

## AVR-микроконтроллеры второго поколения: новые аппаратные возможности

Николай Королев korolev@argussoft.ru

Дмитрий Королев atmel@argussoft.ru

*За время своего относительно недолгого существования — появились на рынке в 1997 году — микроконтроллеры семейства AVR (серия AT90S) фирмы ATMEL приобрели чрезвычайную популярность в мире и в России. 2001 год стал переломным — более половины российских разработок с использованием 8-битных микроконтроллеров было создано именно на AVR. На сегодняшний день продолжается развитие в том же направлении. Это легко объяснить: благодаря очень удачной архитектуре ядра процессора и широкому набору периферийных модулей на кристалле существенно облегчается процесс программирования конечного устройства. Еще одним фактором, привлекательным для разработчиков, стал выпуск в 2002 году AVR-микроконтроллеров следующего поколения. Эти микросхемы производятся по технологическим нормам 0,35 мкм, в отличие от первых AT90S, выпускающихся по технологии 0,5 мкм. Заметное отличие — увеличенная в два раза тактовая частота новых контроллеров — 16 МГц, обеспечивающая производительность до 16 MIPS, (типовая команда у AVR-контроллеров выполняется за один период тактовой частоты), но это, как говорится, вершина айсберга.*

*Ниже рассматриваются основные аппаратные отличия новых микросхем.*

Фирма ATMEL ежегодно устраивает семинары для дистрибьюторов. Обычно семинар проходит в течение 5 рабочих дней. На семинаре, который проходил в конце марта 2003 года, AVR-микроконтроллерам было выделено более полутора дней — больше, чем любому другому направлению. Основную информацию по этим микроконтроллерам представлял Ингар Фредриксен (Ingar Fredriksen) (рис. 1), норвежское отделение ATMEL (стоит напомнить, что ядро этих RISC-микроконтроллеров разработали два норвежца, имена которых и заложены в аббревиатуре AVR).



Рис. 1. Ингар Фредриксен, сотрудник норвежского отделения ATMEL .

Когда фирма ATMEL начала выпуск принципиально нового семейства 8-битных микроконтроллеров, построенных на базе RISC-архитектуры с расширенным набором команд, она условно разбила их на три подсемейства: **tiny** (англ. - крошечный), **classic** и **mega**. Все микросхемы имеют идентичное ядро с регистровым файлом из тридцати двух 8-битных регистров общего назначения и встроенное FLASH-ПЗУ

емкостью от 1 до 128 килобайт. Обобщенная функциональная схема AVR-микроконтроллера представлена на рисунке2.

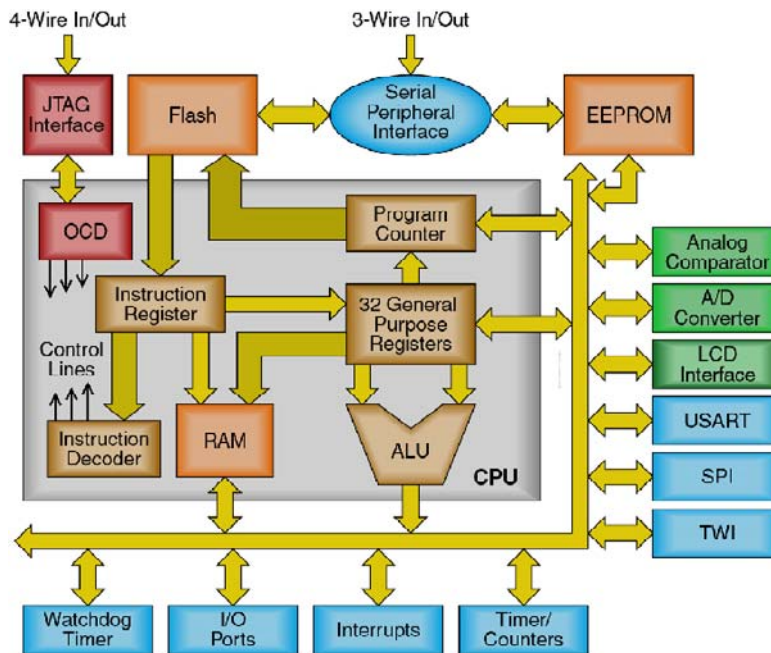


Рис. 2. Функциональная схема AVR

Ширина шины команд составляет 16 бит. 5-битное поле адреса регистра обеспечивает доступ к любому из 32 регистров общего назначения. В результате этих схемотехнических решений стало возможным отказаться от многобанковой организации регистров общего назначения и страничной адресации памяти. Помимо облегчения жизни программиста, пишущего на ассемблере, такая архитектура позволила создать высокоэффективные компиляторы языка Си, генерирующие компактный код.

Микросхемы в подсемействах в основном отличались объемом памяти и количеством портов ввода-вывода. Так, микроконтроллеры **tiny** имели объем FLASH-ПЗУ 1—2 кБайта и корпус 8... 20 выводов (за исключением tiny28), **classic** —объем FLASH-ПЗУ 2—8 кБайт и корпус 28... 44 вывода, а **mega** — 128 кБайт и корпус 64 вывода.

Все микросхемы позволяют производить перепрограммирование памяти программ непосредственно в системе.

С момента начала производства в 1997 году ежегодный объем выпуска AVR-микроконтроллеров неуклонно увеличивается. На рисунке 3 представлены соответствующие показатели. Для справки: в 2002 году было выпущено более 100 миллионов микросхем.

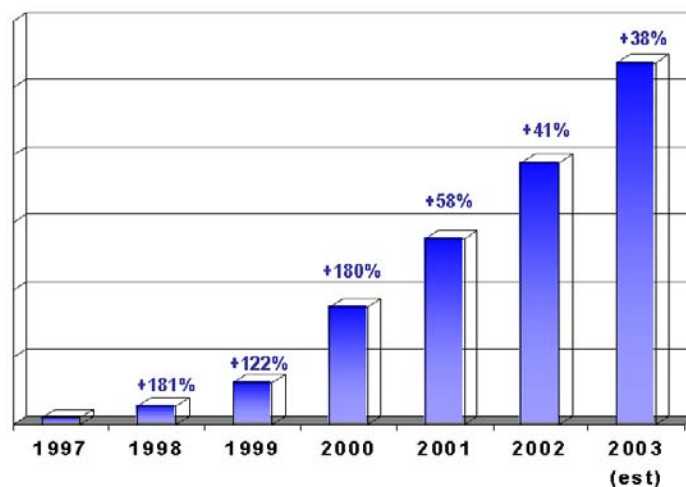


Рис. 3. Диаграмма выпуска микроконтроллеров по годам

С выпуском микроконтроллеров второго поколения в позиционировании подсемейств произошли изменения. Новые микросхемы содержат только два подсемейства: **tiny** и **mega**. Микросхемы **tiny** по-прежнему имеют FLASH-ПЗУ 1 — 2 кБайта и корпуса 8... 20 выводов, а вот ряд **mega** существенно расширился: FLASH-ПЗУ 8—128 кБайт и корпус 28... 64 вывода. Фактически произошло поглощение подсемейства **classic**. Краткие характеристики микросхем представлены в таблице 1.

Таблица 1.

<i>Микросхема</i>	<i>ПЗУ Программ, КВ</i>	<i>ОЗУ</i>	<i>ЭСПЗУ</i>	<i>Таймеры</i>	<i>Посл. порты</i>	<i>АЦП</i>	<i>Корпус выводы</i>	<i>Совместимость</i>
<b>ATtiny13</b>	1	64 b	64 b	2	нет	нет	8	tiny11/12/15
<b>ATtiny26/L</b>	2	128 b	128 b	2	USI	11 вх.	20	нет
<b>ATmega8/L</b>	8	1 KB	512 b	3	USART	6/8 вх.	28, 32	AT90S4433
<b>ATmega8515/L</b>	8	512 b	512 b	2	USART	нет	40, 44	AT90S8515
<b>ATmega8535/L</b>	8	512 b	512 b	2	USART	8 вх.	40, 44	AT90S8535
<b>ATmega16/L</b>	16	1 KB	512 b	3	USART	8 вх.	40, 44	ATmega163
<b>ATmega162/V</b>	16	1 KB	512 b	4	2 USART	нет	40, 44	ATmega161
<b>ATmega169/V</b>	16	1 KB	512 b	3	USART	8 вх.	40, 44	нет
<b>ATmega32/L</b>	32	2 KB	1 KB	3	USART	8 вх.	40, 44	ATmega323
<b>ATmega64/L</b>	64	4 KB	2 KB	4	2 USART	8 вх.	64	ATmega103
<b>ATmega128/L</b>	128	4 KB	4 KB	4	2 USART	8 вх.	64	ATmega103

Из анализа таблицы можно видеть следующее. Для каждой микросхемы **classic** есть «дублер» **mega** в идентичном корпусе. Таким образом, в существующей разработке, например, вместо AT90S8515 можно использовать микроконтроллер ATmega8515, при этом получив преимущества в виде и увеличения скорости процессора, добавления команд умножения и расширенных характеристик последовательных портов. Кстати, если тактовая частота контроллера в устройстве не превышает 8 МГц, рекомендуется использовать «дублер» «L» версии. Это объясняется тем, что диапазон напряжений питания этих микросхем составляет 2,7 ... 5,5 В, и таким образом, повышается помехозащищенность устройства: ему не будут страшны кратковременные (а в ряде случаев и долговременные) провалы в питании до уровня 2,7 В. Для каждого типа микроконтроллера существует файл, где детально описаны отличия кристаллов первого и второго поколений. Как правило, при замене микроконтроллера потребуется перекомпиляция исходного кода программы, однако, в большинстве случаев это является механической операцией замены строки с именем соответствующего подключаемого файла (в данном примере — 8515def.inc на m8515def.inc). В ряду новых AVR-микроконтроллеров отсутствует представитель с емкостью памяти программ 4 кб. Это объясняется тем, что младший кристалл в семействе **mega** — микроконтроллер ATmega8 с памятью программ 8 кб — имеет цену даже ниже, чем 4-килобайтный прототип — микроконтроллер AT90S4433.

В подсемействе **tiny** выпущена интересная микросхема второго поколения — ATtiny26. Этот кристалл при цене менее \$2 содержит высокоскоростной ШИМ, а также 11-канальный АЦП с возможностью работы в дифференциальном режиме, регулируемым входным усилителем и встроенным источником опорного напряжения.

Ниже приведены основные характеристики новых AVR-микроконтроллеров:

- тактовая частота до 16 МГц, время выполнения команды 62,5 нс;
- встроенный программируемый RC-генератор, частота 1, 2, 4, 8 МГц
- FLASH-ПЗУ программ, программируемое в системе, до 128 кБайт (10 000 циклов перезаписи);
- электрически перепрограммируемое ПЗУ данных до 4 кБайт (100 000 циклов перезаписи);
- внутреннее ОЗУ с временем доступа 1 такт, до 4 кБайт;
- мощный набор команд (более 120 инструкций);
- 6 аппаратных команд умножения (для семейства **mega**);
- развитая система адресации, оптимизированная для работы с Си-компиляторами;
- 32 регистра общего назначения (аккумулятора);
- синхронный (USART) или асинхронный (UART) последовательные порты;
- синхронный последовательных порт (SPI);
- двухпроводный интерфейс TWI, совместимый с интерфейсом I<sup>2</sup>C;
- многоканальный 8-, 9-, 10-, 16-битный ШИМ-модулятор;

- 10-битный АЦП со временем преобразования 70 мкс и дифференциальными входами;
- программируемый коэффициент усиления — 1; 10 или 200;
- встроенный источник опорного напряжения 2,56 В;
- аналоговый компаратор;
- сторожевой таймер;
- настраиваемая схема задержки запуска после подачи питания;
- схема слежения за уровнем напряжения питания;
- JTAG интерфейс для подключения эмулятора (в микросхемах с объемом ПЗУ от 16 кБ)
- микромощное потребление (менее 100 мкА в активном режиме на частоте 32 кГц)

Полное техническое описание AVR- микроконтроллера составляет несколько сотен страниц, поэтому в рамках данной статьи есть возможность поговорить только об основных особенностях. Архитектуру микроконтроллеров нового поколения удобно рассматривать на старшем представителе семейства — ATmega128, так как в нем присутствует весь набор периферийных модулей. Итак, чем интересен этот микроконтроллер?

### Система тактирования

Во-первых, ATmega128, как и другие AVR-микроконтроллеры второго поколения, имеют расширенную, достаточно гибкую структуру тактирования. В качестве формирователя тактового сигнала могут быть использованы следующие источники (рис. 4):

- высокочастотный или низкочастотный кварцевый резонатор;
- пьезорезонатор;
- внешняя RC-цепочка;
- внутренний настраиваемый RC-генератор;
- внешний источник тактовых импульсов.

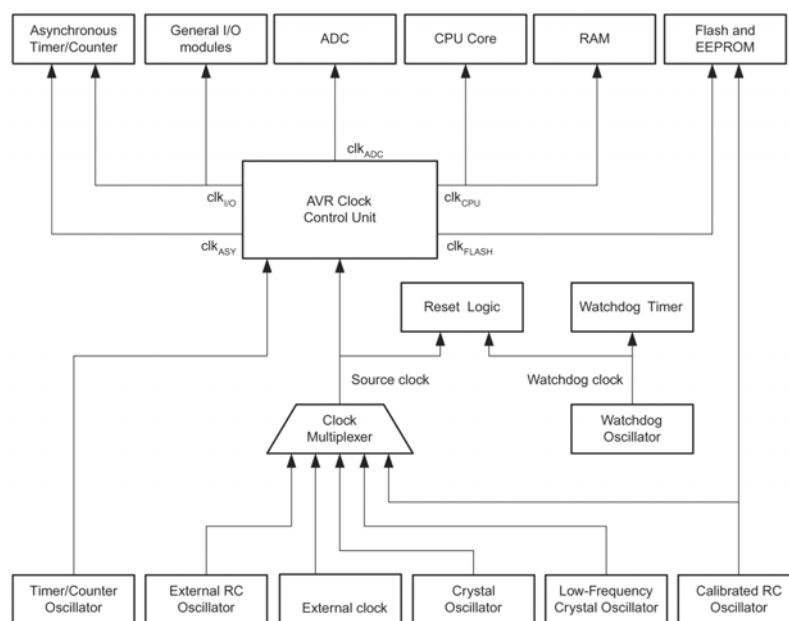


Рис. 4. Схема модуля тактирования

Из приведенной схемы видно, что внутри кристалла используется несколько цепей тактирования. Такое построение позволяет при необходимости отключать неиспользуемые цепи, что снижает потребляемый микросхемой ток.

Выбор способа тактирования производится установкой соответствующих установочных битов (fuse bits) CKSEL0...CKSEL3 при программировании микросхемы. Эти установочные биты управляют также задержкой включения микроконтроллера после выхода из состояния ожидания. В таблице 2 приведены варианты установок этих fuse-битов.

Таблица 2.

<i>CKSEL3...0</i>	<i>Источник тактирования</i>	<i>Комментарий</i>
0000	Внешняя частота	0... 16 МГц
0001	Встроенный RC-генератор	1 МГц
0010	Встроенный RC-генератор	2 МГц
0011	Встроенный RC-генератор	4 МГц
0100	Встроенный RC-генератор	8 МГц
0101	Внешний RC-генератор	< 0,9 МГц
0110	Внешний RC-генератор	0,9... 3,0 МГц
0111	Внешний RC-генератор	3,0... 8,0 МГц
1000	Внешний RC-генератор	8,0... 12 МГц
1001	Низкочастотный пьезорезонатор	32,768 кГц
101x	Кварцевый резонатор	0,4... 0,9 МГц
110x	Кварцевый резонатор	0,9... 3,0 МГц
111x	Кварцевый резонатор	3,0... 8,0 МГц

Такое многообразие возможностей подчас приводит к курьезам. Известно, к сожалению, что не все разработчики внимательно читают технические описания на новые микросхемы, что приводит к неверным действиям при программировании микросхем и необходимости исследования проблемы "а почему микросхема не работает".

Типичная ошибка — неверная установка fuse-битов CKSEL0... CKSEL3. В случае установки этих битов "в нули", т. е. 0000, микросхема переходит в режим работы от внешнего генератора. Таким образом, даже подключенный кварцевый или пьезорезонатор не "запускает" микросхему. Единственный выход "оживить" микроконтроллер — подать на вывод XTAL1 внешние импульсы с частотой 0,5... 16 МГц, не выпаивая его из схемы, и правильно переустановить fuse-биты CKSEL0... CKSEL3.

Попутно следует отметить, что микросхемы поставляются с fuse-битами, установленными для работы от встроенного RC-генератора с частотой 1 МГц, таким образом, простое подключение внешнего кварца не дает ожидаемого эффекта. Встроенный RC-генератор имеет достаточно хорошие характеристики, нестабильность по температуре составляет 3...4% в диапазоне -40 до +85 °С, нестабильность по питанию — в диапазоне от 3,3 до 5,0 В — 4... 5%.

На рис. 5 показано начальное, «заводское», состояние установочных битов микросхемы ATmega128.

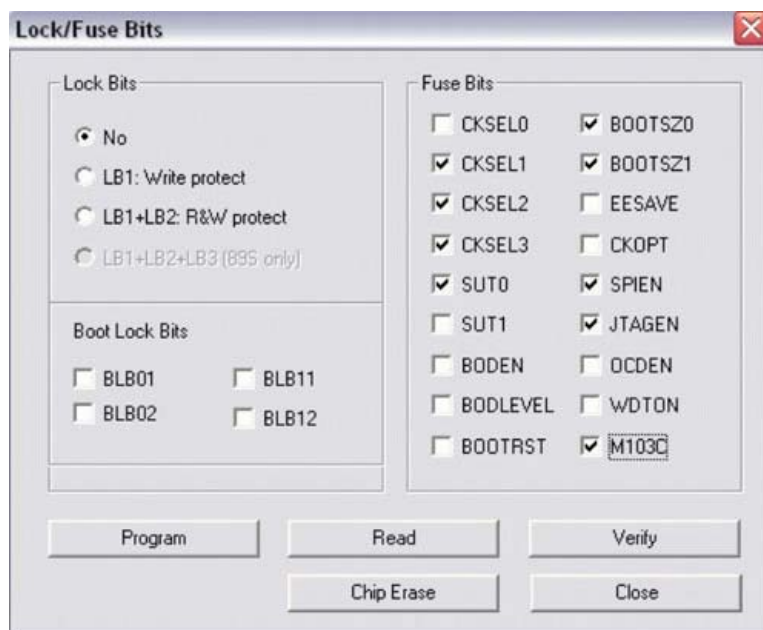


Рис. 5. Окно установочных битов

Рассмотрим более подробно набор fuse-битов. Биты защиты не требуют комментариев, их функции очевидны. Биты BLB0x BLB1x отвечают за защиту от чтения и модификации данных в основной памяти программ и загрузочном блоке соответственно.

Два бита SUT (startup time) определяют задержку пуска микроконтроллера. Это важно при использовании задающих кварцевых резонаторов с большим временем выхода на режим. Для таких кварцев можно установить задержку до 65 мс.

Биты BOD (brown out detector) и BODLEVEL управляют схемой слежения за питанием. Эта схема «наблюдает» за уровнем напряжения питания микроконтроллера, и, в случае снижения этого уровня ниже порога на время, более 2 мкс, производит перезапуск контроллера. Пороговое напряжение составляет 4,0 или 2,7 В, в зависимости от состояния бита BODLEVEL.

Биты BOOTSZ0 и BOOTSZ1 служат для настройки размера boot-блока, от 1 до 4 страниц памяти. Страница памяти в различных микросхемах имеет емкость от 256 Байт до 2 кБайт. Это позволяет более рационально использовать пространство памяти программ.

Бит SKOPT выполняет различные функции, в зависимости от типа используемого задающего генератора. Если это кварцевый резонатор, бит SKOPT отвечает за настройку коэффициента усиления встроенной схемы генератора. Если используется генератор с частотой выше 8 МГц, рекомендуется активизировать этот бит, если же в системе используется кварц на частоту 8 МГц и менее, бит SKOPT можно сбросить. В результате уменьшится потребляемый микроконтроллером ток и снизится электромагнитное излучение цепей генератора. Если же в качестве задающего генератора используется «часовой» пьезорезонатор на частоту 32,768 КГц, при программировании бита SKOPT к выводам XTAL1 XTAL2 подключаются внутренние конденсаторы номиналом 36 пФ, что снимает необходимость использования дополнительных компонентов.

Бит SPIEN разрешает последовательное программирование/чтение микросхемы, и биты JTAGEN OCDEN соответственно активизируют JTAG-интерфейс и разрешают работу встроенных отладочных цепей. Бит OCDEN требует внимательного к себе отношения, так как при его активном состоянии некоторые внутренние цепи тактирования не отключаются, что приводит к повышенному потреблению мощности в режиме ожидания.

Заключительный бит, M103C, следует рассмотреть более подробно. Как уже упоминалось выше, микросхема ATmega128 имеет на кристалле существенно больше периферийных модулей, чем ее прототип — микросхема ATmega103. Следовательно, на кристалле значительно увеличилось количество управляющих регистров — вместо 64 их стало 224. В традиционных микроконтроллерах в таком случае организовывалась дополнительная страница регистров, однако AVR-архитектура основывается на линейной адресации для ускорения доступа к данным. В соответствии с этой концепцией новый массив регистров размещен непосредственно за массивом базовых регистров, а начальный адрес внутреннего ОЗУ сдвинут на 160 Байт, что отображено на карте памяти (рис. 6).

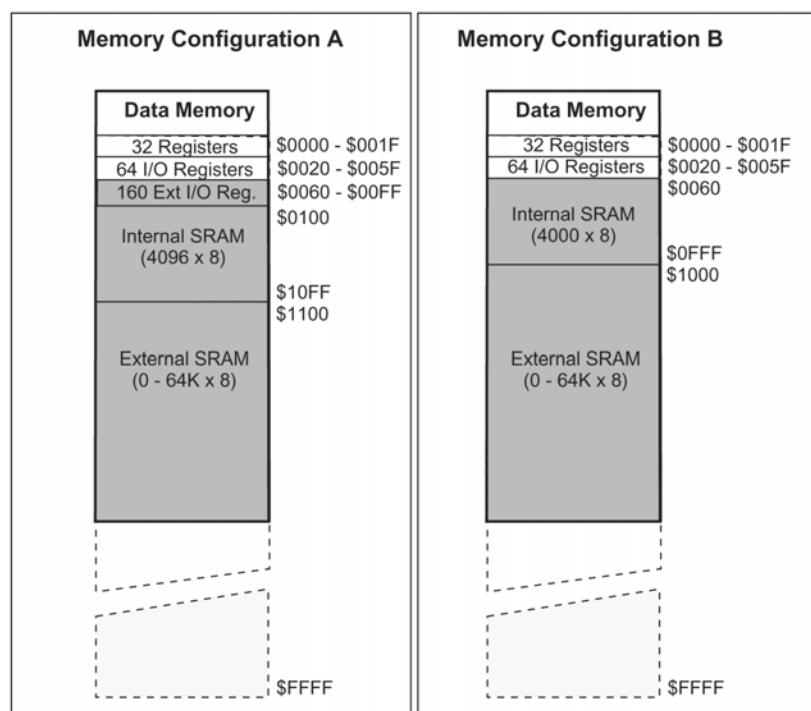


Рис. 6. Карта памяти

Конфигурация А соответствует микросхеме АТmega128, а конфигурация В — микросхеме АТmega103.

Если микросхему АТmega128 устанавливают в ранее разработанное устройство, и ее дополнительные функции не используются, следует установить бит М103С, и тогда она будет работать в режиме эмуляции АТmega103. При этом не надо забывать о правильной установке битов СКSEL0... СКSEL3!

Универсальность AVR-микроконтроллеров заключается в гибком построении кристалла — идентичное ядро и разнообразный набор периферийных модулей. Таким образом, можно использовать программы, написанные для одной микросхемы, в качестве основы программы другой, с более развитой периферией. Например, отлаженная программа, написанная для микросхемы АТtiny26, может служить основой программы для микросхемы АТmega32 — потребуется лишь добавить функции, использующие новые ресурсы и перекомпилировать исходный текст.

## Система прерываний

Система прерываний AVR-микроконтроллеров обслуживает несколько источников. В младших адресах памяти программ находится таблица прерываний, в которой записываются вектора для каждого из обслуживаемых прерываний. Каждое прерывание имеет индивидуальный бит разрешения, также есть бит общего разрешения прерываний. Вектор прерывания занимает один или два адреса памяти программ, в зависимости от размера памяти конкретного микроконтроллера (следует еще раз отметить, что шина команд — 16-разрядная). Прерывания, размещенные по младшим адресам, имеют более высокий приоритет, в нулевом адресе таблицы находится вектор прерывания «Сброс».

В обычном режиме после снятия сигнала низкого уровня на входе «Сброс» начинает выполняться программа по вектору, указанному в нулевом адресе. Если же установлен fuse-бит загрузчика BOOTRST, по сбросу стартует программа, размещенная в загрузочной области памяти микроконтроллера. Эта функция позволяет, например загружать в ПЗУ две различные программы, и управляя состоянием бита BOOTRST запускать любую из них без перепрограммирования всего объема Flash-памяти.

Ниже представлен фрагмент кода с указанием названий векторов прерываний микросхемы АТmega128. Из тридцати пяти векторов восемь (с именами EXT\_INT0 ... EXT\_INT7) являются внешними, остальные — внутренние.

```
$0000 jmp RESET ; Reset Handler
$0002 jmp EXT_INT0 ; IRQ0 Handler
$0004 jmp EXT_INT1 ; IRQ1 Handler
$0006 jmp EXT_INT2 ; IRQ2 Handler
$0008 jmp EXT_INT3 ; IRQ3 Handler
$000A jmp EXT_INT4 ; IRQ4 Handler
$000C jmp EXT_INT5 ; IRQ5 Handler
$000E jmp EXT_INT6 ; IRQ6 Handler
$0010 jmp EXT_INT7 ; IRQ7 Handler
$0012 jmp TIM2_COMP ; Timer2 Compare Handler
$0014 jmp TIM2_OVF ; Timer2 Overflow Handler
$0016 jmp TIM1_CAPT ; Timer1 Capture Handler
$0018 jmp TIM1_COMPA ; Timer1 CompareA Handler
$001A jmp TIM1_COMPB ; Timer1 CompareB Handler
$001C jmp TIM1_OVF ; Timer1 Overflow Handler
$001E jmp TIM0_COMP ; Timer0 Compare Handler
$0020 jmp TIM0_OVF ; Timer0 Overflow Handler
$0022 jmp SPI_STC ; SPI Transfer Complete Handler
$0024 jmp USART0_RXC ; USART0 RX Complete Handler
$0026 jmp USART0_DRE ; USART0,UDR Empty Handler
$0028 jmp USART0_TXC ; USART0 TX Complete Handler
$002A jmp ADC ; ADC Conversion Complete Handler
$002C jmp EE_RDY ; EEPROM Ready Handler
$002E jmp ANA_COMP ; Analog Comparator Handler
$0030 jmp TIM1_COMPC ; Timer1 CompareC Handler
$0032 jmp TIM3_CAPT ; Timer3 Capture Handler
$0034 jmp TIM3_COMPA ; Timer3 CompareA Handler
$0036 jmp TIM3_COMPB ; Timer3 CompareB Handler
$0038 jmp TIM3_COMPC ; Timer3 CompareC Handler
$003A jmp TIM3_OVF ; Timer3 Overflow Handler
$003C jmp USART1_RXC ; USART1 RX Complete Handler
$003E jmp USART1_DRE ; USART1,UDR Empty Handler
$0040 jmp USART1_TXC ; USART1 TX Complete Handler
$0042 jmp TWI ; Two-wire Serial Interface Interrupt Handler
```

```

$0044 jmp SPM_RDY ; SPM Ready Handler
;
$0046 RESET:ldir16, high(RAMEND); Main program start
$0047 out SPH,r16 ; Set stack pointer to top of RAM
$0048 ldi r16, low(RAMEND)
$0049 out SPL,r16
$004A sei ; Enable interrupts
$004B <instr> xxx

```

При входе в прерывные бит общего разрешения прерываний сбрасывается и все прерывания запрещаются. В программе-обработчике прерывания можно взвести этот бит, и, таким образом, вновь разрешить прерывания. В этом случае вновь поступившее прерывание останавливает работу обработчика текущего прерывания. Бит общего разрешения прерываний устанавливается при выходе из прерывания. Существует два типа прерываний. Первый тип – триггерный, при этом устанавливается флажок прерывания. Второй тип прерываний не имеет соответствующего флажка, и такое прерывание может быть потеряно, если в это время бит общего разрешения прерываний был сброшен. Для запрещения и разрешения прерываний в системе команд есть соответствующие инструкции – CLI и STI.

Время отклика на любое прерывание занимает четыре периода тактовой частоты. Если прерывание происходит во время выполнения длинной команды, эта команда корректно завершается до начала обработки прерывания. Если прерывание происходит в то время, когда контроллер находился в режиме ожидания, добавляются еще четыре такта для перехода контроллера в активный режим. Возврат из прерывания также занимает четыре периода тактовой частоты.

Далее будут рассмотрены периферийные модули, которые входят в состав микросхемы ATmega128.

### Таймеры-счетчики

ATmega128 имеет на борту 4 таймера-счетчика. Два 8-разрядных счетчика с независимыми входными делителями и режимами сравнения. Следующая пара счетчиков — 16-разрядные, в дополнение к ранее указанным функциям имеют режимы сравнения и захвата. Один 8-разрядный счетчик может использоваться как таймер часов реального времени. Для этого на входе счетчика есть микромощный генератор, к которому непосредственно можно подключить «часовой» кварц. Основная функция этого счетчика — формирование длинных интервалов времени, максимальный период — 8 секунд. На базе таймеров-счетчиков построены широтно-импульсные модуляторы, причем 2 канала ШИМ 8-разрядные, и 6 каналов — с программируемой разрядностью от 2 до 16 бит.

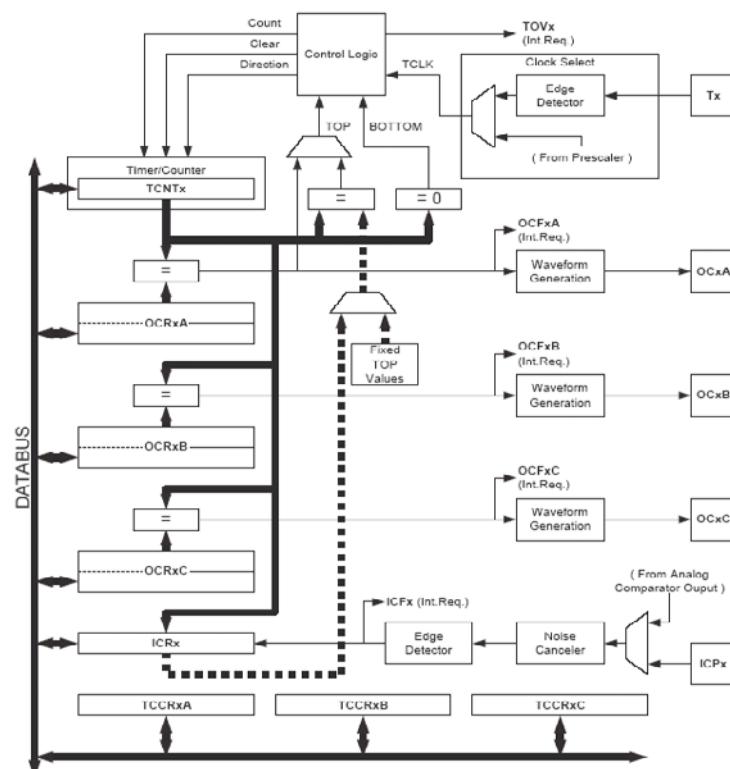


Рис. 7 Структурная схема 16-разрядного таймера-счетчика.

## Модуль АЦП

Многие AVR-микроконтроллеры имеют встроенный аналого-цифровой преобразователь. У микросхем Atmega8535/16/32 из-за небольшого количества выводов пришлось пожертвовать возможностью работы с внешним ОЗУ, однако, ATmega128 не имеет такого ограничения.

В состав микросхемы входит 8-канальный 10-битный АЦП со временем преобразования 65 мкс на канал. Интегральная нелинейность АЦП составляет 0,5 единицы младшего разряда. АЦП может работать в следующих режимах:

- 8-канальный с одиночными входами;
- 7-канальный псевдодифференциальный;
- 2-канальный дифференциальный с программируемым входным усилителем (1x, 10x и 200x).

Также на кристалле размещен источник опорного напряжения 2,56 В. Максимальное напряжение на аналоговых входах не должно превышать уровень аналогового напряжения питания AVCC. Процесс аналого-цифрового преобразования в установившемся режиме занимает 13 тактов генератора.

Структурная схема модуля АЦП представлена на рис. 8.

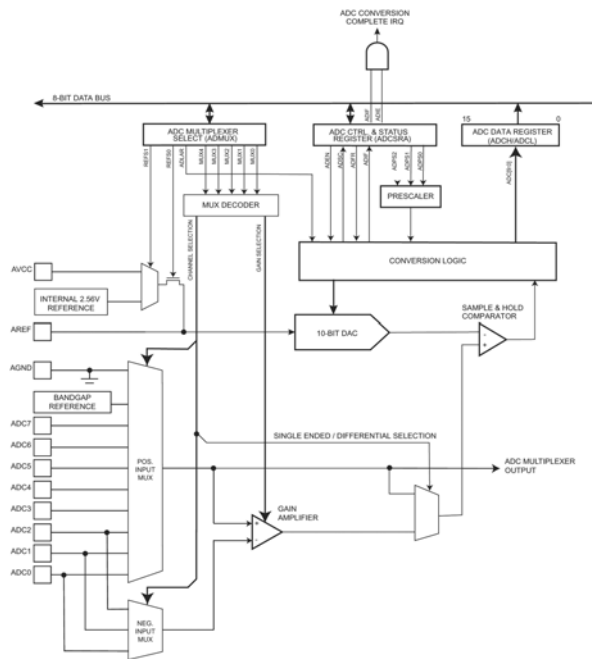


Рис. 8. Схема модуля АЦП

## Последовательные порты

Микроконтроллер ATmega128 щедро оснащен портами ввода-вывода. Стандартно он имеет 7 параллельных портов: 8-битные А, В, С, D, Е, F и 5-битный порт G. Однако, в соответствии с общей тенденцией перехода на последовательные коммуникационные интерфейсы, ATmega128 поддерживает связь с внешним миром через несколько последовательных портов. Естественно, в силу ограниченного количества выводов корпуса микросхемы ATmega128, многие выводы имеют несколько альтернативных режимов. Краткое описание портов представлено в таблице 3.

Таблица 3.

Название порта	Скорость обмена	Число линий связи
UART/USART0	до 2 Мбод	2/3
UART/USART1	до 2 Мбод	2/3
SPI, Master/Slave	до 8 Мбод	3/4
TWI, Multi-master	до 400 кбод	2
JTAG	IEEE std	4

Примечание: порт TWI аппаратно реализует интерфейс I<sup>2</sup>C.

## Характеристики порта USART:

- полнодуплексный обмен;
- асинхронные или синхронные операции;
- двойная скорость обмена в асинхронном режиме;
- работа в многопроцессорной конфигурации;
- тактирование от контроллера или периферийного устройства;
- генератор частоты обмена с высоким разрешением;
- работа со словами 5, 6, 7, 8 или 9 битов данных и 1 либо 2 стоповыми битами;
- аппаратный генератор бита паритета;
- определение ошибки передачи пакета;
- входной фильтр, снижающий влияние шума;
- три независимых прерывания: завершение приема, завершение передачи, опустошение регистра передачи.

Порт UART/USART, как видно из названия, может работать либо в асинхронном, либо в синхронном режиме. В первом случае используются две линии, а во втором — три. Работа порта в асинхронном режиме аналогична работе UART в AVR-микроконтроллерах первого поколения. Отличие – в регистре данных приемника UDR, который стал двухуровневым, таким образом, можно начинать прием следующего пакета до завершения обработки предыдущего. Структурная схема порта USART представлена на рисунке 8.

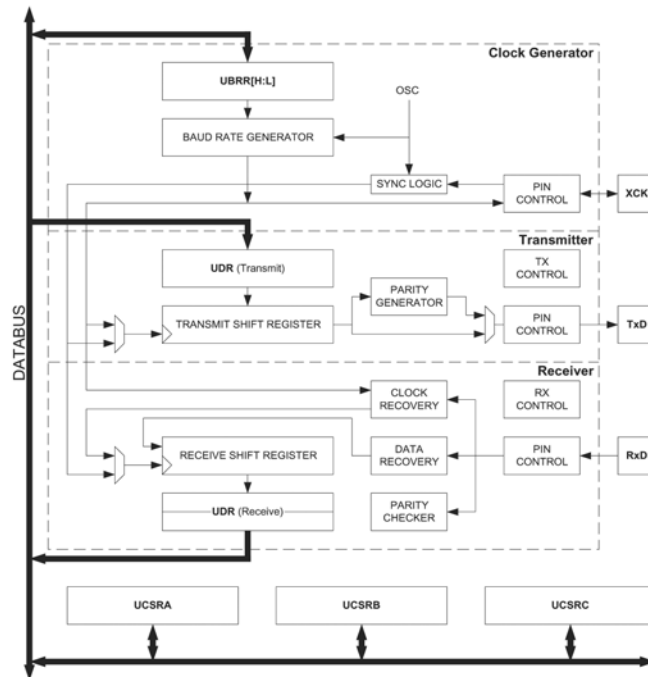


Рис. 8. Схема модуля USART

## Характеристики порта SPI

- полнодуплексный синхронный обмен;
- поддержка режимов Master и Slave;
- начало передачи со старшего или младшего бита;
- семь программируемых скоростей передачи;
- удвоенная скорость передачи (половина тактовой частоты);
- флаг прерывания по окончании передачи;
- обнаружение ошибки записи.

Передача информации по интерфейсу SPI представляет собой обмен данными между двумя закольцованными сдвиговыми регистрами, один из которых размещен в устройстве Master, а второй — в устройстве Slave.

Модуль SPI имеет одиночный буфер на стороне передатчика и двойной буфер на стороне приемника. Это значит, что передаваемый байт не может быть записан в регистр данных SPI до завершения цикла сдвига.

Принятый байт должен быть считан из регистра-приемника до того, как завершится прием следующего байта, иначе произойдет потеря информации. Структурная схема порта SPI представлена на рисунке 9.

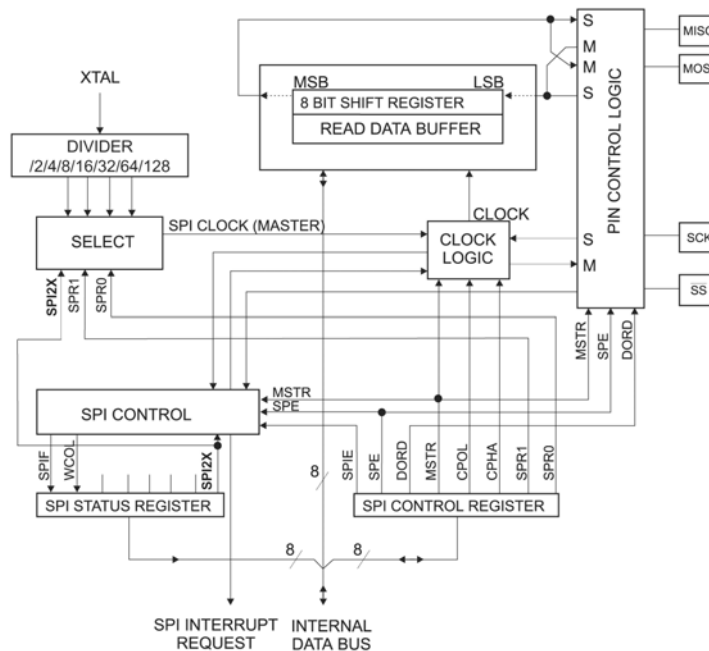


Рис. 9. Схема модуля SPI

### Характеристики порта TWI

- полная совместимость и интерфейсом I<sup>2</sup>C;
- скорость передачи до 400 кГц;
- поддержка режимов Master и Slave;
- 7-битный адрес позволяет использовать до 128 устройств на шине;
- микроконтроллер может функционировать как приемник и как передатчик;
- схема ограничения скорости нарастания фронтов выходных сигналов;
- защита по входам от шумов и кратковременных ложных импульсов;
- переход из «спящего» состояния в активное при распознавании адреса обращения.

АЦП может работать в режиме одиночного запуска или в непрерывном режиме. После окончания преобразования вырабатывается соответствующее прерывание. Для получения максимальной точности измерения аналогового сигнала рекомендуется на время аналого-цифрового преобразования перевести микроконтроллер в спящий режим.

Помимо АЦП на кристалле есть еще аналоговый компаратор. Он сравнивает сигналы на входах AIN1 и AIN2. Есть возможность подать на инвертирующий вход компаратора любой из входных сигналов АЦП.

### Режимы энергосбережения

Режим энергосбережения (режим ожидания, «спящий» режим) позволяет отключить неиспользуемые модули микроконтроллера, и, таким образом, снизить энергопотребление. AVR- микроконтроллеры имеют несколько режимов ожидания, и программист может выбрать оптимальный режим для конкретной задачи.

Для перевода контроллера в любой режим ожидания следует установить биты SM2, SM1 и SM0 в регистре MCUCR и выполнить команду SLEEP. Перечень режимов представлен в таблице 4.

Таблица 4.

SM2	SM1	SM0	Режим
0	0	0	Idle
0	0	1	ADC Noise Reduction
0	1	0	Power-down
0	1	1	Power-save
1	1	0	Standby*
1	1	1	Extended Standby*

\* режимы Standby и Extended Standby доступны при работе с внешним осциллятором.

Если во время нахождения контроллера в режиме ожидания возникает прерывание, контроллер переходит в активный режим, и через 4 такта после окончания периода «пробуждения», необходимые для выполнения процедуры прерывания, продолжает выполнение программы с команды, следующей за командой SLEEP. Содержимое регистров и ОЗУ остается таким же, какое было до «пробуждения».

Рассмотрим коротко каждый из режимов ожидания.

Режим Idle. Останавливается процессор, но продолжают работать модули SPI, USART, TWI, аналоговый компаратор, АЦП, таймеры-счетчики, сторожевой таймер и модуль прерываний.

Режим ADC Noise Reduction. В этом режиме останавливается процессор, модули SPI и USART. Также отключаются цепи тактирования clkI/O, clkCPU и clkFLASH. Режим ADC Noise Reduction используется для получения более точных результатов измерений АЦП.

Режим Power-down. В этом режиме отключается тактовый генератор, но продолжают работать модуль внешних прерываний, интерфейс TWI и сторожевой таймер. Также отключаются все цепи тактирования, остаются работать только асинхронные модули. Для перехода в активный режим требуется дополнительная задержка, необходимая для запуска тактового генератора. Эта задержка устанавливается fuse-битами CKSEL.

Режим Power-save. Этот режим подобен предыдущему за одним исключением. Если таймер-счетчик 0 тактируется асинхронно, он продолжает функционировать после выполнения команды SLEEP. Контроллер может быть «разбужен» по переполнению таймера или по срабатыванию цифрового компаратора, если разрешено соответствующее прерывание.

Режимы Standby и Extended Standby аналогичны режимам Power-down и Power-save соответственно. Однако в этих режимах тактовый генератор не отключается и время перехода в активный режим составляет 6 тактов.

### JTAG-интерфейс

Принципиально новым узлом микроконтроллера ATmega128 является JTAG-интерфейс. Этот интерфейс используется для проведения периферийного тестирования микросхемы в соответствии со стандартом IEEE std 1149.1, а также для подключения внутрисхемного эмулятора ATJTAG ICE. Физически JTAG-интерфейс занимает 4 старших разряда АЦП. Режим отладки с использованием JTAG-эмулятора поддерживается микроконтроллерами ATmega16/162/169/32/64/128. На кристалле микроконтроллера размещен специальный набор отладочных регистров, через которые эмулятор и управляет выполняемой программой. Еще одна функция JTAG-эмулятора — внутрисхемное программирование микроконтроллера. Следует отметить, что при программировании битов защиты работа JTAG-порта запрещается, чтобы исключить возможность использовать этот порт для несанкционированного доступа к содержимому программной памяти микроконтроллера. Применение JTAG-эмулятора позволяет использовать пользовательскую плату как отладочную. Для этого необходимо разместить 10-контактный разъем для подключения JTAG-эмулятора. В качестве отладочного кристалла используется собственно микроконтроллер, на котором запускается отлаживаемая программа.

Следует отметить, что помимо достаточно напряженной официальной части семинара была и неофициальная, вне стен завода фирмы ATMEL. В числе прочих, естественно, была затронута тема развития линии AVR-микроконтроллеров.

Один из авторов статьи задал (можно сказать ретранслировал) Ингару Фредриксену типичный вопрос российских разработчиков – а каков срок жизни у этих новых AVR-микроконтроллеров? Мы совсем недавно привыкли к микросхеме ATmega103, вполне всех устраивающей, а теперь приходится переходить на ATmega128, которая, конечно лучше, однако, требует некоторой модификации исходных кодов (если они доступны) и перекомпиляции проекта. Этот вопрос породил обстоятельный ответ, который вылился в 15-минутный монолог, и его суть в двух словах такова.

Анализ статистики задач, решаемых 8-разрядными микроконтроллерами, показывает, что от хорошего универсального, то есть пригодного к использованию практически в любой задаче своего класса, контроллера требуются два основных качества. Первое – быстродействие не менее 10 MIPS, чтобы не думать о запасе производительности. Второе – большой объем памяти программ, чтобы писать код на языке высокого уровня и не заботиться о размере выходного hex-файла. При наличии этих качеств работу можно сделать быстро, а это очень важно. В свете вышеперечисленных требований AVR-контроллеры второго поколения идеально подходят для применения в любой области, где нужен 8-разрядный контроллер. А наличие идентичного ядра еще более развязывает руки разработчику – очень легко перейти к любой микросхеме этого обширного семейства без существенных временных затрат. Таким образом, можно сказать, что революционные изменения в аппаратной архитектуре контроллеров просто не требуются, то есть AVR-контроллеры теперь набрали «необходимую высоту» и теперь переходят в «горизонтальный полет».

В заключении стоит в двух словах рассказать о новых изделиях. Во-первых, это кристаллы ATmega162 и ATmega169, отличительной особенностью которых является возможность работы при питании 1,8 В и микропотребление в активном режиме. Для справки: у ATmega169 ток потребления в активном режиме при питании 1,8 В на частоте 32 кГц составляет 20 мкА! Кроме этого, ATmega169 имеет контроллер управления сегментным ЖКИ с матрицей 25 сегментов, 4 колонки. К концу года ожидается выпуск кристалла на базе микросхемы ATmega128 с CAN-интерфейсом. А в более отдаленной перспективе прорисовывается микросхема ATmega256...

Свежую информацию по микросхемам, программное обеспечение и примеры исходных текстов программ можно переписать с web-сайта <http://atmel.argussoft.ru>.