EV_HELD | BUT_EV_RELEASED)
```

Использование библиотеки

Библиотека содержит три пользовательские функции.

void BUT_Init(void) - инициализирует внутренние переменные, настраивает кнопки. Эту функцию нужно вызывать в начале main`а.

void BUT_Poll(void) - выполняет опрос кнопок. Функция должна вызываться с периодом от 1 до нескольких десятков миллисекунд. Лучше не "совать" ее в прерывание, а вызывать ее из основного цикла программы.

В зависимости от настройки BUT_POLL_ROTATION, функция может выполнять опрос всех кнопок за раз или по одной кнопке за вызов. Если используется циклический опрос, то частота опроса кнопок будет в N раз меньше частоты вызова функции BUT_Poll(), где N - количество кнопок. Это нужно учитывать при задании настроек BUT_COUNT_THR, BUT_COUNT_THR_2, BUT_COUNT_HELD.

uint8_t BUT_GetBut(void) - возвращает событие из буфера. Функцию нужно вызывать два раза. Первый раз - возвращается код кнопки, второй раз - код события. Если буфер пустой, функция возвращает 0.

Пример использования:

```

...
uint8_t but = 0;
uint8_t code = 0;

int main( void )
{

    BUT_Init();
    ...
    __enable_interrupt();
    while(1){
        delay_ms(2);

        /*опрос кнопок*/
        BUT_Poll();

        /*проверка буфера*/
        but = BUT_GetBut();

        /*если зафиксировано событие*/
        if (but){

            /*берем код события*/
            code = BUT_GetBut();

            /*обрабатываем but и code*/
            ....

        }
    }

    return 0;
}

```

Файлы

buttons-lib.rar (/files/Buttons/buttons-lib.rar) - библиотека опроса кнопок для AVR

Писал для себя. Не проверена на 100%. Поставляется как есть.

Твитнуть



Tagged under [#библиотеки \(/index.php/programming-avr/itemlist/tag/библиотеки.html\)](/index.php/programming-avr/itemlist/tag/библиотеки.html) [#avr \(/index.php/programming-avr/itemlist/tag/avr.html\)](/index.php/programming-avr/itemlist/tag/avr.html)

Related items

- Работа с SD картой. Воспроизведение wav файла. Ч3 (/index.php/programming-avr/item/212-rabota-s-sd-kartoy-vosproizvedenie-wav-fayla-ch3.html)
- Работа с SD картой. Библиотека Petit FatFs. Ч2 (/index.php/programming-avr/item/211-rabota-s-sd-kartoy-biblioteka-petit-fatfs-ch2.html)
- Работа с SD картой. Подключение к микроконтроллеру. Ч1 (/index.php/programming-avr/item/209-rabota-s-sd-kartoy-podklyuchenie-k-mikrokontrolleru-ch1.html)
- AVR315: Использование TWI модуля в качестве ведущего I2C устройства (/index.php/programming-avr/item/208-avr315-ispolzovanie-twi-modulya-v-kachestve-veduschego-i2c-ustroystva.html)
- Библиотека для семисегментного индикатора с произвольным подключением выводов (/index.php/programming-avr/item/206-biblioteka-dlya-semisegmentnogo-indikatora-s-proizvolnym-podklyucheniem-vyvodov.html)

Комментарии (/index.php/component/jcomments/feed/com_k2/218.html)

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4010) **Alyes** 29.04.2014 08:19
Спасибо!!! то, что нужно!!!

[Ответить](#) | [Ответить с цитатой](#) | [Цитировать](#)

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4012) **foxit** 29.04.2014 14:19
Спасибо.

[Ответить](#) | [Ответить с цитатой](#) | [Цитировать](#)

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4013) **Pashgan** 29.04.2014 21:01
Пожалуйста.

[Ответить](#) | [Ответить с цитатой](#) | [Цитировать](#)

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4015) **Andy** 30.04.2014 04:46
Отлично, как всегда.
У меня есть вопрос. Какой самый оптимальный алгоритм опроса кнопок в диспетчере задач? То есть запускаем задачу опроса кнопок раз, допустим в 100 миллисекунд, а дальше что? Если нажата, перезапускаем снова себя же через, ну, скажем, 50 миллисекунд (защита от дребезга) и если кнопка уже нажата или отжата делаем выводы, ставить флаг нажатой кнопки или еще нет, когда и где сбрасывать этот флаг? Прерывания использовать нельзя. Подскажите толковый алгоритм, пожалуйста.

Если важно, какой диспетчер - то <http://andromega.narod.ru/publics.html>
"Компактный диспетчер для встроенных систем"

[Ответить](#) | [Ответить с цитатой](#) | [Цитировать](#)

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4019) **Pashgan** 30.04.2014 20:58
Не знаю какой алгоритм оптимальный. Никогда не заморачивался с этой темой. Я уже давно использую один и тот же алгоритм - <http://chipenable.ru/index.php/programming-avr/item/38-uchebnyy-kurs-opros-knopok.html>.

[Ответить](#) | [Ответить с цитатой](#) | [Цитировать](#)

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4020) **Peter** 01.05.2014 07:50
Посмотрел BUT_Init и сразу непонятно. Как так можно в цикле по "КоличествуКнопок" BUT_AMOUNT+1 обнулять совершенно независимый "БуферСобытий" buf[]? А если количество кнопок будет > буфера событий ? Что мы тогда наобнуляем?

[Ответить](#) | [Ответить с цитатой](#) | [Цитировать](#)

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4021) **Pashgan** 01.05.2014 08:15
Исправил.

[Ответить](#) | [Ответить с цитатой](#) | [Цитировать](#)

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4043) **demiurg** 02.05.2014 15:42
Много буквато. Непонятный алгоритм. Понятно, что хочется удобств. Привязка к таблице резонна, если один автомат. У меня за долгое время прижился следующий алгоритм. Кстати, кое-что взято с цикла статей Татарчевского об автоматах. Сканирование. Здесь, в зависимости от реализации, считывание порта, сканирование матрицы, буфера дискретных входов. Если у нас в проекте только нажатие, то все гораздо проще. Переменные _kbd_drv, kbd_drv_flags, keys_curr, keys_prev. В функции сканирования сброс флага KEYS_PRESSED_FLG, сканирование, если есть нажатие, установка флага... М-м-м, проще кодом. Ссылка: <http://pastebin.com/Vgz2ZyLT>

[Ответить](#) | [Ответить с цитатой](#) | [Цитировать](#)

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4049) **Pashgan** 02.05.2014 18:45
У каждой кнопки есть свой регистр состояния и свой счетчик. При нажатии на кнопку счетчик начинает считать. Как только он досчитал до первого порога, фиксируется событие "нажатие" и устанавливается флаг. Как только он досчитал до второго порога, фиксируется событие "длительное удержание" и устанавливается флаг. Когда кнопка отпускается проверяется флаги регистра состояния. В одном случае фиксируется событие "отпускание после короткого удержания", в другом случае - "отпускание после длительного удержания". Если используется двойной клик, то добавляется второй счетчик, который считает время прошедшее между двумя нажатиями. Если это время меньше порогового, то фиксируется событие "двойной клик". Вот и весь алгоритм.

[Ответить](#) | [Ответить с цитатой](#) | [Цитировать](#)

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4044) **virustek** 02.05.2014 17:24
Пытаюсь использовать библиотеку в Atmel Studio 6.2 выдает ошибку, что undefined reference to 'BUT_Init', итак со всеми вызываемыми функциями.

Код:

```
//#include <stdio.h>
#define F_CPU 8000000UL
#include <util/delay.h>
#include <avr/interrupt.h>
//#include <stdint.h>
#include <button/buttons.h.h>

uint8_t but = 0;
uint8_t code = 0;

int main( void )
{
    BUT_Init();
    DDRB=0x0e; //Инициализация портов, OC1A,OC1B,OC2 - выходы
    sei();//__enable_interrupt();
    while(1){
        _delay_ms(2);

        /*опрос кнопок*/
        BUT_Poll();

        /*проверка буфера*/
        but = BUT_GetBut();

        /*если зафиксировано событие*/
        if (but){

            /*берем код события*/
            code = BUT_GetBut();

            if (code == 1)
            {
                PORTB = (1<<PB1)|(0<<PB2)|(0<<PB3);
            }
            else if (code == 2)
            {
                PORTB = (0<<PB1)|(1<<PB2)|(0<<PB3);
            }
            else if (code == 3)
            {
                PORTB = (0<<PB1)|(0<<PB2)|(1<<PB3);
            }

        }

        return 0;
    }
}
```

в чем проблема может быть?

[Ответить](#) | [Ответить с цитатой](#) | [Цитировать](#)

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4050) **Pashgan** 02.05.2014 18:48
Здесь неправильно

Код:

```
#include <button/buttons.h.h>
```

[Ответить](#) | [Ответить с цитатой](#) | [Цитировать](#)

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4046) **demiurg** 02.05.2014 17:26
button/buttons.h.h

Ответить | Ответить с цитатой | Цитировать

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4057) **virustek** 02.05.2014 20:45
Это моя опечатка, пытался переобозвать файл и скинул не ту версию кода, поправил как надо, не подключает в хедеде buttons.h библиотеку
#include , компилятор просто отказывается показать путь в нее, хотя из той же папки interrupt.h видит прекрасно, пока не смог подключить.
Ответить | Ответить с цитатой | Цитировать

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4058) **virustek** 02.05.2014 20:47
*дополнение: не подключает #include

Ответить | Ответить с цитатой | Цитировать

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4059) **virustek** 02.05.2014 20:51
да блин

Код:

```
#include <avr/io.h>
```

Ответить | Ответить с цитатой | Цитировать

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4060) **virustek** 02.05.2014 21:21
все так же:" Error 1 undefined reference to `BUT_Init()'" Для всех вызываемых функций.

Ответить | Ответить с цитатой | Цитировать

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4061) **Pashgan** 02.05.2014 21:34
Напиши на форум и приложи проект. Вот ветка <http://chipenable.ru/index.php/forum/materialy-sajta-chipenable/3347-biblioteka-dlya-oprosa-knopok.html> (<http://chipenable.ru/index.php/forum/materialy-sajta-chipenable/3347-biblioteka-dlya-oprosa-knopok.html>)
Ответить | Ответить с цитатой | Цитировать

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4728) **Rosin55** 30.01.2016 07:54
Нужно добавить файл библиотеки в проект - PKM/ add extern items.

Ответить | Ответить с цитатой | Цитировать

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4051) **demiurg** 02.05.2014 18:49
В приведенном мной примере очень простая концепция. Все кнопки, сколько бы их не было, обрабатываются как одна. Вы что, для ста кнопок будете 10 регистров отводить?

Ответить | Ответить с цитатой | Цитировать

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4055) **Pashgan** 02.05.2014 19:12
Да и у меня тоже простая концепция. Один алгоритм на все кнопки. Проверяется нажата она или нет и ее состояние передается в функцию, а та уже все анализирует. Я просто не знаю, как можно подсчитывать время нажатия нескольких кнопок, используя только один счетчик. Знаю один алгоритм, но он не позволяет обрабатывать одновременные нажатия.

Ответить | Ответить с цитатой | Цитировать

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4052) **demiurg** 02.05.2014 18:50
поправка 100 регистров

Ответить | Ответить с цитатой | Цитировать

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4054) **Pashgan** 02.05.2014 18:57
Это зависит от задачи. Для 18-и кнопок я отвел 36 байт в ОЗУ и что. У меня 4 кБ.

Ответить | Ответить с цитатой | Цитировать

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4076) **WithLove** 08.05.2014 08:07
"5. Задаем размер буфера событий. Его значение должно быть кратно 2. "

Например, 6?)) И что тогда будет при
head &= (BUT_SIZE_BUF - 1);

Ответить | Ответить с цитатой | Цитировать

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4084) **ibiza11** 16.05.2014 06:22
Автор уже поправил пост.

Цитата:

Его значение должно быть кратно степени двойки

Можно заменить на "кратно" на "равно"

Ответить | Ответить с цитатой | Цитировать

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4253) **Ersafap** 20.07.2014 23:07
Здравствуйте. Попробовал применить либу в таймере. Сбоит. Например если я жму несколько раз одну кнопку а после жму другую, то действие выполняется для первой кнопки(один раз), хотя код второй я даже не обрабатываю. Последующие нажатия на вторую кнопку эффекта не дают. Если после этого снова нажать первую кнопку то можно будет снова нажать 1 раз вторую и она обработается как первая.

Ответить | Ответить с цитатой | Цитировать

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4284) **rusin.aleks** 09.09.2014 14:50
Что такое #define End_m)?

Ответить | Ответить с цитатой | Цитировать

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4303) **stas** 28.09.2014 15:34
Добрый день, как подключить си файл и библиотеку atmel studio 6?
Инклюд прописал, сишный файл в проект добавил, адрес компилятору указал.

Ответить | Ответить с цитатой | Цитировать

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4327) **tipok** 27.10.2014 03:42
Добрый день.. а можно посмотреть код с реализацией удержания кнопки? чота не как не пойму ка это работает.
Ответить | Ответить с цитатой | Цитировать

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4493) **Дима** 12.04.2015 15:18
Error 1 'DDRC' undeclared (first use in this function) C:\Users\Dmitry\Desktop\avrstud\FF\FF\buttons.c 255 3 FF
Error 3 'PORTC' undeclared (first use in this function) C:\Users\Dmitry\Desktop\avrstud\FF\FF\buttons.c 255 3 FF
Error 4 'PINC' undeclared (first use in this function) C:\Users\Dmitry\Desktop\avrstud\FF\FF\buttons.c 396 1 FF

Ответить | Ответить с цитатой | Цитировать

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4494) **Дима** 12.04.2015 15:22
Разобрался.. У меня нет просто порта C

Ответить | Ответить с цитатой | Цитировать

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4504) **MrOops** 26.04.2015 13:43
/*макрос проверки состояния пина. зависит от
активного уровня в настройках*/
#define _TestBit1(var, bit) ((var) & (1

Ответить | Ответить с цитатой | Цитировать

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4505) **MrOops** 26.04.2015 13:45
Код:

```
#define _TestBit(var, bit, lev) _TestBit##lev(var, bit)
#define TestBitLev(var, bit, lev) _TestBit(var, bit, lev)
```

_TestBit##lev(var, bit) как это работает

Ответить | Ответить с цитатой | Цитировать

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4544) **Артём Б** 16.07.2015 15:16
День добрый!!! Сделал всё как написано в статье, а кодевижен мне выдаёт : Error: C:\cvavrevall\inc\buttons.h(17), included from: òàéìäð 2.c: can't
open #include file: ioavr.h что подскажите?

Ответить | Ответить с цитатой | Цитировать

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4545) **demiurg** 16.07.2015 16:11
У кодевижн свои файлы \IO. Ищите какой файл кодевижен соответствует определениям \IO AVR.

Ответить | Ответить с цитатой | Цитировать

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4546) **demiurg** 16.07.2015 16:14
Вообще, если честно, мне эта библиотека не очень. Есть отличный цикл статей Татарчевского. И там в подробностях расписано как
реализовать опрос кнопок.
www.kit-e.ru/articles/circuit/2006_11_164.php (http://www.kit-e.ru/articles/circuit/2006_11_164.php)

Ответить | Ответить с цитатой | Цитировать

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4550) **VladCh** 29.07.2015 04:18
У Татарчевского все как-то заумно написано, чтобы понять надо вчитываться. Здесь же - все просто и понятно.
Может у него и правильней, но для тех, кто занимается программированием в бытовых целях (это я про себя :)) изучать всю теорию КА смысла
нет. Здесь же все просто, понятно и доходчиво объяснено с реальными примерами, за что автору низкий поклон.
В общем, как говорится, каждый выбирает для себя...

Ответить | Ответить с цитатой | Цитировать

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4551) **demiurg** 29.07.2015 06:26
Цитирую VladCh:

...

1 - Ничего заумного нет.
2 - КА решают самую главную проблему - псевдопараллельность процессов.

Ответить | Ответить с цитатой | Цитировать

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4572) **DVF** 21.08.2015 11:20
А если надо выполнить действие по двум одновременно нажатым кнопкам? Или надо, чтобы при нажатой кнопке нажатие другой не оказывало
влияния? Библиотека позволяет это реализовать?

Ответить | Ответить с цитатой | Цитировать

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4733) **ay8910** 04.02.2016 05:05
Отличная библиотека, грамотный, правильный код С. Без особых усилий портировалась под SDCC и используется мною на STM8. Спасибо!
Ответить | Ответить с цитатой | Цитировать

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4734) **Mik** 06.02.2016 23:47
Здравствуйте.
Пользуюсь Atmel Studio 7.0
Судя по тому, что получается, код библиотеки не выполняется вообще. Такой вывод делаю из того, что даже подтяжка на ножке не включается.
Думаю, что накосячил в пункте 2. Подключаем buttons.c к проекту.
В Solution Explorer я кликаю правой кнопкой по названию проекта, далее Add -> Existing Item -> Buttons.c, который лежит в папке, в которой
лежит main.c, то есть исходный файл проекта. Я не правильно понял, как подключать файл?

Ответить | Ответить с цитатой | Цитировать

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4752) **Mic29** 14.03.2016 09:46
Собственно решал похожую задачку. Куча датчиков и кнопок (от 24 до 32) подключены через сдвиговые регистры 74HC597. Объявлены четыре
массива по 32 эл-та, в одном задается какой сигнал (инверсный или прямой), во втором - кол-во считываний входа для решения о
достоверности сигнала, в третьем - результат, четвертый - статический в процедуре опроса. Считывается все подряд, при необходимости
инвертируется. И дальше - если есть сигнал то в стат. массиве +=1 (но не больше числа заданного в массиве 2), нет сигнала -=1 (но не меньше
0). Сразу же смотрим, если набрали требуемое кол-во считываний, записываем в третий массив. Соответственно, за число проходов
определенное во втором массиве для каждого конкретного входа (независимо) или досчитает или - нет. А в третьем массиве - результат.
Остается только интерпретировать. Можно в глоб. цикле опрашивать, можно по прерываниям.

Ответить | Ответить с цитатой | Цитировать

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4851) **Ternity** 21.06.2016 03:58
Вот так работать не будет. Запустится, но ничегошеньки...

```
#define BUT_PRESSED_CODE 0
#define BUT_HELD_CODE 2
#define BUT_RELEASED_CODE 3
#define BUT_RELEASED_LONG_CODE 4
#define BUT_DOUBLE_CLICK_CODE 1
```

[Ответить](#) | [Ответить с цитатой](#) | [Цитировать](#)

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4853) **KentAVR** 21.06.2016 04:30
В м128 PORTF находится в расширенной области PBB
Так, получается кнопку приделать к PF0, но не PF4, PF6...

[Ответить](#) | [Ответить с цитатой](#) | [Цитировать](#)

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4854) **KentAVR** 21.06.2016 04:42
В Atmega128 не работала кнопка на PF6, потому что JTAG был включён. Вырубил фьюзом, полёт нормальный. Спасибо за либу!

[Ответить](#) | [Ответить с цитатой](#) | [Цитировать](#)

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-4920) **SAnyo** 03.10.2016 06:10
Здравствуйте. CVAVR ругается на строку
but = BUT_GetBut();
Error: a value of type 'const void' can't be assigned to an entity of type 'uint8_t'
В чем может быть причина? Спасибо.

[Ответить](#) | [Ответить с цитатой](#) | [Цитировать](#)

(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#comment-5151) **Андрей** 24.08.2017 11:43
Можно ли использовать эту библиотечку с Arduino IDE ?

[Ответить](#) | [Ответить с цитатой](#) | [Цитировать](#)

[Обновить список комментариев](#)

[RSS лента комментариев этой записи \(/index.php/component/jcomments/feed/com_k2/218.html\)](/index.php/component/jcomments/feed/com_k2/218.html)

Добавить комментарий

Имя (обязательное)

E-Mail



Обновить

Отправить

JComments (<http://www.joomlatune.ru>)

[вернуться наверх \(/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#startOfPageId218\)](/index.php/programming-avr/item/218-biblioteka-dlya-oprosa-knopok.html#startOfPageId218)