

Перевод с английского Ю. Андриенко  
FidoNet - 2:461/44.14, 2:461/42.14  
E-Mail - yurik@aqua.kharkov.ua  
Телефон - (057-2) 98-07-58, 44-85-40

Редактирование и вёрстка Александра Труш

<http://trush.da.ru/>

<http://avr.da.ru/>

Версия 1.0

**МИКРОКОНТРОЛЛЕР AT90S1200 ФИРМЫ ATMEL**

# СОДЕРЖАНИЕ

<b>МИКРОКОНТРОЛЛЕР AT90S1200 ФИРМЫ ATMEL .....</b>	<b>3</b>
<b>ОПИСАНИЕ ВЫВОДОВ .....</b>	<b>3</b>
<b>КВАРЦЕВЫЙ ГЕНЕРАТОР.....</b>	<b>4</b>
<b>ВСТРОЕННЫЙ RC ГЕНЕРАТОР.....</b>	<b>4</b>
<b>RISC ПРОЦЕССОР МИКРОКОНТРОЛЛЕРА AT90S1200.....</b>	<b>4</b>
Обзор архитектуры .....	4
Файл регистров общего назначения .....	5
Арифметико-логическое устройство - АЛУ .....	5
Загружаемая память программ.....	5
Режимы адресации программ и данных.....	5
Стек подпрограмм и прерываний .....	5
EEPROM память данных .....	5
Время выполнения команд .....	5
Память ввода/вывода.....	6
Регистр состояния - SREG.....	7
Сброс и обработка прерываний .....	7
Источники сброса .....	8
Сброс по включению питания .....	8
Внешний сброс.....	8
Сброс от сторожевого таймера .....	8
Обработка прерываний .....	8
Внешние прерывания.....	9
Время реакции на прерывание.....	10
Регистр управления микроконтроллером .....	10
Режимы пониженного энергопотребления .....	10
Режим холостого хода .....	11
Экономичный режим.....	11
Таймер/счетчик .....	11
Сторожевой таймер .....	12
Доступ к энергонезависимой памяти.....	12
Аналоговый компаратор .....	13
Порты ввода/вывода .....	14
Порт В.....	14
Порт D .....	15
<b>ПРОГРАММИРОВАНИЕ ПАМЯТИ.....</b>	<b>17</b>
Программирование битов блокировки памяти .....	17
Биты конфигурации (Fuse bits).....	17
Код устройства .....	17
Программирование флэш памяти и EEPROM.....	17
Параллельное программирование .....	17
Последовательная загрузка .....	21
<b>МАКСИМАЛЬНО ДОПУСТИМЫЕ ПАРАМЕТРЫ.....</b>	<b>22</b>
<b>ХАРАКТЕРИСТИКИ ПО ПОСТОЯННОМУ ТОКУ.....</b>	<b>22</b>
<b>ПАРАМЕТРЫ ВНЕШНЕГО ТАКТОВОГО СИГНАЛА.....</b>	<b>23</b>
<b>ИНФОРМАЦИЯ ДЛЯ ЗАКАЗА.....</b>	<b>23</b>
<b>СПИСОК РЕГИСТРОВ.....</b>	<b>24</b>
<b>НАБОР КОМАНД AT90S1200 .....</b>	<b>25</b>

# Микроконтроллер AT90S1200 фирмы Atmel

AT90S1200 - экономичный 8 битовый КМОП микроконтроллер, построенный с использованием расширенной RISC архитектуры AVR. Устройство имеет 64 байта EEPROM. Исполняя по одной команде за период тактовой частоты, AT90S1200 имеет производительность около 1MIPS на МГц, что позволяет разработчикам создавать системы оптимальные по скорости и потребляемой мощности.

В основе ядра AVR лежит расширенная RISC архитектура, объединяющая развитый набор команд и 32 регистра общего назначения. Все 32 регистра непосредственно подключены к арифметико-логическому устройству (ALU), что дает доступ к любым двум регистрам за один машинный цикл. Подобная архитектура дает десятикратный выигрыш в эффективности кода по сравнению с традиционными CISC микроконтроллерами.

Подобная архитектура эффективно поддерживает языки высокого уровня и дает высокую плотность ассемблерных программ. AT90S1200 предлагает следующие возможности: 1кБ загружаемой флэш памяти; 64 байта EEPROM; 15 двунаправленных линий ввода/вывода; 32 регистра общего назначения; настраиваемый таймер/счетчик; внешние и внутренние прерывания; программируемый сторожевой таймер со встроенным генератором; SPI последовательный порт для загрузки программ; два выбираемых программно режима низкого энергопотребления.

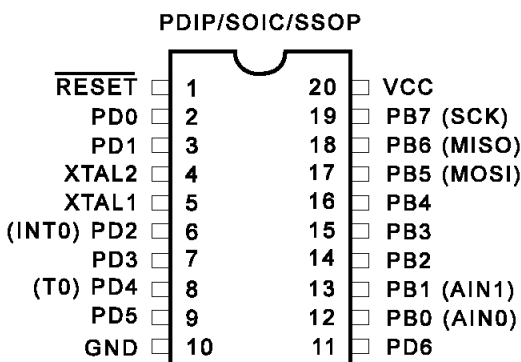
Холостой режим (Idle Mode) отключает ЦПУ, оставляя в рабочем состоянии регистры, таймер/счетчик, сторожевой таймер и систему прерываний.

Экономичный режим (Power Down Mode) сохраняет содержимое регистров, но отключает генератор, запрещая функционирование всех встроенных устройств до внешнего прерывания или аппаратного сброса.

Микросхемы производятся с использованием технологии энергонезависимой памяти высокой плотности фирмы Atmel. Загружаемая флэш память на кристалле может быть перепрограммирована прямо в системе через последовательный интерфейс SPI или доступным программатором энергонезависимой памяти. Объединяя на одном кристалле усовершенствованный 8-битовый RISC процессор с загружаемой флэш памятью, AT90S1200 является мощными микроконтроллерами, которые позволяют создавать достаточно гибкие и эффективные по стоимости устройства.

AT90S1200 поддерживается полной системой разработки включающей в себя макроассемблер, программный отладчик/симулятор, внутрисхемный эмулятор и отладочный комплект.

## ОПИСАНИЕ ВЫВОДОВ



VCC - вывод источника питания

GND - земля

Port B (PB7..PB0) - Порт В является 8-битовым двунаправленным портом ввода/вывода. Для выводов порта предусмотрены внутренние подтягивающие резисторы (выбираются для каждого бита). Выводы PB0 и PB1 также являются положительным (AIN0) и отрицательным (AIN1) входами встроенного аналогового компаратора. Выходные буферы порта В могут поглощать ток до 20мА и непосредственно управлять светодиодными индикаторами. Если выводы PB0..PB7 используются как входы и извне устанавливаются в низкое состояние, они являются источниками тока, если включены внутренние подтягивающие резисторы. Кроме того, Порт В обслуживает некоторые специальные функции, которые будут описаны ниже.

Port D (PD6..PD0) - Порт D является 7-битовым двунаправленным портом с внутренними подтягивающими резисторами. Выходные буферы порта D могут поглощать ток до 20мА. Как входы установленные в низкое состояние, выводы порта D являются источниками тока, если задействованы подтягивающие резисторы. Кроме того, Порт D обслуживает некоторые специальные функции, которые будут описаны ниже.

RESET - Вход сброса. При удержании на входе низкого уровня в течение двух машинных циклов (если генератор работает), сбрасывает устройство.

XTAL1 - вход инвертирующего усилителя генератора и вход внешнего тактового сигнала.

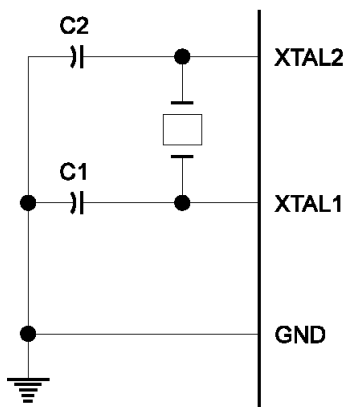
XTAL2 - Выход инвертирующего усилителя генератора.

## КВАРЦЕВЫЙ ГЕНЕРАТОР

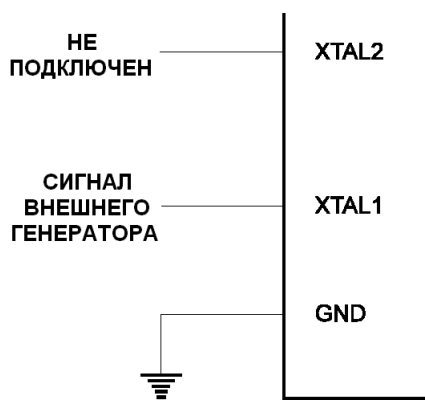
XTAL1 и XTAL2 являются входом и выходом инвертирующего усилителя, который можно использовать для генератора тактовых импульсов. Можно использовать как кварцевые, так и керамические резонаторы. При подключении внешнего тактового сигнала вывод XTAL2 остается неподключенным, а XTAL1 подключается к выходу внешнего генератора.

## ВСТРОЕННЫЙ RC ГЕНЕРАТОР

Встроенный RC генератор работает на фиксированной частоте 1МГц и может быть выбран в качестве источника тактовых сигналов процессора. При необходимости AT90S1200 может работать без внешних компонентов. Управляющий бит RCEN во флэш памяти при установке в единицу выбирает в качестве источника тактовых сигналов встроенный RC генератор. Микросхемы поставляются с незапрограммированным битом RCEN (0), устройства с запрограммированным битом можно заказывать как AT90S1200A. Бит RCEN может быть установлен только при параллельном программировании. При использовании внутреннего RC генератора для загрузки программы через последовательный порт, первоначально параллельным программированием должен быть установлен бит RCEN.



Подключение резонатора



Подключение внешнего генератора

## RISC процессор микроконтроллера AT90S1200

Микроконтроллер AT90S1200 совместим снизу вверх с другими AVR RISC микроконтроллерами. Программы, написанные для AT90S1200, совместимы со всеми 8-разрядными AVR процессорами (AT90Sxxxx) как по исходным кодам, так и по времени исполнения в машинных циклах.

### Обзор архитектуры

Регистровый файл быстрого доступа содержит 32 8-разрядных регистра общего назначения, доступ к которым осуществляется за один машинный цикл. Поэтому за один машинный цикл выполняется одна операция АЛУ. Два операнда выбираются из регистрового файла, выполняется операция, результат ее записывается в регистровый файл - все за один машинный цикл.

АЛУ поддерживает арифметические и логические операции между регистрами и между константами и регистрами. Операции над отдельными регистрами также выполняются в АЛУ. AVR использует Гарвардскую архитектуру - отдельные шины и память программ и данных. Доступ к памяти программ производится через одноуровневый буфер. Во время выполнения одной команды, следующая выбирается из памяти. Такая конструкция позволяет выполнять операции за один машинный цикл. Памятью программ является внутренняя загружаемая флэш память.

При помощи команд относительных переходов и относительных вызовов подпрограмм осуществляется доступ ко всем 512 адресам памяти. Все команды AVR имеют формат одного 16-разрядного слова, поэтому по каждому адресу памяти программ хранится одна команда.

При обработке прерываний и вызове подпрограмм, адрес возврата (значение программного счетчика) сохраняется в стеке. Стек выполнен как 3-уровневый аппаратный стек и используется для подпрограмм и прерываний.

Пространство ввода/вывода содержит 64 адреса периферийных устройств ЦПУ, таких как управляющие регистры, таймер/счетчик, АЦП и другие устройства ввода/вывода. Все области памяти в архитектуре AVR линейны.

Модуль прерываний имеет собственный управляющий регистр в пространстве ввода/вывода, и флаг глобального разрешения прерываний в регистре состояния. Каждому прерыванию назначен свой вектор в начальной области памяти программ. Различные прерывания имеют приоритет в соответствии с расположением их векторов. По младшим адресам расположены векторы с большим приоритетом.

## **Файл регистров общего назначения**

Все оперирующие регистрами команды в наборе команд прямо адресуются к любому из регистров за один машинный цикл. Единственное исключение - пять команд оперирующих с константами SBCI, SUBI, CPI, ANDI, ORI и команда LDI, загружающая регистр константой. Эти команды работают только со второй половиной регистрового файла - R16..R31. Команды SBC, SUB, CP, AND и OR, также как и все остальные, работают со всем регистровым файлом.

Регистр R30 также используется как 8-разрядный указатель для непосредственной адресации регистрового файла.

## **Арифметико-логическое устройство - АЛУ**

АЛУ процессора непосредственно подключено к 32 регистрам общего назначения. За один машинный цикл АЛУ производит операции между регистрами регистрового файла. Команды АЛУ разделены на три основных категории - арифметические, логические и битовые. Некоторые микроконтроллеры семейства AVR имеют аппаратный умножитель в арифметической части АЛУ.

## **Загружаемая память программ.**

AT90S1200 содержит 1кБ загружаемой флэш памяти для хранения программ. Поскольку все команды занимают одно 16-разрядное слово, флэш память организована как 512 16-разрядных слов. Флэш-память выдерживает не менее 1000 циклов перезаписи.

Программный счетчик имеет ширину 9 бит и таким образом адресуется к 512 словам программной флэш-памяти.

Подробно загрузка флэш памяти будет рассмотрена дальше.

## **Режимы адресации программ и данных.**

Усовершенствованные RISC микроконтроллеры AVR поддерживают эффективную и мощную систему адресации.

- Прямая регистровая, отдельный регистр Rd. - адрес регистра указывается в битах 0..4 кода команды, операнд содержится в регистре Rd.
- Косвенная регистровая, - регистр адресуется через указатель Z-регистр (R30).
- Прямая регистровая, два регистра Rd и Rr, - адреса регистров указываются в битах 0..4 (Rd) и 5..9 (Rr) кода команды. Операнды содержатся в регистрах Rd и Rr. Результат помещается в регистр Rd.
- Прямая ввода/вывода. - Адрес операнда содержится в битах 0..5 кода команды, адрес регистра в битах 6..10.
- Относительная программная адресация, RJMP и RCALL. - Смещение относительно программного счетчика указывается в битах 0..11 кода операции. Выполнение программы передается по адресу PC+k. Относительный адрес k лежит в диапазоне от +2кБ до -(2кБ-1).

## **Стек подпрограмм и прерываний**

Для обработки прерываний и вызовов подпрограмм AT90S1200 использует 3-х уровневый аппаратный стек. Аппаратный стек имеет ширину 9 бит и сохраняет значение программного счетчика - адрес возврата из подпрограмм и программ обслуживания прерываний.

Команда RCALL и прерывания записывают значение программного счетчика на 0 уровень стека. Данные, хранящиеся в стеке на уровнях 1 и 2, сдвигаются на один уровень вглубь. При выполнении команды RET или RETI значение программного счетчика выбирается с 0-го уровня стека, данные с уровней 1 и 2 сдвигаются на один уровень вверх.

При вызове более 3-х подпрограмм или прерываний, в стеке сохраняются только 3 последних адреса.

## **EEPROM память данных**

AT90S1200 содержит 64 байта электрически стираемой энергонезависимой памяти (EEPROM). EEPROM организована как отдельная область данных, каждый байт которой может быть прочитан и перезаписан. EEPROM выдерживает не менее 100000 циклов записи/стирания. Доступ к энергонезависимой памяти данных рассмотрен дальше и задается регистром адреса, регистром данных и управляющим регистром. Ниже рассмотрено и программирование памяти данных через SPI интерфейс.

## **Время выполнения команд.**

ЦПУ процессора AVR управляется системной частотой генерируемой внешним резонатором. Внутреннее деление частоты генератора не используется.

В процессоре организован буфер (pipeline) команд, при выборе команды из памяти программ происходит выполнение предыдущей команды. Подобная концепция позволяет достичь быстродействия 1MIPS на МГц, уникальных показателей стоимости, быстродействия и потребления процессора.

## Память ввода/вывода

Ниже приведено описание пространства ввода/вывода для процессоров AT90S1200.

Все устройства ввода/вывода и периферийные устройства AT90S1200 располагаются в пространстве ввода/вывода. Различные ячейки этого пространства доступны через команды IN и OUT, пересылающие данные между одним из 32-х регистров общего назначения и пространством ввода/вывода. К регистрам \$00..\$19 можно осуществлять побитовый доступ командами SBI и CBI. Значение отдельного бита этих регистров можно проверить командами SBIC и SBIS. Дополнительную информацию по этому вопросу можно найти в описании системы команд.

Таблица 1. Пространство ввода/вывода AT90S1200

\$3F	SREG	Status REGister	Регистр Состояния
\$3B	GIMSK	General Interrupt MaSK register	Общий регистр маски прерываний
\$39	TIMSK	Timer/counter Interrupt mask register	Регистр маски прерываний от таймера/счетчика
\$38	TIFR	Timer/counter Interrupt Flag register	Регистр флага прерывания таймера/счетчика
\$35	MCUCR	MCU general Control Register	Общий регистр управления микроконтроллером
\$33	TCCR0	Timer/Counter 0 Control Register	Регистр управления таймером счетчиком 0
\$32	TCNT0	Timer/Counter 0 (8-бит)	Таймер/счетчик 0 (8 бит)
\$21	WDTCR	Watchdog Timer Control Register	Регистр управления сторожевым таймером
\$1E	EEAR	EEPROM Address Register	Регистр адреса энергонезависимой памяти
\$1D	EEDR	EEPROM Data Register	Регистр данных энергонезависимой памяти
\$1C	EECR	EEPROM Control Register	Регистр управления энергонезависимой памяти
\$18	PORTB	Data Register, Port B	Регистр данных порта B
\$17	DDRB	Data Direction Register Port B	Регистр направления данных порта B
\$16	PINB	Input pins, Port B	Выводы порта B
\$12	PORTD	Data Register, Port D	Регистр данных порта D
\$11	DDRD	Data Direction Register Port D	Регистр направления данных порта D
\$10	PIND	Input pins, Port D	Выводы порта D
\$08	ACSR	Analog Comparator Control and Status Register	Регистр управления и состояния аналогового компаратора

## Регистр состояния - SREG

Регистр состояния расположен по адресу \$3F пространства ввода/вывода и определен следующим образом:

Бит	7	6	5	4	3	2	1	0	
\$3F	<b>I</b>	<b>T</b>	<b>H</b>	<b>S</b>	<b>V</b>	<b>N</b>	<b>Z</b>	<b>C</b>	SREG
Чт./зап. (R/W)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальн.знач.	0	0	0	0	0	0	0	0	

Бит 7 - **I**: Общее разрешение прерываний. Для разрешения прерываний этот бит должен быть установлен в единицу. Управление отдельными прерываниями производится регистром маски прерываний - GIMSK/TIMSK. Если флаг сброшен (0), независимо от состояния GIMSK/TIMSK, прерывания не разрешены. Бит I очищается аппаратно после входа в прерывание и восстанавливается командой RETI, для разрешения обработки следующих прерываний.

Бит 6 - **T**: Хранение копируемого бита. Команды копирования битов BLD (Bit Load) и BST (Bit Store) используют этот бит как источник и приемник обрабатываемого бита. Бит из регистра регистравого файла может быть скопирован в T командой BST, бит T может быть скопирован в бит регистравого файла командой BLD.

Бит 5 - **H**: Флаг половинного переноса. Этот флаг индицирует перенос из младшей половины байта при некоторых арифметических операциях. Более подробно об этом можно прочитать в описании системы команд.

Бит 4 - **S**: бит знака,  $S = N \text{ XOR } V$ . Бит S всегда равен исключаящему ИЛИ между флагами N (отрицательный результат) и V (переполнение дополнения до двух). Более подробно об этом можно прочитать в описании системы команд.

Бит 3 - **V**: Флаг переполнения дополнения до двух. Этот флаг поддерживает арифметику с дополнением до двух. Более подробно об этом можно прочитать в описании системы команд.

Бит 2 - **N**: Флаг отрицательного результата. Этот флаг индицирует отрицательный результат различных арифметических и логических операций. Более подробно об этом можно прочитать в описании системы команд.

Бит 1 - **Z**: Флаг нулевого результата. Этот флаг индицирует нулевой результат различных арифметических и логических операций. Более подробно об этом можно прочитать в описании системы команд.

Бит 0 - **C**: Флаг переноса. Этот флаг индицирует перенос в арифметических и логических операциях. Более подробно об этом можно прочитать в описании системы команд.

## Сброс и обработка прерываний.

В AT90S1200 предусмотрены 3 различных источника прерываний. Эти прерывания и сброс имеют различные векторы в области памяти программ. Каждое из прерываний разрешается отдельным битом разрешающим данное прерывание и битом I разрешающим обработку всех прерываний.

Младшие адреса памяти программ отведены под векторы прерываний, список векторов прерываний приведен в таблице 2. Этот список также определяет и приоритет различных прерываний. Чем младше адрес, тем выше приоритет прерывания. Наивысший приоритет имеет сброс, за ним INT0 и т.д.

Таблица 2. Сброс и векторы прерываний.

Номер вектора	Адрес	Источник	Описание прерывания
1	\$000	RESET	Ножка сброса и сброс от сторожевого таймера
2	\$001	INT0	Внешнее прерывание 0
4	\$002	TIMER0,OVF0	Переполнение таймера/счетчика 0
5	\$003	ANA_COMP	Аналоговый компаратор

Чаще всего используется следующая установка векторов прерываний в программе:

```

Адрес  Метка      Код          Комментарий
$000   ;             rjmp RESET   ; обработка сброса
$001   ;             rjmp EXT_INT0 ; обработка IRQ0
$002   ;             rjmp TIM0_OVF  ; обработка переполнения таймера 0
$003   ;             rjmp ANA_COMP  ; обслуживание аналогового компаратора
;
$004   MAIN:    <instr> xxx    ; начало основной программы

```

## ИСТОЧНИКИ СБРОСА

AT90S1200 имеет три источника сброса.

- \* Сброс по включению питания. Процессор сбрасывается при подаче питания на выводы VCC и GND.
- \* Внешний сброс. Процессор сбрасывается при подаче низкого уровня на вывод RESET на время более двух периодов тактовой частоты.
- \* Сброс от сторожевого таймера. Процессор сбрасывается по окончании времени сторожевого таймера, если разрешена его работа.

Во время сброса все регистры ввода/вывода устанавливаются в начальные значения, программа начинает выполняться с адреса \$000, по этому адресу должна быть записана команда RJMP - относительный переход на программу обработки сброса. Если в программе не разрешаются прерывания и векторы прерываний не используются, в первых адресах памяти может быть записана программа.

Таблица 3. Характеристики сброса.

		Min	Тип	Max	
V <sub>por</sub>	Напряжение срабатывания сброса по включению питания	1.8	2	2.2	В
V <sub>rst</sub>	Напряжение срабатывания сброса по выводу RESET		VCC/2		В
T <sub>por</sub>	Сигн. сброса по вкл. питания	2	3	4	мс
T <sub>tout</sub>	Задержка на отработку сброса	11	16	21	мс

## СБРОС ПО ВКЛЮЧЕНИЮ ПИТАНИЯ

Цепь сброса по включению питания обеспечивает задержку запуска устройства до тех пор, пока не стабилизируется генератор тактовых сигналов. Внутренний таймер, работающий от тактовых сигналов сторожевого таймера, задерживает запуск процессора на несколько периодов после того, как напряжение питания достигнет значения V<sub>por</sub>. Общее время сброса состоит из длительности сигнала сброса по включению питания и задержки на отработку сброса T<sub>por</sub>+T<sub>tout</sub>.

Поскольку вывод RESET подтянут к источнику питания внутренним резистором, вывод может оставаться неподключенным, если внешний сброс не требуется. Того же эффекта можно достичь подключив вывод RESET к источнику питания. При удержании вывода RESET в низком уровне после подачи напряжения питания, время отработки сброса можно расширить.

## ВНЕШНИЙ СБРОС

Внешнее прерывание генерируется низким уровнем на выводе RESET. Вывод должен удерживаться в низком состоянии, по крайней мере, на два периода тактовой частоты. После того как напряжение на выводе RESET достигнет значения V<sub>rst</sub>, внутренний таймер запустит процессор после отработки времени T<sub>tout</sub>.

## СБРОС ОТ СТОРОЖЕВОГО ТАЙМЕРА

После отработки сторожевого таймера, генерируется короткий импульс сброса длительностью в один период тактовой частоты. По окончании этого импульса внутренний таймер начинает отсчитывать время T<sub>tout</sub>. Подробно работа сторожевого таймера будет рассмотрена дальше.

## ОБРАБОТКА ПРЕРЫВАНИЙ

AT90S1200 имеет два регистра маски прерываний GIMSK - общий регистр маски прерываний, расположенный по адресу \$3B и TIMSK - регистр маски прерываний от таймера/счетчика - по адресу \$39.

Когда возникает прерывание, общий бит разрешения прерываний I очищается (ноль) и прерывания запрещаются. Программа пользователя может установить этот бит для разрешения прерываний. Флаг разрешения прерываний I устанавливается в 1 при выполнении команды выхода из прерывания - RETI.

Для прерываний включаемых статическими событиями (т.е. переключаемыми уровнем) флаг прерывания взводится, когда происходит событие. Если флаг прерывания очищен и присутствует условие возникновения прерывания, флаг не будет установлен, пока не произойдет следующее событие.

Когда программный счетчик устанавливается на текущий вектор прерывания для обработки прерывания, соответствующий флаг, сгенерированный прерыванием, аппаратно сбрасывается. Некоторые флаги прерывания могут быть сброшены записью логической единицы в бит соответствующий флагу.



### ОБЩИЙ РЕГИСТР МАСКИ ПЕРЕРЫВАНИЙ - GIMSK

Бит	7	6	5	4	3	2	1	0
\$3B	-	<b>INT0</b>	-	-	-	-	-	-
Чт./зап. (R/W)	R	R/W	R	R	R	R	R	R
Начальн.знач.	0	0	0	0	0	0	0	0

Бит 7 - зарезервирован. В AT90S1200 этот бит зарезервирован и всегда читается как 0.

Бит 6 - INT0: Запрос внешнего прерывания 0 разрешен. Когда этот бит установлен, а также установлен бит I регистра состояния, разрешается прерывание от внешнего вывода. Биты управления запуском прерывания (ISC01 и ISC00) в регистре управления микроконтроллером (MCUCR) определяют по какому событию отрабатывается прерывание - по спадающему или нарастающему фронту или же по уровню. Если вывод INT0 используется для работы с внешним источником прерывания, бит DDD2 в регистре направления данных порта D (DDRD), должен быть сброшен в 0, чтобы вывод INT0 работал как вход.

Биты 5..0 - зарезервированы. В AT90S1200 эти биты зарезервированы и всегда читаются как 0.

### РЕГИСТР МАСКИ ПЕРЕРЫВАНИЯ ОТ ТАЙМЕРА/СЧЕТЧИКА - TIMSK

Бит	7	6	5	4	3	2	1	0
\$39	-	-	-	-	-	-	<b>TOIE0</b>	-
Чт./зап. (R/W)	R	R	R	R	R	R	R/W	R
Начальн.знач.	0	0	0	0	0	0	0	0

Биты 7..2 - зарезервированы. В AT90S1200 эти биты зарезервированы и всегда читаются как 0.

Бит 1 - TOIE0: Разрешение прерывания по переполнению таймера/счетчика. Если этот бит установлен в 1, и бит I в регистре состояния установлен в 1, разрешены прерывания от таймера/счетчика. При возникновении переполнения выполняется соответствующий вектор прерывания (\$003). Флаг переполнения (TOV0) во флаговом регистре прерываний (TIFR) от таймера/счетчика устанавливается в 1.

Бит 0 - зарезервирован. В AT90S1200 этот бит зарезервирован и всегда читается как 0.

### ФЛАГОВЫЙ РЕГИСТР ПЕРЕРЫВАНИЙ ОТ ТАЙМЕРА/СЧЕТЧИКА - TIFR

Бит	7	6	5	4	3	2	1	0
\$38	-	-	-	-	-	-	<b>TOV0</b>	-
Чт./зап. (R/W)	R	R	R	R	R	R	R/W	R
Начальн.знач.	0	0	0	0	0	0	0	0

Биты 7..2 - зарезервированы. В AT90S1200 эти биты зарезервированы и всегда читаются как 0.

Бит 1 - Бит TOV0 устанавливается в 1, когда происходит переполнение таймера/счетчика 0. TOV0 очищается аппаратно при выходе из соответствующей программы обработки прерывания. Другой способ обнуления флага - записать в него 1. Прерывание по переполнению таймера/счетчика возникает, если установлен флаг I регистра состояния, разрешены прерывания от таймера/счетчика и установлен бит TOV0.

Бит 0 - зарезервирован. В AT90S1200 этот бит зарезервирован и всегда читается как 0.

## ВНЕШНИЕ ПЕРЕРЫВАНИЯ

Внешние прерывания управляются выводом INT0. Прерывания могут возникать по нарастающему и спадающему фронтам сигнала, а также по уровню. Это устанавливается управляющим регистром MCUCR. Если прерывание отрабатывается по уровню, оно вызывается до тех пор, пока вывод INT0 удерживается в низком состоянии.

Прерывания отрабатываются даже если вывод INT0 сконфигурирован на вывод. Это дает возможность генерировать программные прерывания. Пользователь не имеет непосредственного доступа к флагу прерывания. Если ожидается внешнее прерывание по фронту импульса, этот флаг можно очистить следующим способом:

1. Запретить внешние прерывания, очистив флаг INT0 в GIMSK
2. Выбрать прерывание по уровню
3. Выбрать нужный вид фронта прерывания
4. Повторно разрешить внешние прерывания, установив INT0 в GIMSK.

## ВРЕМЯ РЕАКЦИИ НА ПРЕРЫВАНИЕ

Минимальное время реакции на любое из предусмотренных в процессоре прерываний - 4 периода тактовой частоты. После четырех циклов вызывается программный вектор обрабатывающий данное прерывание. За эти 4 цикла программный счетчик (9 бит) записывается в стек. Программный вектор представляет собой относительный переход на подпрограмму обслуживания прерывания и этот переход занимает 2 периода тактовой частоты. Если прерывание происходит во время выполнения команды длящейся несколько циклов, перед вызовом прерывания завершается выполнение этой команды.

Выход из программы обслуживания прерывания занимает 4 периода тактовой частоты. За эти 4 периода из стека восстанавливается программный счетчик. После выхода из прерывания процессор всегда выполняет еще одну команду, прежде чем обслужить любое отложенное прерывание.

Заметим, что регистр состояния SREG аппаратно не обрабатывается процессором, как при вызове подпрограмм, так и при обслуживании прерываний. Если программа требует сохранения SREG, то это должно производиться программой пользователя.

Стек прерываний и подпрограмм имеет глубину 3 уровня, если выполняются больше трех вложенных прерываний или подпрограмм, запоминаются только 3 последних адреса.

## РЕГИСТР УПРАВЛЕНИЯ МИКРОКОНТРОЛЛЕРОМ

Этот регистр содержит биты общего управления микроконтроллером.

Бит	7	6	5	4	3	2	1	0	
\$35	-	-	SE	SM	-	-	ISC01	ISC00	MCUCR
Чт./зап. (R/W)	R	R	R/W	R/W	R	R	R/W	R/W	
Начальн.знач.	0	0	0	0	0	0	0	0	

Биты 7,6 - зарезервированы. В AT90S1200 эти биты зарезервированы и всегда читаются как 0.

Бит 5 - Sleep Enable -Разрешение режима Sleep. Этот бит должен быть установлен в 1, чтобы при выполнении команды SLEEP процессор переходил в режим пониженного энергопотребления (Sleep). Для использования режима пониженного энергопотребления этот бит рекомендуется устанавливать в 1 до исполнения команды SLEEP.

Бит 4 - Sleep Mode - Режим Sleep. Этот бит выбирает один из доступных режимов пониженного энергопотребления. Если бит сброшен (0), то в качестве режима Sleep выбирается холостой режим (Idle mode). Если бит установлен, - выбирается экономичный режим (Power down). Особенности каждого из режимов будут рассмотрены ниже.

Биты 3,2 - зарезервированы. В AT90S1200 эти биты зарезервированы и всегда читаются как 0.

Биты 1,0 - ISC01, ISC00 - биты управления срабатыванием прерывания 0. Внешнее прерывание активируется выводом INT0, если установлен флаг I регистра состояния SREG и установлена соответствующая маска в регистре GIMSK. Срабатывание по уровню и фронтам задается следующим образом:

Таблица 4. Управление срабатыванием прерывания.

ISC01	ISC00	Описание
0	0	Запрос прерывания генерируется по низкому уровню на входе
0	1	Зарезервировано
1	0	Запрос на прерывание по спадающему фронту на входе
1	1	Запрос на прерывание по нарастающему фронту на входе

Примечание: При изменении битов ISC01 и ISC00, прерывания по входу INT0 должны быть запрещены сбросом бита разрешения прерывания в регистре GIMSK. Иначе прерывание может произойти при изменении значения битов.

## Режимы пониженного энергопотребления.

Для запуска режима пониженного энергопотребления должен быть установлен (1) бит SE регистра MCUCR, и должна быть исполнена команда SLEEP. Если во время нахождения в режиме пониженного потребления происходит одно из разрешенных прерываний, процессор начинает работать, исполняет подпрограмму обработки прерывания и продолжает выполнение программы с команды следующей за SLEEP. Содержимое регистрового файла и памяти ввода/вывода не изменяется. Если в режиме пониженного потребления происходит сброс, процессор начинает выполнение программы с вектора сброса.

Если используется прерывание по уровню, для вывода из режима Power Down, низкий уровень должен удерживаться на время достаточное для запуска генератора тактовых импульсов - 16 мс. Иначе флаг прерывания может установиться в 0 до того как процессор начнет работу.

## Режим холостого хода.

Когда бит SM сброшен (0), команда SLEEP переводит процессор в режим холостого хода (Idle mode). ЦПУ останавливается, но Таймер/Счетчик, сторожевой таймер и система прерываний продолжают работать. Это позволяет процессору возобновлять работу как от внешних прерываний, так и по переполнению таймера/счетчика или по сбросу от сторожевого таймера. Если прерывание от аналогового компаратора не требуется, аналоговый компаратор может быть отключен установкой бита ACD регистра ACSR. Это уменьшает потребляемую мощность в режиме холостого хода.

## Экономичный режим.

Когда бит SM установлен (1), команда SLEEP переводит процессор в экономичный режим (Power Down Mode). В этом режиме останавливается внешний генератор тактовых импульсов. Пользователь может разрешить работу сторожевого таймера в этом режиме. Если сторожевой таймер разрешен, процессор выходит из экономичного режима после отработки периода сторожевого таймера. Если сторожевой таймер запрещен, выход из экономичного режима может произойти только по внешнему сбросу или прерыванию по уровню.

## ТАЙМЕР/СЧЕТЧИК

В AT90S1200 предусмотрен 8-разрядный таймер/счетчик общего назначения. Таймер/счетчик получает импульсы от 10-битового предварительного делителя. Таймер/Счетчик можно использовать как таймер, с внутренним источником импульсов или как счетчик, управляемый внешними счетными импульсами.

8-разрядный таймер/счетчик может получать импульсы тактовой частоты - СК, импульсы с предварительного делителя (СК/8, СК/64, СК/256 или СК/1024), импульсы с внешнего вывода или быть остановлен соответствующими установками регистра TCCR0. Флаг переполнения таймера находится в регистре TIFR. Биты управления таймером расположены в регистре TCCR0. Разрешение и запрещение прерываний от таймера управляется регистром TIMSK.

При работе таймера/счетчика от внешнего сигнала, внешний сигнал синхронизируется с тактовым генератором ЦПУ. Для правильной обработки внешнего сигнала, минимальное время между соседними импульсами должно превышать период тактовой частоты процессора. Сигнал внешнего источника обрабатывается по спадающему фронту тактовой частоты процессора.

8-разрядный таймер/счетчик можно использовать как счетчик с высоким разрешением, так и для точных применений с низким коэффициентом деления тактовой частоты. Более высокие коэффициенты деления можно использовать для медленных функций или измерения временных интервалов между редкими событиями.

### РЕГИСТР УПРАВЛЕНИЯ ТАЙМЕРОМ/СЧЕТЧИКОМ 0 - TCCR0

Бит	7	6	5	4	3	2	1	0	
\$33	-	-	-	-	-	CS02	CS01	CS00	TCCR0
Чт./зап. (R/W)	R	R	R	R	R	R/W	R/W	R/W	
Начальн.знач.	0	0	0	0	0	0	0	0	

Биты 7..3 - зарезервированы. В AT90S1200 эти биты зарезервированы и всегда читаются как 0.

Биты 2,1,0 - CS02, CS01, CS00 - выбор тактовой частоты. Эти биты задают коэффициент деления предварительного делителя.

Таблица 5. Выбор коэффициента предварительного деления

CS02	CS01	CS00	Описание
0	0	0	Таймер/счетчик остановлен
0	0	1	СК
0	1	0	СК/8
0	1	1	СК/64
1	0	0	СК/256
1	0	1	СК/1024
1	1	0	Внешний вывод T0, нарастающий фронт
1	1	1	Внешний вывод T0, спадающий фронт

Условие Stop запрещает/разрешает функционирование таймера/счетчика. В режимах деления частота берется непосредственно от тактовых импульсов. При использовании работы от внешнего вывода предварительно должен быть установлен вывод регистра направления данных (0 - включает ножку на ввод).

#### ТАЙМЕР/СЧЕТЧИК 0 - TCNT0.

Бит	7	6	5	4	3	2	1	0	
\$32	<b>MSB</b> - - - - - <b>LSB</b>								TCNT0
Чт./зап. (R/W)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальн.знач.	0	0	0	0	0	0	0	0	

Таймер/счетчик реализован как нарастающий счетчик с возможностью чтения и записи. При записи таймера/счетчика, если присутствуют тактовые импульсы, таймер/счетчик продолжает счет в цикле следующем за операцией записи.

## СТОРОЖЕВОЙ ТАЙМЕР

Сторожевой таймер работает от отдельного встроенного генератора работающего на частоте 1 МГц. Управляя предварительным делителем сторожевого таймера можно задавать интервал сброса таймера от 16 до 2048 периодов. Частота встроенного RC генератора зависит от напряжения питания процессора.

Команда WDR сбрасывает сторожевой таймер. Для работы сторожевого таймера можно выбрать одно из 8-ми значений частоты, что позволяет в широких пределах изменять время между исполнением команды WDR и сбросом процессора. При отработке периода работы сторожевого таймера, если не поступила команда WDR, AT90S1200 сбрасывается, выполнение программы продолжается с вектора сброса.

#### РЕГИСТР УПРАВЛЕНИЯ СТОРОЖЕВЫМ ТАЙМЕРОМ - WDTCR

Бит	7	6	5	4	3	2	1	0	
\$21	-	-	-	-	<b>WDE</b>	<b>WDP2</b>	<b>WDP1</b>	<b>WDP0</b>	WDTCR
Чт./зап. (R/W)	R	R	R	R	R/W	R/W	R/W	R/W	
Начальн.знач.	0	0	0	0	0	0	0	0	

Биты 7..4 - зарезервированы. В AT90S1200 эти биты зарезервированы и всегда читаются как 0.

Бит 3 - WDE- разрешение сторожевого таймера. Если бит установлен (1), то работа сторожевого таймера разрешена, если бит сброшен - запрещена.

Биты 2..0 - WDP2..0 - Биты предварительного делителя сторожевого таймера. Если работа сторожевого таймера разрешена, эти биты определяют предварительный коэффициент деления для сторожевого таймера. В таблице 6 приведены различные значения установок предварительного делителя и соответствующие им временные интервалы для напряжения питания Vcc=5V.

Таблица 6. Установки предварительного делителя сторожевого таймера

WDP2	WDP1	WDP0	Период времени
0	0	0	16 мс
0	0	1	32 мс
0	1	0	64 мс
0	1	1	128 мс

WDP2	WDP1	WDP0	Период времени
1	0	0	256 мс
1	0	1	512 мс
1	1	0	1024 мс
1	1	1	2048 мс

## Доступ к энергонезависимой памяти

Регистры доступа к энергонезависимой памяти (EEPROM) расположены в пространстве ввода/вывода.

Время записи лежит в диапазоне 2.5-4 мс, в зависимости от напряжения питания. Это самоактивируемая функция которая, однако, позволяет пользователю узнать, когда можно записывать следующий байт. Детектор работы EEPROM позволяет защитить данные EEPROM от записи, если напряжение питания лежит ниже некоторого уровня.

При записи или чтении EEPROM процессор приостанавливается на 2 машинных цикла до начала выполнения следующей команды.

#### РЕГИСТР АДРЕСА EEPROM - EEAR

Бит	7	6	5	4	3	2	1	0	
\$1E	-	-	<b>EEAR5</b>	<b>EEAR4</b>	<b>EEAR3</b>	<b>EEAR2</b>	<b>EEAR1</b>	<b>EEAR0</b>	EEAR
Чт./зап. (R/W)	R	R	R/W	R/W	R/W	R/W	R/W	R/W	
Начальн.знач.	0	0	0	0	0	0	0	0	

Биты 7,6 - зарезервированы. В AT90S1200 эти биты зарезервированы и всегда читаются как 0.

Биты 5..0 - EEAR5..0 - Адрес EEPROM. Адресный регистр EEPROM задает адрес в 64-байтном пространстве EEPROM. Байты данных EEPROM адресуются линейно в диапазоне 0..63.

#### РЕГИСТР ДАННЫХ EEPROM - EEDR

Бит	7	6	5	4	3	2	1	0	
\$1D	<b>MSB</b>	-	-	-	-	-	-	<b>LSB</b>	EEDR
Чт./зап. (R/W)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальн.знач.	0	0	0	0	0	0	0	0	

Биты 7..0 - EEDR7..0 - Данные EEPROM. Для операции записи регистр EEDR содержит данные, которые будут записаны в EEPROM по адресу в регистре EEAR. Для операции чтения в этот регистр читаются данные прочитанные из EEPROM по адресу в регистре EEAR.

#### РЕГИСТР УПРАВЛЕНИЯ EEPROM - EECR

Бит	7	6	5	4	3	2	1	0	
\$1C	-	-	-	-	-	-	<b>EEWE</b>	<b>EERE</b>	EECR
Чт./зап. (R/W)	R	R	R	R	R	R	R/W	R/W	
Начальн.знач.	0	0	0	0	0	0	0	0	

Биты 7..2 - зарезервированы. В AT90S1200 эти биты зарезервированы и всегда читаются как 0.

Бит 1 - EEWE - Разрешение записи в EEPROM. Сигнал EEWE является стробом записи в EEPROM. После установки правильного адреса и данных для записи в EEPROM необходимо установить бит EEWE. После того как время записи истечет (типично 2.5 мс для Vcc=5V и 4 мс для Vcc=2.7), бит EEWE очищается аппаратно. Пользователь может отслеживать этот бит и ожидать установки его в ноль, перед тем как записывать следующий байт. При установке EEWE, ЦПУ останавливается на два цикла перед исполнением следующей команды.

Бит 0 - EERE - разрешение чтения из EEPROM. Сигнал EERE является стробом чтения из EEPROM. После установки нужного адреса в регистре EEAR, необходимо установить бит EERE. После того как бит EERE будет аппаратно очищен, в регистре EEDR. Чтение EEPROM занимает одну команду и не требует отслеживания бита EERE. При установке бита EERE, ЦПУ останавливается на два цикла, перед тем как будет выполнена следующая команда.

## АНАЛОГОВЫЙ КОМПАРАТОР

Аналоговый компаратор сравнивает входные напряжение на положительном входе PB0 (AIN0) и отрицательном входе (AIN1). Когда напряжение на положительном входе больше напряжения на отрицательном, устанавливается бит ACO (Analog Comparator Output). Выход аналогового компаратора можно включить на прерывание от аналогового компаратора. Пользователь может установить срабатывание прерывание по нарастающему или спадающему фронту, или по переключению.

#### РЕГИСТР УПРАВЛЕНИЯ И СОСТОЯНИЯ АНАЛОГОВОГО КОМПАРАТОРА.

Бит	7	6	5	4	3	2	1	0	
\$08	<b>ACD</b>	-	<b>ACO</b>	<b>ACI</b>	<b>ACIE</b>	-	<b>ACIS1</b>	<b>ACIS0</b>	ACSR
Чт./зап. (R/W)	R/W	R	R/W	R/W	R/W	R	R/W	R/W	
Начальн.знач.	0	0	0	0	0	0	0	0	

Бит 7 - ACD - Запрещение аналогового компаратора. Когда этот бит установлен, питание от аналогового компаратора отключается. Для отключения компаратора этот бит можно установить в любое время. Обычно это свойство используется, если критично потребление процессора в холостом режиме и восстановление работы процессора от аналогового компаратора не требуется. При изменении бита ACD прерывания от аналогового компаратора должны быть запрещены сбросом ACIE в регистре ACSR. В противном случае прерывание может произойти во время изменения бита.

Бит 6 - зарезервирован. В AT90S1200 этот бит зарезервирован и всегда читается как 0.

Бит 5 - ACO - Выход аналогового компаратора. Бит ACO непосредственно подключен к выходу аналогового компаратора.

Бит 4 - ACI - Флаг прерывания от аналогового компаратора. Этот бит устанавливается, когда переключение выхода компаратора совпадает с режимом прерывания установленным битами ACIS1 и ACIS0. Программа обработки прерывания от аналогового компаратора выполняется, если установлен бит ACIE (1) и установлен бит I в регистре состояния. ACI сбрасывается аппаратно при выполнении соответствующего вектора прерывания. Другой способ очистить ACI - записать во флаг логическую единицу.

Бит 3 - ACIE - Разрешение прерывания от аналогового компаратора. Когда установлен этот бит и бит I регистра состояния, прерывания от аналогового компаратора отрабатываются. Если бит очищен (0), прерывания запрещены.

Бит 2 - зарезервирован. В AT90S1200 этот бит зарезервирован и всегда читается как 0.

Биты 1,0 - ACIS1, ACIS0 - Выбор режима прерывания аналогового компаратора. Различные установки приведены в табл.7.

Таблица 7. Установки ACIS1/ACIS0

ACIS1	ACIS0	Описание
0	0	Прерывание от компаратора по переключению выхода
0	1	Зарезервировано
1	0	Прерывание от компаратора по спадающему фронту выхода
1	1	Прерывание от компаратора по нарастающему фронту выхода

Примечание: При изменении битов ACIS1/ACIS0 прерывания от аналогового компаратора должны быть запрещены сбросом бита разрешения прерывания в регистре ACSR. Иначе прерывание может произойти при изменении битов.

## ПОРТЫ ВВОДА/ВЫВОДА

### Порт В

Порт В 8-разрядный двунаправленный порт.

Для обслуживания порта отведено три регистра: регистр данных PORTB (\$18), регистр направления данных - DDRB (\$17) и ножки порта В (\$16). Адрес ножек порта В предназначен только для чтения, в то время как регистр данных и регистр направления данных - для чтения/записи.

Все выводы порта имеют отдельно выбираемые нагрузки. Выходы порта В могут поглощать ток до 20 мА и непосредственно управлять светодиодными индикаторами. Если выводы PB0..PB7 используются как входы и замыкаются на землю, если включены внутренние подтягивающие резисторы, выводы являются источниками тока ( $I_{IL}$ ). Дополнительные функции выводов порта В приведены в таблице 8.

Таблица 8. Альтернативные функции выводов порта В

Вывод	Альтернативная функция
PB0	AIN0 (Положительный вход аналогового компаратора)
PB1	AIN1 (Отрицательный вход аналогового компаратора)
PB5	MOSI (Вход данных для загрузки памяти)
PB6	MISO (Выход данных для чтения памяти)
PB7	SCK (Вход тактовых импульсов последовательного обмена)

При использовании альтернативных функций выводов. Регистры DDRB и PORTB должны быть установлены в соответствии с описанием альтернативных функций.

#### РЕГИСТР ДАННЫХ ПОРТА В - PORTB

Бит	7	6	5	4	3	2	1	0	
\$18	<b>PORTB7</b>	<b>PORTB6</b>	<b>PORTB5</b>	<b>PORTB4</b>	<b>PORTB3</b>	<b>PORTB2</b>	<b>PORTB1</b>	<b>PORTB0</b>	PORTB
Чт./зап. (R/W)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальн.знач.	0	0	0	0	0	0	0	0	

#### РЕГИСТР НАПРАВЛЕНИЯ ДАННЫХ ПОРТА В - DDRB

Бит	7	6	5	4	3	2	1	0	
\$17	<b>DDB7</b>	<b>DDB6</b>	<b>DDB5</b>	<b>DDB4</b>	<b>DDB3</b>	<b>DDB2</b>	<b>DDB1</b>	<b>DDB0</b>	DDRB
Чт./зап. (R/W)	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальн.знач.	0	0	0	0	0	0	0	0	

### ВЫВОДЫ ПОРТА В - PINB

Бит	7	6	5	4	3	2	1	0	
\$16	<b>PINB7</b>	<b>PINB6</b>	<b>PINB5</b>	<b>PINB4</b>	<b>PINB3</b>	<b>PINB2</b>	<b>PINB1</b>	<b>PINB0</b>	PINB
Чт./зап. (R/W)	R	R	R	R	R	R	R	R	
Начальн.знач.	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	

PINB не является регистром, по этому адресу осуществляется доступ к физическим значениям каждого из выводов порта В. При чтении PORTB, читаются данные из регистра-защелки, при чтении PINB читаются логические значения присутствующие на выводах порта.

Порт В, как порт ввода/вывода общего назначения

Все 8 бит порта В при использовании для ввода/вывода одинаковы.

Бит DDBn регистра DDRB выбирает направление передачи данных. Если бит установлен (1), вывод сконфигурирован как выход. Если бит сброшен (0) - вывод сконфигурирован как вход. Если PORTBn установлен и вывод сконфигурирован как вход, включается КМОП подтягивающий резистор. Для отключения резистора, PORTBn должен быть сброшен (0) или вывод должен быть сконфигурирован как выход.

### АЛЬТЕРНАТИВНЫЕ ФУНКЦИИ ПОРТА В

SCK - бит 7 порта В - тактовые импульсы для записи/чтения памяти

MISO - бит 6 порта В - выход данных для чтения памяти

MOSI - бит 5 порта В - вход данных для записи памяти

AIN1 - бит 1 порта В - Отрицательный вход аналогового компаратора. Если вывод сконфигурирован как вход (DDB1 сброшен) и отключен внутренний подтягивающий резистор (PB0 сброшен), этот вывод работает как отрицательный вход встроенного аналогового компаратора.

AIN0 - бит 0 порта В - Положительный вход аналогового компаратора. Если вывод сконфигурирован как вход (DDB0 сброшен) и отключен внутренний подтягивающий резистор (PB1 сброшен), этот вывод работает как положительный вход встроенного аналогового компаратора.

Таблица 9. Влияние DDBn на выводы порта В

DDBn	PORTBn	Вх/Вых	Подт.резист	Комментарий
0	0	Вход	Нет	Третье состояние (Hi-Z)
0	1	Вход	Да	PBn источник тока $I_{IL}$ , если извне соединен с землей
1	0	Выход	Нет	Выход установлен в 0
1	1	Выход	Нет	Выход установлен в 1

n = 7,6...0 - номер вывода

### Порт D

Для порта D зарезервированы 3 ячейки памяти - регистр PORTD (\$12), регистр направления данных - DDRD (\$11) и выводы порта D - PIND (\$10). Регистры данных и направления данных могут читаться/записываться, ячейка PIND - только для чтения.

Порт D - 7-разрядный двунаправленный порт с встроенными подтягивающими регистрами. Выходные буферы порта могут поглощать ток до 20 мА. Если выводы используются как входы и на них подан низкий уровень, они являются источниками тока  $I_{IL}$ , если подключены подтягивающие резисторы. Некоторые из выводов порта имеют альтернативные функции, как показано в следующей таблице:

Таблица 10. Альтернативные функции порта D

Вывод порта	Альтернативная функция
PD2	INT0 (вход внешнего прерывания 0)
PD4	T0 (внешний вход таймера счетчика 0)

РЕГИСТР ДАННЫХ ПОРТА D - PORTD

Бит	7	6	5	4	3	2	1	0	
\$12	-	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0	PORTD
Чт./зап. (R/W)	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальн.знач.	0	0	0	0	0	0	0	0	

РЕГИСТР НАПРАВЛЕНИЯ ДАННЫХ ПОРТА D - DDRD

Бит	7	6	5	4	3	2	1	0	
\$11	-	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	DDRD
Чт./зап. (R/W)	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальн.знач.	0	0	0	0	0	0	0	0	

ВЫВОДЫ ПОРТА D - PIND

Бит	7	6	5	4	3	2	1	0	
\$10	-	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	PIND
Чт./зап. (R/W)	R	R	R	R	R	R	R	R	
Начальн.знач.	0	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	

PIND не является регистром, по этому адресу осуществляется доступ к физическим значениям каждого из выводов порта D. При чтении PORTD, читаются данные из регистра-защелки, при чтении PIND читаются логические значения присутствующие на выводах порта.

Порт D, как порт ввода/вывода общего назначения

Все 7 бит порта D при использовании для ввода/вывода одинаковы.

Бит DDDn регистра DDRD выбирает направление передачи данных. Если бит установлен (1), вывод сконфигурирован как выход. Если бит сброшен (0) - вывод сконфигурирован как вход. Если PORTDn установлен и вывод сконфигурирован как вход, включается КМОП подтягивающий резистор. Для отключения резистора, PORTDn должен быть сброшен (0) или вывод должен быть сконфигурирован как выход.

Таблица 11. Влияние DDDn на выводы порта D

DDDn	PORTDn	Вх/Вых	Подт.резист	Комментарий
0	0	Вход	Нет	Третье состояние (Hi-Z)
0	1	Вход	Да	PDn источник тока I <sub>IL</sub> , если извне соединен с землей
1	0	Выход	Нет	Выход установлен в 0
1	1	Выход	Нет	Выход установлен в 1

n = 6...0 - номер вывода

Альтернативные функции порта D

T0 - Порт D, бит 4 - тактовый вход таймера/счетчика 0. Подробнее см. описание таймера.

INT0 - Порт D, бит 2 - вход внешних прерываний. Подробнее см. описание прерываний.



# ПРОГРАММИРОВАНИЕ ПАМЯТИ

## Программирование битов блокировки памяти

Микроконтроллер AT90S1200 имеет два бита блокировки, которые могут быть оставлены незапрограммированными (1) или программироваться (0), при этом достигаются свойства приведенные в таблице 12.

Таблица 12. Режимы защиты и биты блокировки

Биты блокировки			Тип защиты
Режим	LB1	LB2	
1	1	1	защита не установлена
2	0	1	дальнейшее программирование флэш памяти запрещено
3	0	0	как режим 2, но запрещено и чтение

Примечание: биты блокировки стираются только при полном стирании памяти

## Биты конфигурации (Fuse bits)

В AT90S1200 предусмотрено два бита конфигурации - RCEN и SPIEN. Когда бит RCEN запрограммирован (0), микроконтроллер тактируется от встроенного RC генератора. По умолчанию этот бит стерт (1). Микросхемы AT90S1200А поставляются с запрограммированным битом RCEN.

Когда запрограммирован бит SPIEN (0) разрешен режим последовательного программирования. По умолчанию бит запрограммирован (0).

Эти биты недоступны при последовательном программировании и не изменяются при стирании памяти.

## Код устройства

Все микроконтроллеры фирмы Atmel имеют 3-байтовый сигнатурный код, по которому идентифицируется устройство. Этот код может быть прочитан в параллельном и последовательном режимах. Эти три байта размещены в отдельном адресном пространстве и для AT90S1200 имеют следующее значение:

1. \$000: \$1E - код производителя - Atmel
2. \$001: \$90 - 1 кБ флэш памяти
3. \$002: \$01 - при \$001=\$90 - м/сх AT90S1200

## Программирование флэш памяти и EEPROM

AT90S1200 имеет 1кБ перепрограммируемой флэш памяти программ и 64 байта энергонезависимой памяти данных.

При поставке AT90S1200 флэш память и память данных стерты (содержат \$FF) и готовы к программированию. Эта микросхема поддерживает высоковольтный (12В) параллельный режим программирования и низковольтный режим последовательного программирования. Напряжение +12В используется только для разрешения программирования, этот вывод не потребляет тока. Последовательный режим программирования предусмотрен для загрузки программы и данных в AT90S1200 в системе пользователя.

Память программ и данных в AT90S1200 программируются байт за байтом в обоих режимах программирования. Для энергонезависимой памяти предусмотрен режим автоматического стирания при последовательном программировании.

## Параллельное программирование.

Ниже рассмотрено параллельное программирование флэш памяти программ, энергонезависимой памяти данных, битов блокировки и конфигурации. Некоторые выводы AT90S1200 ниже называются именами, отражающими их функциональное назначение сигналов при параллельном программировании. Выводы, не приведенные в следующей таблице, называются своими обычными именами.

Таблица 13. Имена выводов

Имя сигнала при программирован.	Имя вывода	Вх/ Вых	Функция
RDY/BSY	PD1	Вых	0:м/сх занята программированием 1:м/сх готова к приему команды
OE	PD2	Вх	Разрешение выходов (активный 0)
WR	PD3	Вх	Импульс записи (активный 0)
BS	PD4	Вх	Выбор байта
XA0	PD5	Вх	Действие XTAL бит 0
XA1	PD6	Вх	Действие XTAL бит 1

Биты XA0 и XA1 определяют действие происходящее по положительному импульсу XTAL1. Установки битов приведены в следующей таблице:

Таблица 14. Установка XA1 и XA0

XA1	XA0	Действие при подаче импульса XTAL1
0	0	Загрузка адреса памяти программ или данных (старший/младший байт задается выводом BS)
0	1	Загрузка данных (старший/младший байт для флэш памяти задается выводом BS)
1	0	Загрузка команды
1	1	Не работает, холостой ход

По импульсу WR или OE загружается команда, определяющая действие по вводу или выводу. В байте команды каждому биту присвоена функция, как показано в таблице 15

Таблица 15. Биты конфигурации командного байта

Бит	Значение при установке
7	Стирание кристалла
6	Запись битов конфигурации, расположенных в следующих позициях байта данных: D5 - SPIEN, D0 - RCEN (0 для программирования, 1 для стирания)
5	Запись битов блокировки, расположенных в следующих позициях байта данных: D2 - LB2, D1 - LB1 (для программирования устанавливать 0)
4	Запись памяти программ или данных (определяется битом 0)
3	Чтение сигнатуры
2	Чтение битов блокировки и конфигурации, расположенных в следующих позициях байта данных: D7 - LB1, D6 - LB2, D5 - SPIEN, D0 - RCEN
1	Чтение памяти программ или данных (определяется битом 0)
0	0 - доступ к памяти программ, 1 - к памяти данных

## ВХОД В РЕЖИМ ПРОГРАММИРОВАНИЯ

Следующий алгоритм переводит устройство в режим параллельного программирования:

1. Подать напряжение 4.5...5.5 В на выводы питания
2. Установить RESET и BS в 0 и выдержать не меньше 100 нс.
3. Подать 12В на RESET и выждать не менее 100 нс перед изменением BS.

## СТИРАНИЕ КРИСТАЛЛА

При стирании кристалла стираются память программ и данных, а также биты блокировки. Биты блокировки не сбрасываются до полного стирания памяти программ и данных. Биты конфигурации не изменяются. (Стирание кристалла производится перед его программированием).

Загрузка команды "стереть кристалл"

1. Установить XA1, XA0 в '10'. Это разрешает загрузку команды.
2. Установить BS в 0
3. Установить PB в '1000 0000'. Это команда стирания кристалла.
4. Подать положительный импульс на XTAL1. При этом загружается команда и начинается стирание памяти программ и данных. После импульса XTAL1, подать отрицательный импульс на WR, чтобы разрешить стирание битов блокировки по окончании цикла стирания. Подождать около 10 мс до окончания цикла стирания. Стирание кристалла не генерирует сигнала RDY/BSY.

## ПРОГРАММИРОВАНИЕ ФЛЭШ ПАМЯТИ

Загрузить команду "программирование флэш памяти"

1. Установить XA1, XA0 в '10'. Это разрешает загрузку команды.
2. Установить BS в 0
3. Установить PB в '0001 0000'. Это команда программирования флэш памяти
4. Подать положительный импульс на XTAL1. При этом загружается команда.

Загрузить младший байт адреса

1. Установить XA1, XA0 в '00'. Это разрешает загрузку адреса.
2. Установить BS в 0. Это выбирает младший байт адреса.
3. Установить на PB младший байт адреса.
4. Подать положительный импульс на XTAL1. При этом загружается младший байт адреса.

Загрузить старший байт адреса

1. Установить XA1, XA0 в '00'. Это разрешает загрузку адреса.
2. Установить BS в 1. Это выбирает старший байт адреса.
3. Установить на PB старший байт адреса. (\$00.\$01)
4. Подать положительный импульс на XTAL1. При этом загружается старший байт адреса.

Загрузить байт данных

1. Установить XA1, XA0 в '01'. Это разрешает загрузку данных.
2. Установить на PB младший байт данных.
3. Подать положительный импульс на XTAL1. При этом загружается младший байт данных.

Запись младшего байта данных.

1. Установить BS в '0'. Это выбирает младший байт данных
2. Подать на WR отрицательный импульс. Это инициирует программирование байта. RDY/BSY переходит в низкое состояние.
3. Перед программированием следующего байта подождать, пока RDY/BSY перейдет в высокое состояние.

Загрузить байт данных

1. Установить XA1, XA0 в '01'. Это разрешает загрузку данных.
2. Установить на PB старший байт данных.
3. Подать положительный импульс на XTAL1. При этом загружается старший байт данных.

Запись старшего байта данных.

1. Установить BS в '1'. Это выбирает старший байт данных
2. Подать на WR отрицательный импульс. Это инициирует программирование байта. RDY/BSY переходит в низкое состояние.
3. Перед программированием следующего байта подождать, пока RDY/BSY перейдет в высокое состояние.

Загруженные адрес и данные сохраняются в устройстве после программирования, при этом процесс программирования упрощается.

\* Команду программирования флэш памяти необходимо подать только перед программированием первого байта

\* Старший байт адреса необходимо менять только перед программированием следующей страницы памяти программ (256 слов).

## ПРОГРАММИРОВАНИЕ ПАМЯТИ ДАННЫХ

Алгоритм программирования памяти данных следующий (обратитесь к программированию памяти программ за описанием загрузки команды, адреса и данных):

1. Загрузить команду '0001 0001'.
2. Загрузить младший адрес памяти данных (\$00 - \$3F).
3. Загрузить младший байт данных
4. Подать отрицательный импульс на WR и подождать перевода RDY/BSY в 1.

Загрузка команды необходима только перед программированием первого байта.

## ЧТЕНИЕ ПАМЯТИ ПРОГРАММ

Алгоритм чтения флэш памяти следующий (обратитесь к программированию памяти программ за описанием загрузки команды, адреса и данных):

1. Загрузить команду '0000 0010'
2. Загрузить младший байт адреса (\$00 - \$FF)
3. Загрузить старший байт адреса (\$00 - \$01)
4. Установить OE в '0', BS в '0'. Теперь на выводах PB можно прочесть младший байт данных.
5. Установить BS в '1'. Теперь на выводах PB можно прочесть старший байт данных.
6. Установить OE в '1'.

Загрузка команды необходима только перед чтением первого байта.

## ЧТЕНИЕ ПАМЯТИ ДАННЫХ

Алгоритм чтения памяти данных следующий (обратитесь к программированию памяти программ за описанием загрузки команды, адреса и данных):

1. Загрузить команду '0000 0011'
2. Загрузить младший байт адреса (\$00 - \$3F)
3. Установить OE в '0', BS в '0'. Теперь на выводах PB можно прочесть байт данных.
4. Установить OE в '1'.

Загрузка команды необходима только перед чтением первого байта.

## ПРОГРАММИРОВАНИЕ БИТОВ КОНФИГУРАЦИИ

Алгоритм программирования битов конфигурации следующий (обратитесь к программированию памяти программ за описанием загрузки команды, адреса и данных):

1. Загрузить команду '0100 0000'
2. Загрузить данные
  - Бит5=0 - запрограммировать бит SPIEN, Бит5=1 - стереть бит SPIEN
  - Бит0=0 - запрограммировать бит RCEN, Бит0=1 - стереть бит RCEN
3. Подать на WR отрицательный импульс и ждать перехода RDY/BSY в единицу.

**ВНИМАНИЕ!** WR должен удерживаться в низком состоянии не меньше 1мс.

## ПРОГРАММИРОВАНИЕ БИТОВ БЛОКИРОВКИ

Алгоритм программирования битов блокировки следующий (обратитесь к программированию памяти программ за описанием загрузки команды, адреса и данных):

1. Загрузить команду '0010 0000'
2. Загрузить данные
  - Бит2=0 - запрограммировать бит LB2
  - Бит1=0 - запрограммировать бит LB1
3. Подать на WR отрицательный импульс и ждать перехода RDY/BSY в единицу.

Биты блокировки стираются только при стирании всей микросхемы.

## ЧТЕНИЕ БИТОВ КОНФИГУРАЦИИ И БЛОКИРОВКИ

Алгоритм чтения битов блокировки и конфигурации следующий (обратитесь к программированию памяти программ за описанием загрузки команды, адреса и данных):

1. Загрузить команду '0000 0100'
2. Установить OE в '0', BS в '1'. Теперь на выводах PB можно прочесть биты блокировки и конфигурации.
  - Бит7 - LB1 - '0' запрограммирован
  - Бит6 - LB2 - '0' запрограммирован
  - Бит5 - SPIEN - '0' запрограммирован
  - Бит0 - RCEN - '0' запрограммирован
3. Установить OE в '1'.

Следите чтобы бит BS был установлен в '1'.

## ПОСЛЕДОВАТЕЛЬНАЯ ЗАГРУЗКА

Как память программ, так и память данных могут быть запрограммированы с использованием последовательной шины SPI, при этом вывод RESET должен быть подключен к земле. Последовательный интерфейс работает с выводами SCK, MOSI (вход) и MISO (выход). После подачи низкого уровня на RESET перепрограммированием/стиранием необходимо исполнить команду разрешения программирования.

При программировании памяти данных, во внутренний алгоритм программирования встроен цикл стирания (только при последовательном программировании), поэтому нет необходимости в выполнении команды стирания памяти. Команда стирания микросхемы переводит все ячейки памяти программ и данных в состояние \$FF.

Флэш память программ и энергонезависимая память данных имеют отдельное адресное пространство: \$000 - \$3FF для памяти программ и \$000 - \$03F для памяти данных.

При программировании необходимо подавать внешнюю тактовую частоту на вывод XTAL1 или подключить внешний тактовый генератор к выводам XTAL1 и XTAL2. Минимальные длительности низкого и высокого уровня сигнала SCK определены следующим образом:

- Низкий: > 1 период XTAL1
- Высокий: > 4 периодов XTAL1

### АЛГОРИТМ ПОСЛЕДОВАТЕЛЬНОГО ПРОГРАММИРОВАНИЯ

Для программирования и проверки AT90S1200 в режиме последовательного программирования рекомендуется следующая последовательность действий (см. формат четырех байтовой команды в табл.16):

1. Последовательность включения питания:  
Подать напряжение питания между VCC и GND, при этом RESET и SCK должны быть установлены в '0'. (Если программатор не гарантирует установки SCK в '0' при подаче питания, после того как на SCK будет установлен '0', на RESET необходимо подать положительный импульс.) Если кварцевый резонатор не подключен к выводам XTAL1 и XTAL2, подайте частоту от 0 до 16 МГц на вывод XTAL1.
2. Подождать 20 мс и разрешить последовательное программирование, послав команду разрешения на вывод MOSI/PB5. Выше указаны минимальные длительности положительного и отрицательного полупериодов сигнала SCK.
3. Если необходимо стирать микросхему (всегда при программировании памяти программ), подождать 10 мс, выдать положительный импульс на RESET и повторить шаг 2.
4. Память программ и память данных программируются по одному байту выдачей адреса и данных в команде записи. Ячейки памяти данных перед записью новых данных автоматически стираются. Следующий байт можно записывать через 4 мс.
5. Любую ячейку памяти можно проверить, используя команду чтения, которая выдает содержимое указанной ячейки на последовательный вывод MISO/PB6.  
По окончании программирования вывод RESET может быть установлен в '1' для возобновления нормальной работы схемы.
6. Последовательность выключения питания (если необходимо):  
Установить XTAL1 в '0' (если не используется кварцевый резонатор)  
Установить RESET в '1'.  
Отключить питание

Таблица 16. Команды последовательного программирования для AT90S1200

Команда	Формат команды				Действие
	Байт 1	Байт 2	Байт 3	Байт 4	
Разрешение программир.	1010 1100	0101 0011	xxxx xxxx	xxxx xxxx	Разрешен послед. программирования после установки RESET в '0'
Стирание микросх.	1010 1100	100x xxxx	xxxx xxxx	xxxx xxxx	Стирание памяти прог. и данных
Чтение памяти прог.	0010 H000	0000 000a	bbbb bbbb	oooo oooo	Читать старший или младший байт о по адресу ab
Запись памяти прог.	0100 H000	0000 000a	bbbb bbbb	iiii iiii	Запись старш. или младш. байта i в память прог по адресу ab
Чтение памяти данных	1010 0000	0000 0000	xxbb bbbb	oooo oooo	Читать данные о по адресу b
Запись памяти данных	1100 0000	0000 0000	xxbb bbbb	iiii iiii	Запись данных i по адресу b
Запись битов блокир.	1010 1100	111x x21x	xxxx xxxx	xxxx xxxx	Запись битов бло кировки. Установить 1,2=0 для блокировки памяти
Читать код устройства	0011 0000	xxxx xxxx	xxxx xxbb	oooo oooo	Читать код устройства о с адреса b

a - старшие биты адреса; b - младшие биты адреса; o - выход данных  
H = '0'-младший байт, '1'-старший байт; i - вход данных;  
x - не используется; 1 - 1-й бит блокировки; 2 - 2-й бит блокировки

При записи последовательных данных в AT90S1200 данные читаются по нарастающему фронту сигнала CLK. При чтении данных из AT90S1200 данные читаются по спадающему фронту сигнала CLK.

## МАКСИМАЛЬНО ДОПУСТИМЫЕ ПАРАМЕТРЫ

Рабочая температура .....	-55°C - +125°C
Температура хранения .....	-65°C - +150°C
Напряжение на любом выводе кроме RESET .....	-1.0V - +7.0V
Максимальное рабочее напряжение .....	6.6V
Постоянный ток через вывод порта .....	40.0 мА
Постоянный ток между VCC и GND .....	140.0 мА

ПРИМЕЧАНИЕ: выход параметров за пределы указанные в таблице может привести к нарушению работоспособности микросхемы. Это предельные значения параметров, рабочие параметры микросхемы приведены ниже. Удержание предельных значений на выводах м/сх в течение длительного времени может привести к потере работоспособности м/сх.

## ХАРАКТЕРИСТИКИ ПО ПОСТОЯННОМУ ТОКУ

Ta = -40...+85°C, Vcc=2.7...6.0V (если не указано иначе)

Параметр	Мин.	Тип.	Макс.	Ед. изм.	
Входное напряжение '0'	-0.5		0.3 Vcc	В	
Входное напряжение '1' (кроме XTAL1 и RESET)	0.7 Vcc		Vcc+0.5	В	
Входное напряжение '1' на XTAL1 и RESET	0.7 Vcc		Vcc+0.5	В	
Выходное напряжение '0' <sup>(1)</sup> (Порты В и D)	I <sub>IL</sub> =25mA, Vcc = 5V I <sub>IL</sub> =15mA, Vcc = 3V		0.5	В	
Выходное напряжение '1' (Порты В и D)	I <sub>OH</sub> =3mA, Vcc = 5V I <sub>OH</sub> =3mA, Vcc = 3V	Vcc-0.5		В	
Выходной ток (Порты В и D)	Vcc=5V, V <sub>OH</sub> = 4.5V Vcc=3V, V <sub>OH</sub> = 2.7V		4 2	мА	
Поглощаемый ток (Порты В и D)	Vcc=5V, V <sub>IL</sub> = 0.5V Vcc=3V, V <sub>IL</sub> = 0.3V		28 11	мА	
Подтягивающий резистор сброса	100		500	кОм	
Подтягивающий резистор вывода порта	35		120	кОм	
Потребляемый ток:	Активный режим, 3V, 4 МГц		2	3	мА
	холостой ход (idle mode), 3V, 4 МГц		0.7	1	мА
	пониженное потребление <sup>(2)</sup> WDT включен, 3V		10	20	мкА
	пониженное потребление <sup>(2)</sup> WDT выключен, 3V		0.15	2	мкА
Напряжение смещения аналогового компаратора Vcc = 5V		±20		мВ	
Входной ток утечки аналогового компаратора V <sub>IN</sub> = 1 В		±10	±30	нА	
Время срабатывания аналогового компаратора	Vcc=2.7V		750		нс
	Vcc=4.0V		500		

ПРИМЕЧАНИЯ:

- В рабочем состоянии ток через выводы должен ограничиваться следующими условиями:
  - Максимальный ток через вывод - 20 мА
  - Максимальный ток через все выводы - 80 мА
- Минимальное напряжение для режима пониженного потребления - 2V

## ПАРАМЕТРЫ ВНЕШНЕГО ТАКТОВОГО СИГНАЛА

Параметр	V <sub>cc</sub> =2.7...6.0 V		V <sub>cc</sub> =4.0...6.0 V		Ед.
	мин.	макс.	мин.	макс.	
Частота	0	4	0	16	МГц
Период	250		62.5		нс
Длительность '1'	40		16.7		нс
Длительность '0'	40		16.7		нс
Длительность фронта		10		4.15	нс

## ИНФОРМАЦИЯ ДЛЯ ЗАКАЗА

Частота	Напряж. питания	Маркировка	Корпус	Диапазон температур
4 МГц	2.7...6.0 V	AT90S1200-4PC AT90S1200-4SC AT90S1200-4YC	20P3 20S 20Y	Коммерческий (0°C ... 70°C)
		AT90S1200-4PI AT90S1200-4SI AT90S1200-4YI	20P3 20S 20Y	Промышленный (-40°C ... 85°C)
16 МГц	4.0...6.0 V	AT90S1200-16PC AT90S1200-16SC AT90S1200-16YC	20P3 20S 20Y	Коммерческий (0°C ... 70°C)
		AT90S1200-16PI AT90S1200-16SI AT90S1200-16YI	20P3 20S 20Y	Промышленный (-40°C ... 85°C)

Корпус 20P3 - PDIP  
20S - SOIC  
20Y - SSOP

Микросхемы с суффиксом "А" (AT90S1200А-XXXX) поставляются с запрограммированным битом RCEN.

## СПИСОК РЕГИСТРОВ

Адрес	Название	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Стр.
\$3F	SREG	I	T	H	S	V	N	Z	C	7
\$3E		ЗАРЕЗЕРВИРОВАН								
\$3D		ЗАРЕЗЕРВИРОВАН								
\$3C		ЗАРЕЗЕРВИРОВАН								
\$3B	GIMSK	-	INT0	-	-	-	-	-	-	9
\$3A		ЗАРЕЗЕРВИРОВАН								
\$39	TIMSK	-	-	-	-	-	-	TOIE0	-	9
\$38	TIFR	-	-	-	-	-	-	TOV0	-	9
\$37		ЗАРЕЗЕРВИРОВАН								
\$36		ЗАРЕЗЕРВИРОВАН								
\$35	MCUCR	-	-	SE	SM	-	-	ISC01	ISC00	10
\$34		ЗАРЕЗЕРВИРОВАН								
\$33	TCCR0	-	-	-	-	-	CS02	CS01	CS00	11
\$32	TCNT0	Таймер / счетчик (8 бит)								12
\$31		ЗАРЕЗЕРВИРОВАН								
...		ЗАРЕЗЕРВИРОВАНЫ								
\$22		ЗАРЕЗЕРВИРОВАН								
\$21	WDTCR	-	-	-	-	WDE	WDP2	WDP1	WDP0	12
\$20		ЗАРЕЗЕРВИРОВАН								
\$1F		ЗАРЕЗЕРВИРОВАН								
\$1E	EEAR	Регистр адреса энергонезависимой памяти								12
\$1D	EEDR	Регистр данных энергонезависимой памяти								13
\$1C	EEDR	-	-	-	-	-	-	EEWE	EERE	13
\$1D		ЗАРЕЗЕРВИРОВАН								
\$1A		ЗАРЕЗЕРВИРОВАН								
\$19		ЗАРЕЗЕРВИРОВАН								
\$18	PORTB	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	14
\$17	DDRB	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	14
\$16	PINB	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	15
\$15		ЗАРЕЗЕРВИРОВАН								
\$14		ЗАРЕЗЕРВИРОВАН								
\$13		ЗАРЕЗЕРВИРОВАН								
\$12	PORTD	-	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0	16
\$11	DDRD	-	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	16
\$10	PIND	-	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	16
\$0F		ЗАРЕЗЕРВИРОВАН								
...		ЗАРЕЗЕРВИРОВАНЫ								
\$09		ЗАРЕЗЕРВИРОВАН								
\$08	ACSR	ACD	-	ACO	ACI	ACIE	-	ACIS1	ACIS0	13
\$07		ЗАРЕЗЕРВИРОВАН								
...		ЗАРЕЗЕРВИРОВАНЫ								
\$00		ЗАРЕЗЕРВИРОВАН								



# Набор команд AT90S1200

Мнемон.	Операнды	Описание	Действие	Флаги	Цкл
<u>АРИФМЕТИЧЕСКИЕ И ЛОГИЧЕСКИЕ КОМАНДЫ</u>					
ADD	Rd, Rr	Сложить два регистра	$Rd \leftarrow Rd + Rr$	Z,C,N,V,H	1
ADC	Rd, Rr	Сложить с переносом	$Rd \leftarrow Rd + Rr + C$	Z,C,N,V,H	1
SUB	Rd, Rr	Вычесть два регистра	$Rd \leftarrow Rd - Rr$	Z,C,N,V,H	1
SUBI	Rd, K	Вычесть константу	$Rd \leftarrow Rd - K$	Z,C,N,V,H	1
SBC	Rd, Rr	Вычесть с переносом	$Rd \leftarrow Rd - Rr - C$	Z,C,N,V,H	1
SBCI	Rd, K	Вычесть с переносом	$Rd \leftarrow Rd - K - C$	Z,C,N,V,H	1
AND	Rd, Rr	Логическое И	$Rd \leftarrow Rd \text{ AND } Rr$	Z,N,V	1
ANDI	Rd, K	Логическое И	$Rd \leftarrow Rd \text{ AND } K$	Z,N,V	1
OR	Rd, Rr	Логическое ИЛИ	$Rd \leftarrow Rd \text{ OR } Rr$	Z,N,V	1
ORI	Rd, K	Логическое ИЛИ	$Rd \leftarrow Rd \text{ OR } K$	Z,N,V	1
EOR	Rd, Rr	Исключающее ИЛИ	$Rd \leftarrow Rd \text{ XOR } Rr$	Z,N,V	1
COM	Rd	Дополнение до 1	$Rd \leftarrow \$FF - Rd$	Z,C,N,V	1
NEG	Rd	Дополнение до 2	$Rd \leftarrow \$00 - Rd$	Z,C,N,V,H	1
SBR	Rd, K	Установка бит(-ов) в регистре	$Rd \leftarrow Rd \text{ OR } K$	Z,N,V	1
CBR	Rd, K	Сброс бит(-ов) в регистре	$Rd \leftarrow Rd \text{ AND } (FFh - K)$	Z,N,V	1
INC	Rd	Увеличить на 1	$Rd \leftarrow Rd + 1$	Z,N,V	1
DEC	Rd	Уменьшить на 1	$Rd \leftarrow Rd - 1$	Z,N,V	1
TST	Rd	Проверить на 0 или минус	$Rd \leftarrow Rd \text{ AND } Rd$	Z,N,V	1
CLR	Rd	Очистить регистр	$Rd \leftarrow Rd \text{ XOR } Rd$	Z,N,V	1
SER	Rd	Установить регистр	$Rd \leftarrow \$FF$	Не меняет	1

## КОМАНДЫ ВЕТВЛЕНИЯ

RJMP	k	Относительный переход	$PC \leftarrow PC+k+1$	Не меняет	2
RCALL	k	Относительный вызов подпрограммы	$PC \leftarrow PC+k+1$	Не меняет	3
RET		Выход из подпрограммы	$PC \leftarrow \text{STACK}$	Не меняет	4
RETI		Выход из прерывания	$PC \leftarrow \text{STACK}$	I	4
CPSE	Rd, Rr	Сравнить, пропуск если равно	$\text{if}(Rd=Rr) PC \leftarrow PC+2$ или 3	Не меняет	1/2
CP	Rd, Rr	Сравнить	$Rd - Rr$	Z,N,V,C,H	1
CPC	Rd, Rr	Сравнить с переносом	$Rd - Rr - C$	Z,N,V,C,H	1
CPI	Rd, K	Сравнить с константой	$Rd - K$	Z,N,V,C,H	1
SBRC	Rr, b	Пропуск если бит в регистре сброшен	$\text{if}(Rr(b)=0) PC \leftarrow PC+2$ или 3	Не меняет	1/2
SBRS	Rr, b	Пропуск если бит в регистре установлен	$\text{if}(Rr(b)=1) PC \leftarrow PC+2$ или 3	Не меняет	1/2
SBIC	P, b	Пропуск если бит в рег. ввода/выв. сброшен	$\text{if}(P(b)=0) PC \leftarrow PC+2$ или 3	Не меняет	1/2
SBIS	P, b	Пропуск если бит в рег. ввода/выв. установлен	$\text{if}(P(b)=1) PC \leftarrow PC+2$ или 3	Не меняет	1/2
BRBS	s, k	Переход если установлен флаг s	$\text{if}(SREG(s)=1) PC \leftarrow PC+k+1$	Не меняет	1/2
BRBC	s, k	Переход если сброшен флаг s	$\text{if}(SREG(s)=0) PC \leftarrow PC+k+1$	Не меняет	1/2
BREQ	k	Переход если равно	$\text{if}(Z=1) PC \leftarrow PC+k+1$	Не меняет	1/2
BRNE	k	Переход если неравно	$\text{if}(Z=0) PC \leftarrow PC+k+1$	Не меняет	1/2
BRCS	k	Переход если установлен перенос	$\text{if}(C=1) PC \leftarrow PC+k+1$	Не меняет	1/2
BRCC	k	Переход если сброшен перенос	$\text{if}(C=0) PC \leftarrow PC+k+1$	Не меняет	1/2
BRSH	k	Переход если равно или больше	$\text{if}(C=0) PC \leftarrow PC+k+1$	Не меняет	1/2
BRLO	k	Переход если меньше	$\text{if}(C=1) PC \leftarrow PC+k+1$	Не меняет	1/2
BRMI	k	Переход если минус	$\text{if}(N=1) PC \leftarrow PC+k+1$	Не меняет	1/2
BRPL	k	Переход если плюс	$\text{if}(N=0) PC \leftarrow PC+k+1$	Не меняет	1/2
BRGE	k	Переход если больше или равно, со знаком	$\text{if}(N \text{ XOR } V=0) PC \leftarrow PC+k+1$	Не меняет	1/2
BRLT	k	Переход если меньше нуля, со знаком	$\text{if}(N \text{ XOR } V=1) PC \leftarrow PC+k+1$	Не меняет	1/2
BRHS	k	Переход если установлен флаг H	$\text{if}(H=1) PC \leftarrow PC+k+1$	Не меняет	1/2
BRHC	k	Переход если сброшен флаг H	$\text{if}(H=0) PC \leftarrow PC+k+1$	Не меняет	1/2
BRTS	k	Переход если установлен флаг T	$\text{if}(T=1) PC \leftarrow PC+k+1$	Не меняет	1/2
BRTC	k	Переход если сброшен флаг T	$\text{if}(T=0) PC \leftarrow PC+k+1$	Не меняет	1/2
BRVS	k	Переход если установлен флаг V (переполнение)	$\text{if}(V=1) PC \leftarrow PC+k+1$	Не меняет	1/2
BRVC	k	Переход если сброшен флаг V	$\text{if}(V=0) PC \leftarrow PC+k+1$	Не меняет	1/2
BRIE	k	Переход если разрешены прерывания	$\text{if}(I=1) PC \leftarrow PC+k+1$	Не меняет	1/2
BRID	k	Переход если запрещены прерывания	$\text{if}(I=0) PC \leftarrow PC+k+1$	Не меняет	1/2

Мнемон.	Операнды	Описание	Действие	Флаги	Цкл
КОМАНДЫ ПЕРЕСЫЛКИ					
LD	Rd,Z	Загрузить регистр непосредственно	$Rd \leftarrow (Z)$	Не меняет	2
ST	Z,Rr	Записать регистр непосредственно	$(Z) \leftarrow Rr$	Не меняет	2
MOV	Rd, Rr	Пересылка между регистрами	$Rd \leftarrow Rr$	Не меняет	1
LDI	Rd, K	Загрузить константу	$Rd \leftarrow K$	Не меняет	1
IN	Rd, P	Ввод из порта	$Rd \leftarrow P$	Не меняет	1
OUT	P, Rr	Вывод в порт	$P \leftarrow Rr$	Не меняет	1
КОМАНДЫ РАБОТЫ С БИТАМИ					
SBI	P,b	Установить бит в регистре ввода/вывода	$I/O(P,b) \leftarrow 1$	Не меняет	2
CBI	P,b	Сбросить бит в регистре ввода/вывода	$I/O(P,b) \leftarrow 0$	Не меняет	2
LSL	Rd	Логический сдвиг влево	$Rd(n+1) \leftarrow Rd(n), Rd(0) \leftarrow 0$	Z,C,N,V	1
LSR	Rd	Логический сдвиг вправо	$Rd(n) \leftarrow Rd(n+1), Rd(7) \leftarrow 0$	Z,C,N,V	1
ROL	Rd	Сдвиг влево через C	$Rd(0) \leftarrow C,$ $Rd(n+1) \leftarrow Rd(n), C \leftarrow Rd(7)$	Z,C,N,V	1
ROR	Rd	Сдвиг вправо через C	$Rd(7) \leftarrow C,$ $Rd(n) \leftarrow Rd(n+1), C \leftarrow Rd(0)$	Z,C,N,V	1
ASR	Rd	Арифметический сдвиг вправо	$Rd(n) \leftarrow Rd(n+1), n=0..6$	Z,C,N,V	1
SWAP	Rd	Обмен байт	$Rd(3-0) \leftarrow Rd(7-4),$ $Rd(7-4) \leftarrow Rd(3-0)$	Не меняет	1
BSET	s	Установить флаг	$SREG(s) \leftarrow 1$	SREG(s)	1
BCLR	s	Сбросить флаг	$SREG(s) \leftarrow 0$	SREG(s)	1
BST	Rr,b	Запомнить бит в T	$T \leftarrow Rr(b)$	T	1
BLD	Rd, b	Прочитать бит из T	$Rd(b) \leftarrow T$	Не меняет	1
SEC		Установить перенос	$C \leftarrow 1$	C	1
CLC		Сбросить перенос	$C \leftarrow 0$	C	1
SEN		Установить флаг N	$N \leftarrow 1$	N	1
CLN		Сбросить флаг N	$N \leftarrow 0$	N	1
SEZ		Установить флаг Z	$Z \leftarrow 1$	Z	1
CLZ		Сбросить флаг Z	$Z \leftarrow 0$	Z	1
SEI		Разрешить прерывания	$I \leftarrow 1$	I	1
CLI		Запретить прерывания	$I \leftarrow 0$	I	1
SES		Установить флаг S	$S \leftarrow 1$	S	1
CLS		Сбросить флаг S	$S \leftarrow 0$	S	1
SEV		Установить флаг V	$V \leftarrow 1$	V	1
CLV		Сбросить флаг V	$V \leftarrow 0$	V	1
SET		Установить флаг T	$T \leftarrow 1$	T	1
CLT		Сбросить флаг T	$T \leftarrow 0$	T	1
SEH		Установить флаг H	$H \leftarrow 1$	H	1
CLH		Сбросить флаг H	$H \leftarrow 0$	H	1
NOP		Нет операции		Не меняет	1
SLEEP		Останов		Не меняет	3
WDR		Сброс сторожевого таймера		Не меняет	1