

## **ИМС 1887BE7T**

### **1. Перечень отличий ИМС от Atmega128**

№	Описание	Комментарии
1	Тип использованной памяти: EEPROM	Стертое значение памяти – 00
2	Размер буфера страницы области RWW: 128 слов	
3	Размер буфера страницы области NRWW: 32 слова	
4	Длительность команды ChipErase	1,5 сек

### **2. Память программ**

Используется многократно программируемая память типа EEPROM, объемом 64Kx16. Количество циклов перезаписи – 100 000.

Состояние ячеек памяти после команд стирания – 0. Запись можно производить как постранично, размер страницы 128 (для области NRWW - 32) 16-разрядных слов, так и пословно (запись одного 16-разрядного слова). Режим записи одного слова полезен, в случае самопрограммирования контроллера. Если необходимо изменить одно или несколько слов на страницы памяти программ, то нет необходимости производить предварительное считывание и сохранение в буфере не изменяемых данных со страницы памяти. Подробно описание данной возможности описано в разделе 2.2.

#### **2.1 Управление процессом самопрограммирования.**

Изменение содержимого памяти программ и ячеек защиты загрузчика осуществляется с помощью команды SPM (Store Program Memory). Параметрами этой команды являются адрес области памяти, загружаемый предварительно в индексный регистр Z и нулевой бит регистра RAMPZ, данные, находящиеся в регистровой паре R1:R0.

Управление процессом программирования и, в частности, определение операции, выполняемой при вызове команды SPM, осуществляется с помощью регистра ввода/вывода SPMCR.

Для адресации памяти программ при использовании команды SPM используются индексный регистр Z, получаемый объединением двух регистров общего назначения - R30 (младший байт) и R31 (старший байт) и нулевой бит регистра RAMPZ.

Адресация памяти программ с использованием команды SPM показана на рисунке 2.1.

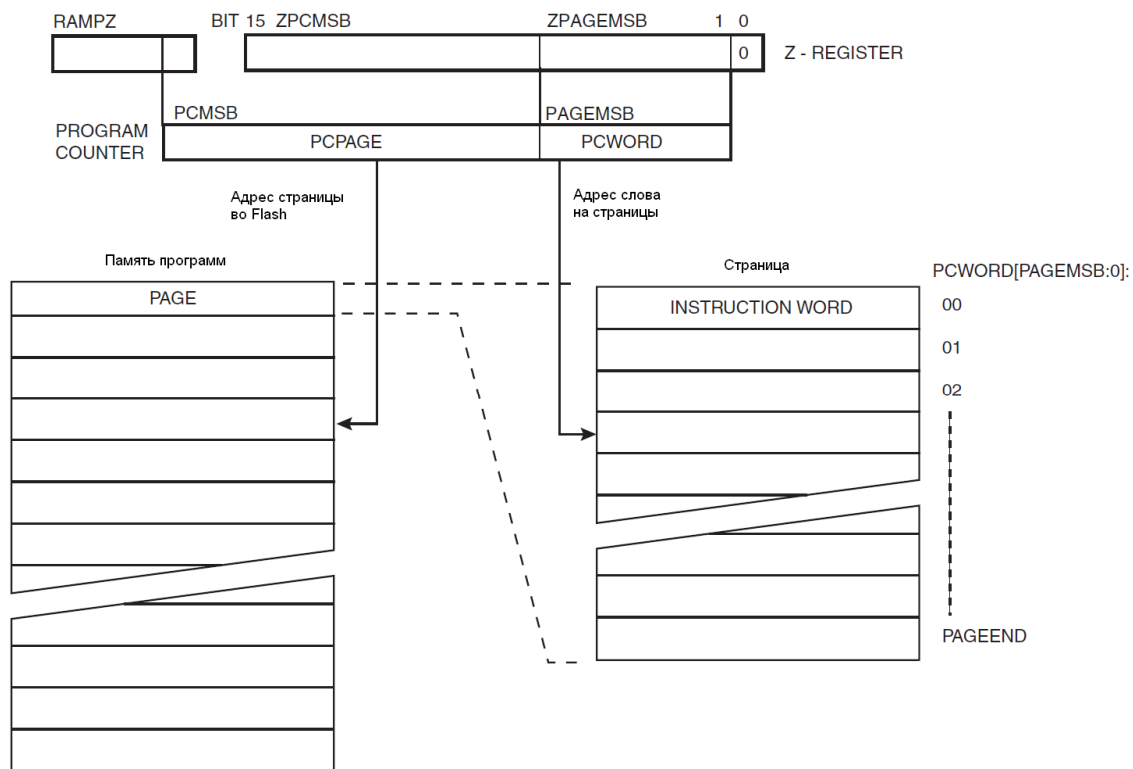


Рисунок 2.1. Адресация памяти программ при использовании команды SPM

## 2.2 Изменение памяти программ.

Изменение памяти программ осуществляется в следующей последовательности:

### 1. Заполнение буфера страницы новым содержимым:

Индексный регистр Z используется для адресации памяти программ. Биты 1-7 отвечают за адресацию слова в буфере, а биты 6-15 и нулевой бит регистра RAMPZ за адресацию страницы в памяти программ. Данные кода операции должны находиться в регистрах R1:R0.

Изменение страницы памяти программ возможно как целиком всю страницу, так и любое слово на странице, необходимо лишь загрузить адрес слова в буфер страницы.

Загружаем адрес слова на странице и адрес страницы в памяти программ в индексный регистры Z и RAMPZ. Выполняем команду SPM с установленным битом SP MEN. Загружаем следующий адрес слова в регистры Z и RAMPZ. Выполняем команду SPM с установленным битом SP MEN. Повторяем данные операции до заполнения буфера или необходимого количества слов (если требуется изменить только часть страницы).

### 2. Перенос содержимого буфера в память программ:

После загрузки последнего слова в буфер страницы выполняем команду SPM с установленными одновременно битами SP MEN и одним из битов PGWRT или PGERS, в зависимости от того, что необходимо сделать, либо записать, либо стереть данные на странице, при этом адрес страницы в памяти программ должен быть занесен в регистры Z и RAMPZ (секция PCPAGE).

Следует заметить, что при записи данных в память программ необходимость в стирании страницы отсутствует, поэтому процедуру стирания страницы перед записью данных, можно пропустить.

При стирании страницы памяти программ содержимое регистров R1 и R0 игнорируется.

При записи страницы или слова в память программ в регистрах R1 и R0 содержится код операции.

Для определения момента окончания выполнения операций можно либо опрашивать состояние флага SP MEN регистра SPMCR, дожидаясь его сброса, либо воспользоваться прерыванием «Готовность SPM». Это прерывание генерируется все время, пока флаг SP MEN сброшен. В последнем случае таблица векторов прерываний должна находиться в секции загрузчика, а это прерывание должно быть разрешено установкой флага SPMIE регистра SPMCR.

## Пример кода на ассемблере

### Запись страницы памяти программ данными из ОЗУ

```
;*****
ldi    XH,high(RAMSTART)      ;Для адресации ОЗУ используем указатель X
ldi    XL,low(RAMSTART)
clr    ZH                      ;указатель flash
clr    ZL
clr    count                   ;счетчик записанных слов
ldi    temp1,PROG_SIZE        ;размер программы в словах

save_rww:
    clr    looplo              ;счетчик слов на странице

;передаем данные из ОЗУ в буфер страницы

load_buffer:
    ld     r0,X+
    ld     r1,X+
    ldi    spmcrval,(1<<SPMEN)
    rcall  Do_spm
    inc    count
    inc    looplo
    adiw   ZH:ZL,2
    cpi    looplo,$20
    brne   load_buffer
    sbiw   ZH:ZL,2

;записываем страницу

    ldi    spmcrval,(1<<PGWRT)|(1<<SPMEN)
    rcall  Do_spm
    adiw   ZH:ZL,2
    cp     count,temp1
    brlo   save_rww
;*****
```

### Запись слова памяти программ данными из ОЗУ

```
;*****
ldi    XH,high(RAMSTART)      ;Для адресации ОЗУ используем указатель X
ldi    XL,low(RAMSTART)
clr    ZH                      ;указатель flash
clr    ZL
clr    count                   ;счетчик записанных слов
ldi    temp1,PROG_SIZE        ;размер программы в словах

save_word:
    ld     r0,X+
    ld     r1,X+
    ldi    spmcrval,(1<<SPMEN)
    rcall  Do_spm
    ldi    spmcrval,(1<<PGWRT)|(1<<SPMEN)
    rcall  Do_spm
    inc    count
    adiw   ZH:ZL,2
    cp     count,temp1
    brne   save_word
;*****
```

### 3. Память данных

Используется многократно программируемая память типа EEPROM, объемом 4Кх8. Количество циклов перезаписи – 100 000. Запись можно производить как постранично, размер страницы 32 байта, так и побайтно.

### 4. ISP и параллельное программирование 1887BE7T из среды AVRstudio

В микросхеме 1887BE7T отличается размер буфера области загрузчика (NRWW) от ATmega128. Для программирования полного объема памяти программ посредством программаторов STK500, STK600, AVRISP, JTAGICE MkII из среды AVRstudio необходимо добавить файл описания МК 1887BE7T.xml в папку PartDescriptionfiles программы AVRstudio.

*Пример:* c:\Program Files\Atmel\AVR Tools\Partdescriptionfiles\

При программировании выбирать данный тип МК.

Стертое значение памяти отличается от аналога, поэтому автоматическая верификация после программирования (Verify device after programming) в режиме User current Simulator/Emulator FLASH Memory будет выдавать ошибку, также будет ошибка программирования при установке флага автоматического стирания перед программированием (Erase device before flash programming) из за разницы длительности стирания командой Chiperase. Необходимо эти флаги убирать при программировании рисунок 3.1.

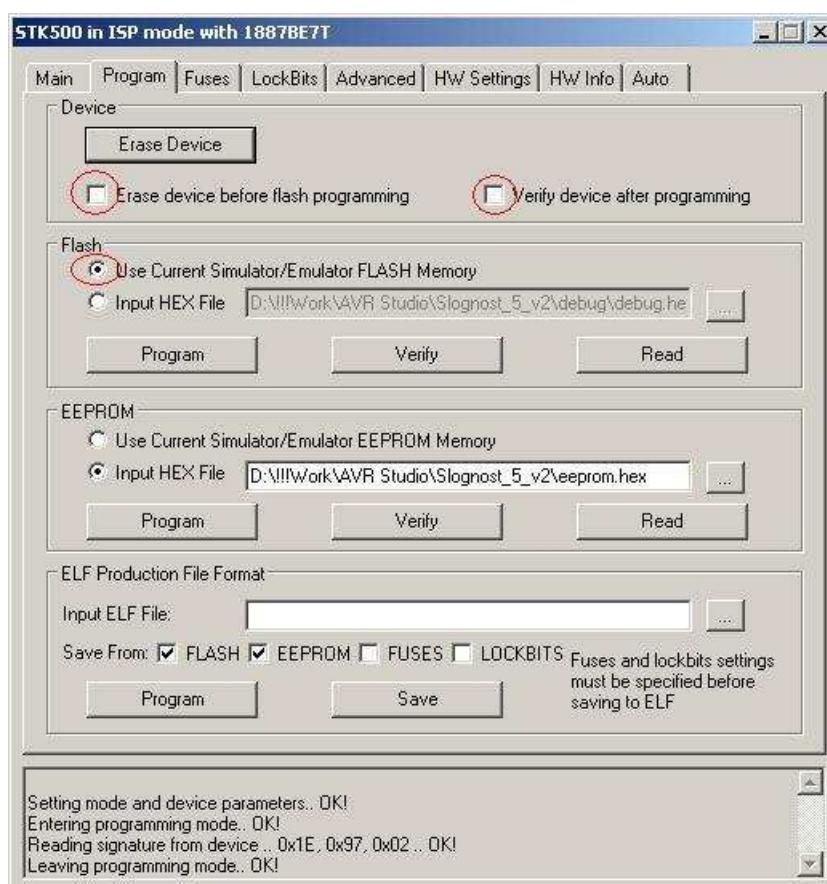


Рисунок 3.1 Окно программы AVRprog AVRstudio 4.19

Программирование в режиме Input HEX File (выбирается готовый HEX файл) поддерживает автоматическую верификацию после программирования рисунок 3.2

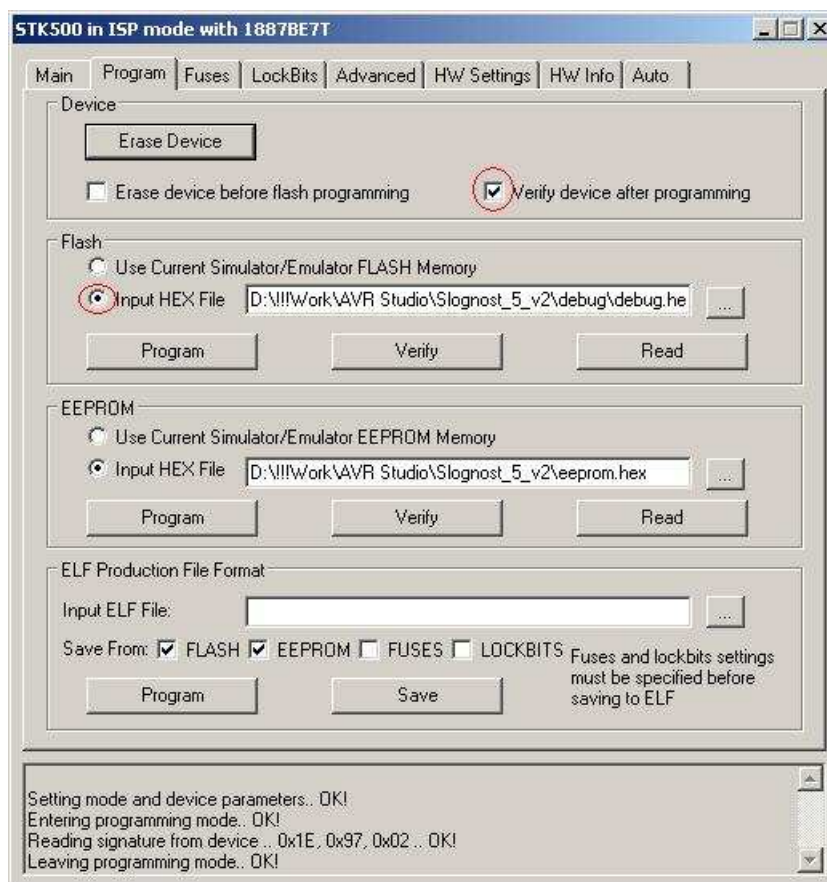


Рисунок 3.2 Окно программы AVRprog AVRstudio 4.19

## 5. Программирование и внутрисхемная отладка через интерфейс JTAG

Программирование МК через интерфейс JTAG с помощью программаторов STK600 поддерживается полностью. При программировании выбирать МК 1887BE7T во вкладке Main программатора AVRprog AVRstudio.

При выполнении команды Erase Device программное обеспечение может выдать ошибку, образец показан на рисунке 5.1

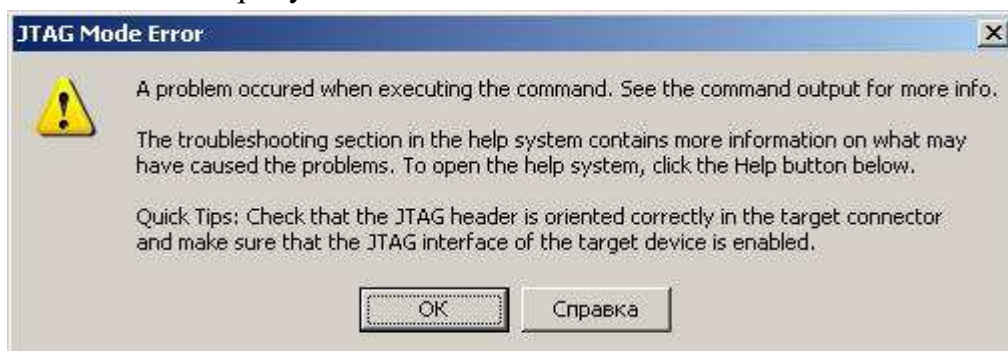


Рисунок 5.1 Ошибка команды Erase Device

Ошибка связана с различной длительностью процесса стирания кристалла (Chip Erase), программатор опрашивает состояние контроллера в течении 200 мс, а фактическое время стирания кристалла составляет 1500 мс. На сам процесс это не влияет, стирание произойдет полностью.

Для программирования микроконтроллера через порт JTAG с помощью JTAGICE и JTAGICE MkII необходимо использовать переходник КФДЛ.441461.009. Внешний вид представлен на рисунке 5.2. При использовании JTAGICE3 переходник не нужен.



Рисунок 5.2 Переходник для JTAGICE КФДЛ.441461.009

При использовании внутрисхемной отладки в МК 1887BE7T существует ряд отличий от аналога (ATmega128).

Проект создаем на ATmega128. В случае с аналогом после компиляции программы в среде AVR Studio, происходит автоматическое стирание кристалла и запись в него нового кода. Так как, длительность команды ChipErase в МК 1887BE7T отличается от аналога (ATmega128), то автоматическая запись обновленного кода программы не пройдет. Поэтому необходимо, сначала запрограммировать микроконтроллер, а потом войти в режим отладки.

Для того, что бы не происходило автоматическое обновление прошивки МК при каждой компиляции необходимо в настройках AVR Studio снять соответствующий флаг, как показано на рисунке 5.2. В меню *Debug / JTAG ICE Options* Alt + O

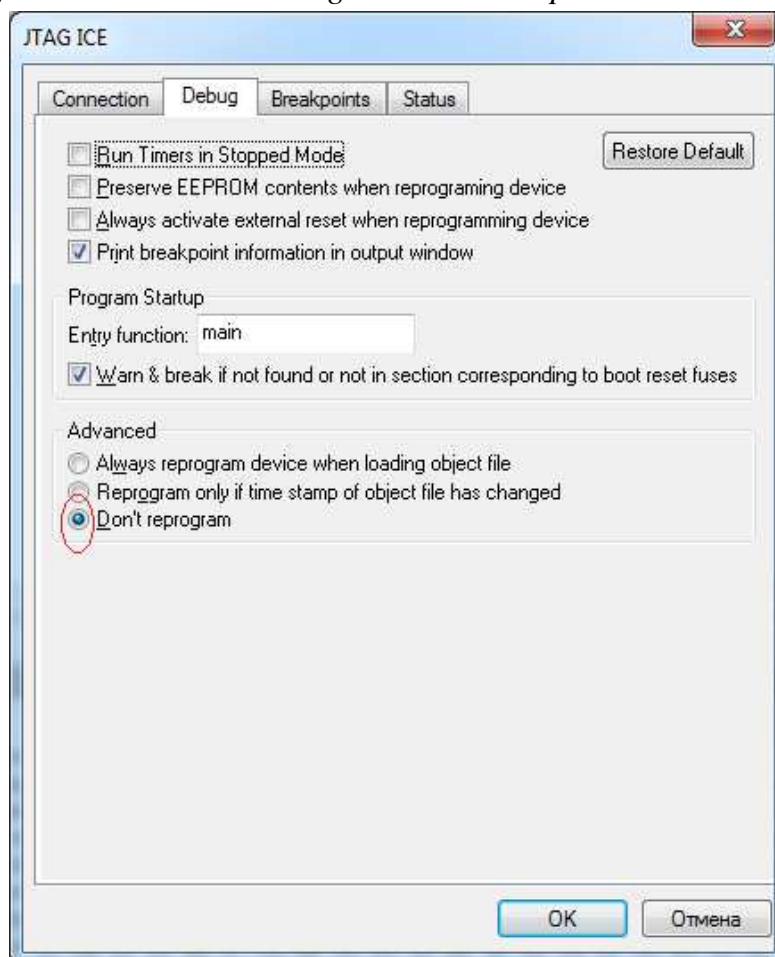


Рисунок 5.2 Настройка JTG ICE Options

## 6. Программирование 1887BE7T из среды Atmel Studio 6

Для программирования МК посредством программаторов STK500, STK600, AVRISPMkII, JTAGICEmkII, JTAGICE3 из среды Atmel Studio 6 необходимо добавить файл описания МК 1887BE7T.xml в папку devices программы Atmel Studio:

*Пример:* c:\Program Files\Atmel\Atmel Studio 6.2\devices\

Также необходимо скопировать в папку tools xml-файлы для различных отладочных средств:

*Пример:* c:\Program Files\Atmel\Atmel Studio 6.2\tools\STK500\xml\

c:\Program Files\Atmel\Atmel Studio 6.2\tools\STK600\xml

c:\Program Files\Atmel\Atmel Studio 6.2\tools\JTAGICEmkII\xml\

c:\Program Files\Atmel\Atmel Studio 6.2\tools\JTAGICE3\xml\

c:\Program Files\Atmel\Atmel Studio 6.2\tools\AVRISPMkII\xml\

При создании проекта в Atmel Studio 6 необходимо выбирать МК ATmega128.

При программировании различными средствами необходимо выбирать МК 1887BE7T!

При программировании необходимо учесть тот факт, что длительность стирания МК 1887BE7T больше чем у аналога (ATmega128), соответственно программирование микроконтроллера с автоматическим предварительным стиранием не будет выполняться. Необходимо убрать флаг *Erase device before programming*. На рисунке 6.1 показан данный флаг.

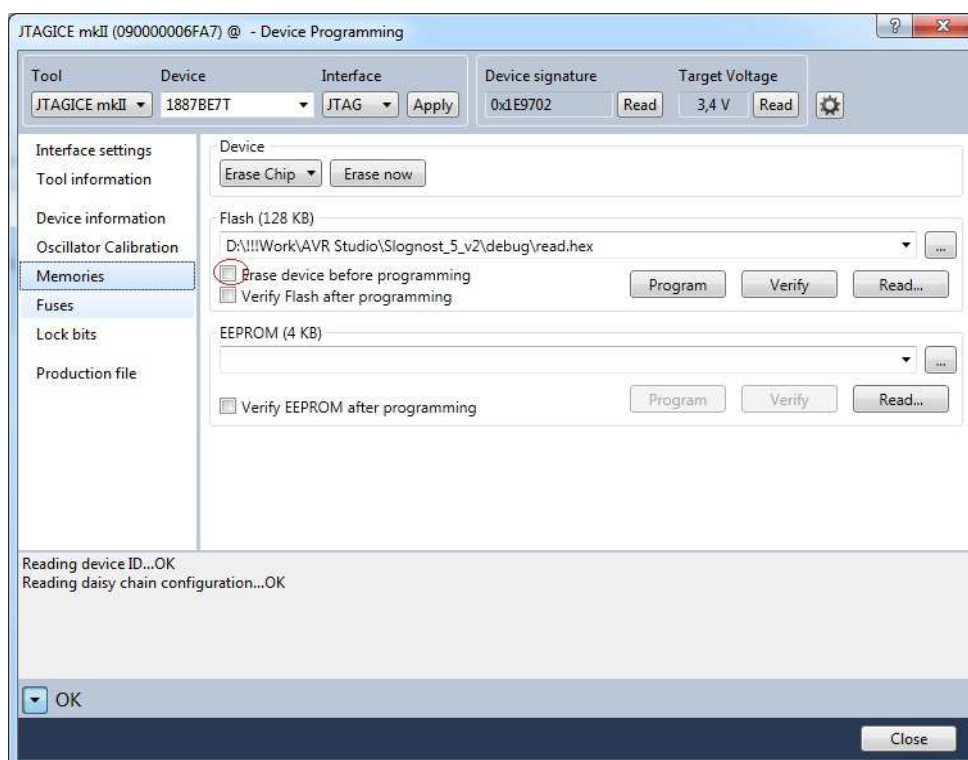


Рисунок 6.1 Окно Device Programming среды Atmel Studio 6