
ТЕХНИЧЕСКОЕ ОПИСАНИЕ

НА МИКРОКОНТРОЛЛЕРЫ ФИРМЫ ATMEL

СОДЕРЖАНИЕ

8-разрядные КМОП Flash микроконтроллеры семейства AT89.....	9
Высокопроизводительные 8-разрядные RISC микроконтроллеры семейства AT90S.....	31
8-разрядные микроконтроллеры с Flash памятью емкостью 64/128Кбайт ATmega603, ATmega603L, ATmega103, ATmega103L.....	55

8-разрядные КМОП Flash микроконтроллеры семейства AT89.

Отличительные особенности

- 8-разрядное ЦПУ, оптимизированное для функций управления
- расширенные возможности по-битовой обработки
- встроенная Flash память программ
- встроенное ОЗУ данных
- двунаправленные индивидуально адресуемые линии ввода/вывода
- встроенные 16-разрядные таймеры/счетчики событий
- полный дуплексный UART
- несколько источников прерываний с несколькими уровнями приоритета
- встроенный тактовый генератор
- встроенное ЭСППЗУ (серия AT89S)
- интерфейс последовательной шины SPI (серия AT89S)
- сторожевой таймер (серия AT89S)
- пассивный (idle) и стоповый (power down) режимы
- возможность расширения внешнего ОЗУ и ПЗУ до 64 Кбайт
- режим внутрисхемной эмуляции (ONCE™ - on circuit emulation)

КМОП микроконтроллеры семейства AT89 оснащены Flash программируемым и стираемым ПЗУ, совместимы по системе команд и по выводам со стандартными приборами семейства MCS-51™. Микроконтроллеры содержат Flash ПЗУ, емкостью от 1 Кбайта до 8 Кбайт, ОЗУ, емкостью от 64 байтов до 256 байтов, большое количество программируемых линий ввода/вывода, 16-разрядные таймеры/счетчики событий, полнодуплексный последовательный порт (UART), набор векторных прерываний с несколькими уровнями приоритета, встроенные генератор и схему формирования тактовой последовательности. Программирование микроконтроллеров семейства AT89 возможно, в зависимости от типа, внутрисистемно, с использованием при программировании напряжения питания системы, и с использованием программаторов. Содержимое Flash памяти программ может быть защищено от несанкционированной записи/считывания. У ряда микроконтроллеров имеется возможность очистки Flash памяти за одну операцию, возможность считывания встроенного кода идентификации.

В активном режиме микроконтроллеры на частоте 12 МГц потребляют порядка 25 мА и в пассивном режиме, при котором остановлено ЦПУ но система прерываний, ОЗУ, таймеры/счетчики событий и последовательный порт остаются активными, потребление снижается до 15% от потребления в активном режиме. В стоповом режиме потребление не превышает 100 мкА. Микроконтроллеры семейства AT89 ориентированы на использование в качестве встроенных управляющих контроллеров в промышленном (-40°С...85°С) и коммерческом (0°С...70°С) диапазонах температур, имеются исполнения микроконтроллеров AT89C51 и AT89C52, соответствующие требованиям, предъявляемым к приборам используемым в автомобильном (-40°С...125°С), Military и Military/883C (-55°С...125°С) диапазонах температур.

Основные характеристики микроконтроллеров семейства AT89

Тип прибора	Совместимость	Напряж. питания (Mil), VCC, В	Объем Flash ПЗУ (ЭСПП ЗУ), байт	Объем ОЗУ, байт	Группы по тактовой частоте, МГц	Потребление макс. в активн. (пасс.) режиме, f=12 МГц, VCC=6В, мА	Температурный диапазон	Тип корпуса (Military и Military/883C версии)
AT89C1051	80C31	2,7...6	1 К	64	12, 24	15 (5)	Пром., Комм.	20PDIP, 20SOIC
AT89C2051	80C31	2,7...6	2 К	128	12, 24	15 (5)	Пром., Комм.	20PDIP, 20SOIC
AT89C51	80C31	5±20% (5±10%)	4 К	128	12, 16, 20, 24	20 (5)	Пром., Комм. Авто., (Mil)*	44TQFP, 44PLCC, 44PQFP, 40PDIP (40Cerdip, 44PLCC)
AT89LV51	80C31	2,7...6	4 К	128	12	20 (5)	Комм.	44TQFP, 44PLCC, 44PQFP, 40PDIP
AT89C52	80C32	5±20% (5±10%)	8 К	256	12, 16, 20, 24	25 (6,5)	Пром., Комм. Авто., (Mil)*	44TQFP, 44PLCC, 44PQFP, 40PDIP (40Cerdip, 44PLCC)
AT89LV52	80C32	2,7...6	8 К	256	12	25 (6,5)	Комм.	44TQFP, 44PLCC, 44PQFP, 40PDIP
AT89S8252	80C32	2,7±10% 5±20%	8 К, (2 К)	256	12(2,7В), 24(5 В)		Пром., Комм.	44TQFP, 44PLCC, 44PQFP, 40PDIP

Примечание:* Military и Military/883C версии возможны только с частотой 12 МГц.

8-разрядный КМОП микроконтроллер с Flash ПЗУ

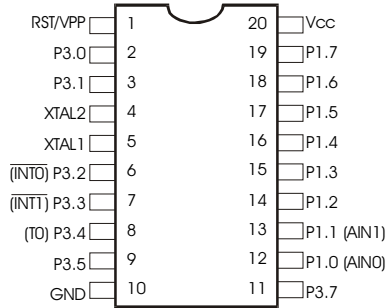
Отличительные особенности

- Совместимость с приборами семейства MCS-51™
- Емкость перепрограммируемой Flash памяти: 1 Кбайт, 1000 циклов стирание/запись.
- Диапазон рабочих напряжений от 2,7 В до 6 В
- Полностью статический прибор - диапазон рабочих частот от 0 Гц до 24 МГц
- Двухуровневая блокировка памяти программ
- ОЗУ емкостью 64 байта
- 15 программируемых линий ввода/вывода
- 16-разрядный таймер/счетчик событий
- Три источника сигналов прерывания
- Выходы прямого управления СИД
- Встроенный аналоговый компаратор
- Пассивный (idle) и стоповый (power down) режимы
- Промышленный (-40°C...85°C) и коммерческий (0°C...70°C) диапазоны температур
- 20-выводные корпуса PDIP и SOIC

КМОП микроконтроллер AT89C1051, оснащенный Flash программируемым и стираемым ПЗУ, совместим по системе команд и по выводам со стандартными приборами семейства MCS-51™. Микроконтроллер содержит 1 Кбайт Flash ПЗУ, 64 байта ОЗУ, 15 линий ввода/вывода, один 16-разрядный таймер/счетчик событий, три векторных двухуровневых прерывания, встроенный прецизионный аналоговый компаратор, встроенные генератор и схему формирования тактовой последовательности. Программирование Flash памяти программ ведется с использованием напряжения 12 В, ее содержимое может быть защищено от несанкционированных записи/считывания. Имеется возможность очистки Flash памяти за одну операцию, возможность считывания встроенного кода идентификации.

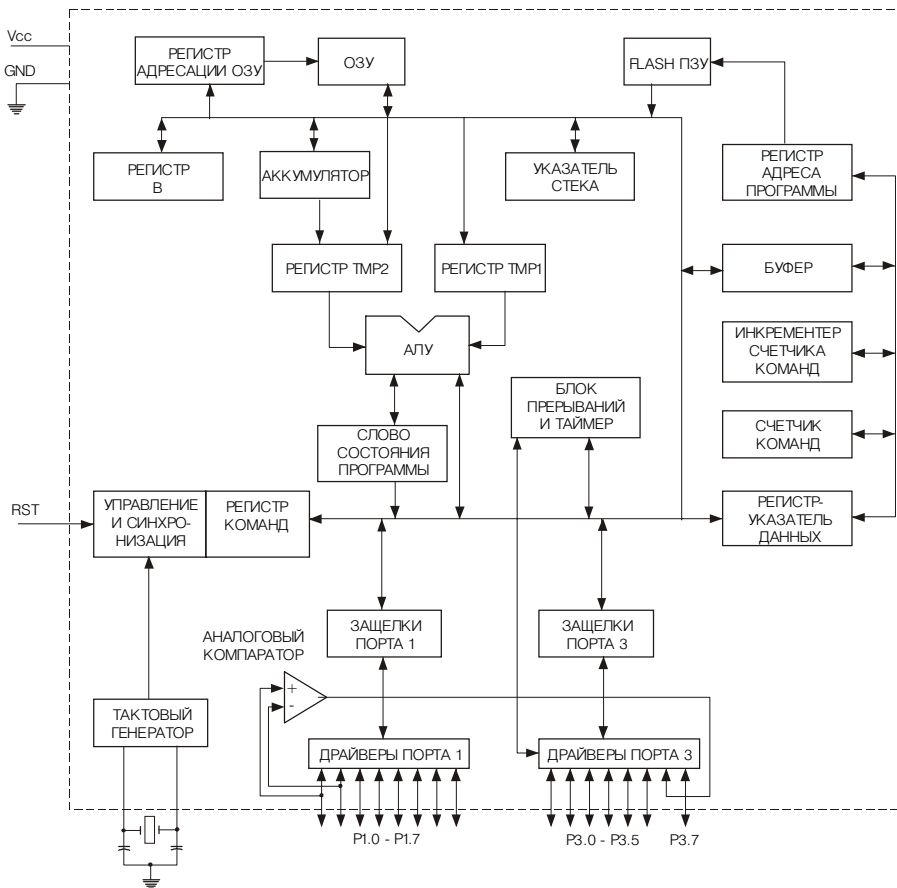
Потребление в активном режиме на частоте 12 МГц не превышает 15 мА и 5,5 мА при напряжении питания 6 В и 3 В, соответственно. При тех же условиях в пассивном режиме, при котором остановлено ЦПУ но система прерываний, ОЗУ, таймер/счетчик событий и последовательный порт остаются активными, потребление не превышает 5 мА и 1 мА. В стоповом режиме потребление не превышает 100 мкА и 20 мкА при напряжении питания 6 В и 3 В, соответственно.

Микроконтроллер AT89C1051 ориентирован на использование в качестве встроенного управляющего контроллера.



Корпус PDIP/SOIC

Блок-схема микроконтроллера AT89C1051.



8-разрядный КМОП микроконтроллер с Flash ПЗУ

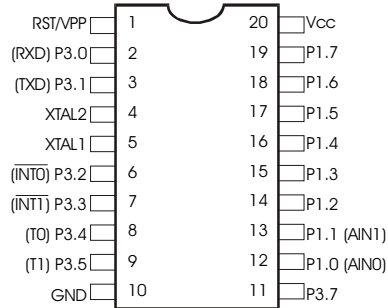
Отличительные особенности

- Совместимость с приборами семейства MCS-51™
- Емкость перепрограммируемой Flash памяти: 2 Кбайт, 1000 циклов стирание/запись
- Диапазон рабочих напряжений от 2,7 В до 6 В
- Полностью статический прибор - диапазон рабочих частот от 0 Гц до 24 МГц
- Двухуровневая блокировка памяти программ
- ОЗУ емкостью 128 байтов
- 15 программируемых линий ввода/вывода
- Два 16-разрядных таймера/счетчика событий
- Шесть источников сигналов прерывания
- Программируемый последовательный канал UART
- Выходы прямого управления СИД
- Встроенный аналоговый компаратор
- Пассивный (idle) и стоповый (power down) режимы
- Промышленный (-40°C...85°C) и коммерческий (0°C...70°C) диапазоны температур
- 20-выводные корпуса PDIP и SOIC

КМОП микроконтроллер AT89C2051, оснащенный Flash программируемым и стираемым ПЗУ, совместим по системе команд и по выводам со стандартными приборами семейства MCS-51™. Микроконтроллер содержит 2 Кбайта Flash ПЗУ, 128 байтов ОЗУ, 15 линий ввода/вывода, два 16-разрядных таймера/счетчика событий, полнодуплексный последовательный порт (UART), пять векторных двухуровневых прерываний, встроенный прецизионный аналоговый компаратор, встроенные генератор и схему формирования тактовой последовательности. Программирование Flash памяти программ ведется с использованием напряжения 12 В, ее содержимое может быть защищено от несанкционированных записи/считывания. Имеется возможность очистки Flash памяти за одну операцию, возможность считывания встроенного кода идентификации.

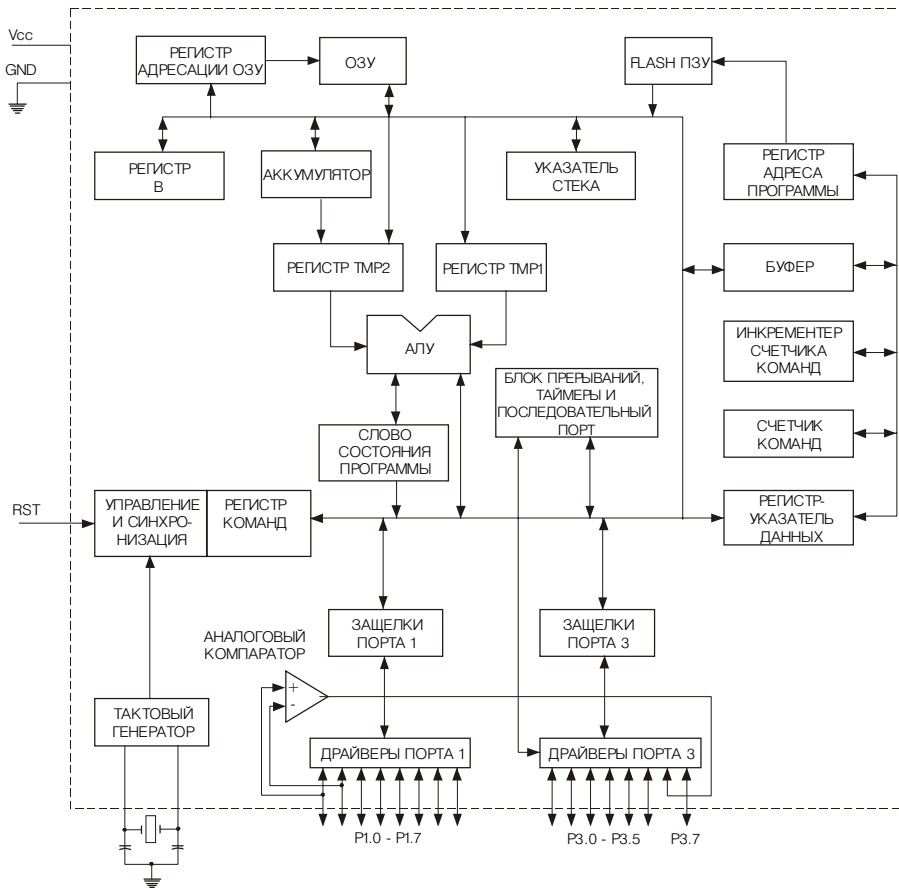
Потребление в активном режиме на частоте 12 МГц не превышает 15 мА и 5,5 мА при напряжении питания 6 В и 3 В, соответственно. При тех же условиях в пассивном режиме, при котором остановлено ЦПУ но система прерываний, ОЗУ, таймер/счетчик событий и последовательный порт остаются активными, потребление не превышает 5 мА и 1 мА. В стоповом режиме потребление не превышает 100 мкА и 20 мкА при напряжении питания 6 В и 3 В, соответственно.

Микроконтроллер AT89C2051 ориентирован на использование в качестве встроенного управляющего контроллера.



Корпус PDIP/SOIC

Блок-схема микроконтроллера AT89C2051.



8-разрядный КМОП микроконтроллер с Flash ПЗУ

Отличительные особенности

- Совместимость с приборами семейства MCS-51™
- Емкость перепрограммируемой Flash памяти: 4Кбайт, возможность внутрисистемного перепрограммирования, 1000 циклов стирание/запись
- Напряжение питания $5\pm 20\%$ В ($5\pm 10\%$ В для приборов Military и Military/883C исполнения)
- Полностью статический прибор - диапазон рабочих частот от 0 Гц до 24 МГц
- Группы по частотам: 12 МГц, 16 МГц, 20 МГц и 24 МГц (Military и Military/883C исполнения только в группе 12 МГц, автомобильное исполнение только в группах 12 МГц и 16 МГц)
- Трехуровневая блокировка памяти программ
- ОЗУ емкостью 128 байтов
- 32 программируемых линий ввода/вывода
- Два 16-разрядных таймера/счетчика событий
- Шесть источников сигналов прерывания
- Программируемый последовательный канал UART
- Пассивный (idle) и стоповый (power down) режимы
- Промышленный (-40°C...85°C), коммерческий (0°C...70°C), автомобильный (-40°C...125°C), Military и Military/883C (-55°C...125°C) диапазоны температур.
- 40-выводной корпус PDIP, 44-выводные корпуса TQFP и PQFP и 44-выводной носитель кристалла PLCC для промышленного коммерческого и автомобильного диапазонов температур.
- 40-выводной корпус Cerdip и 44-выводной носитель кристалла LCC для приборов Military и Military/883C исполнения.

КМОП микроконтроллер AT89C51, оснащенный Flash программируемым и стираемым ПЗУ, совместим по системе команд и по выводам со стандартными приборами семейства MCS-51™. Микроконтроллер содержит 4 Кбайта Flash ПЗУ, 128 байтов ОЗУ, 32 программируемых линии ввода/вывода, два 16-разрядных таймера/счетчика событий, полнодуплексный последовательный порт (UART), пять векторных двухуровневых прерывания, встроенные генератор и схему формирования тактовой последовательности.

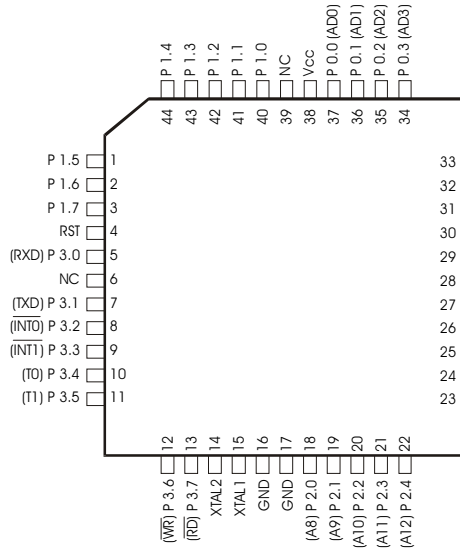
Существуют два варианта микроконтроллеров AT89C51: с возможностью внутрисистемного программирования с использованием при программировании напряжения 5 В, и программирование с использованием напряжения 12 В, применяемого в большинстве программаторов. Содержимое Flash памяти программ может быть защищено от несанкционированной записи/считывания. Имеется возможность очистки Flash памяти за одну операцию, возможность считывания встроенного кода идентификации.

Потребление в активном режиме на частоте 12 МГц не превышает 20 мА и в пассивном режиме, при котором остановлено ЦПУ но система прерываний, ОЗУ, таймеры/счетчики событий и последовательный порт остаются активными, потребление не

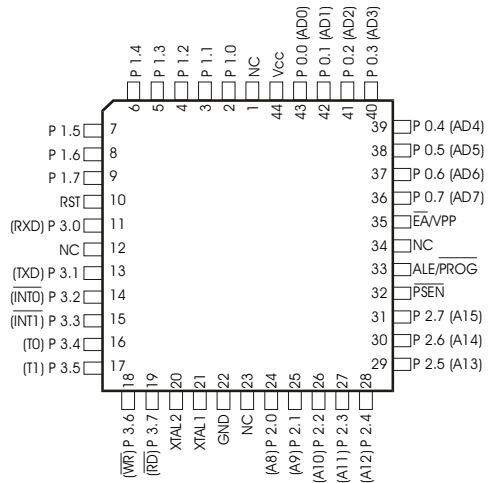
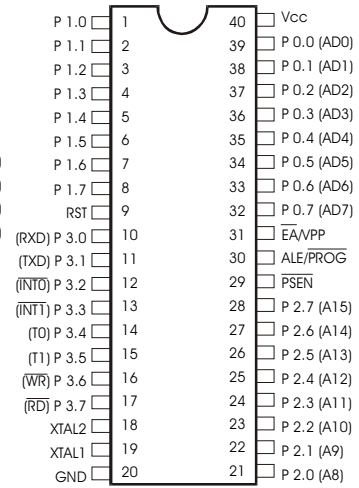
превышает 5 мА. В стоповом режиме потребление не превышает 100 мкА и 20 мкА при напряжении питания 6 В и 3 В, соответственно.

Микроконтроллер AT89C51 ориентирован на использование в качестве встроенного управляющего контроллера.

Корпус PQFP/TQFP

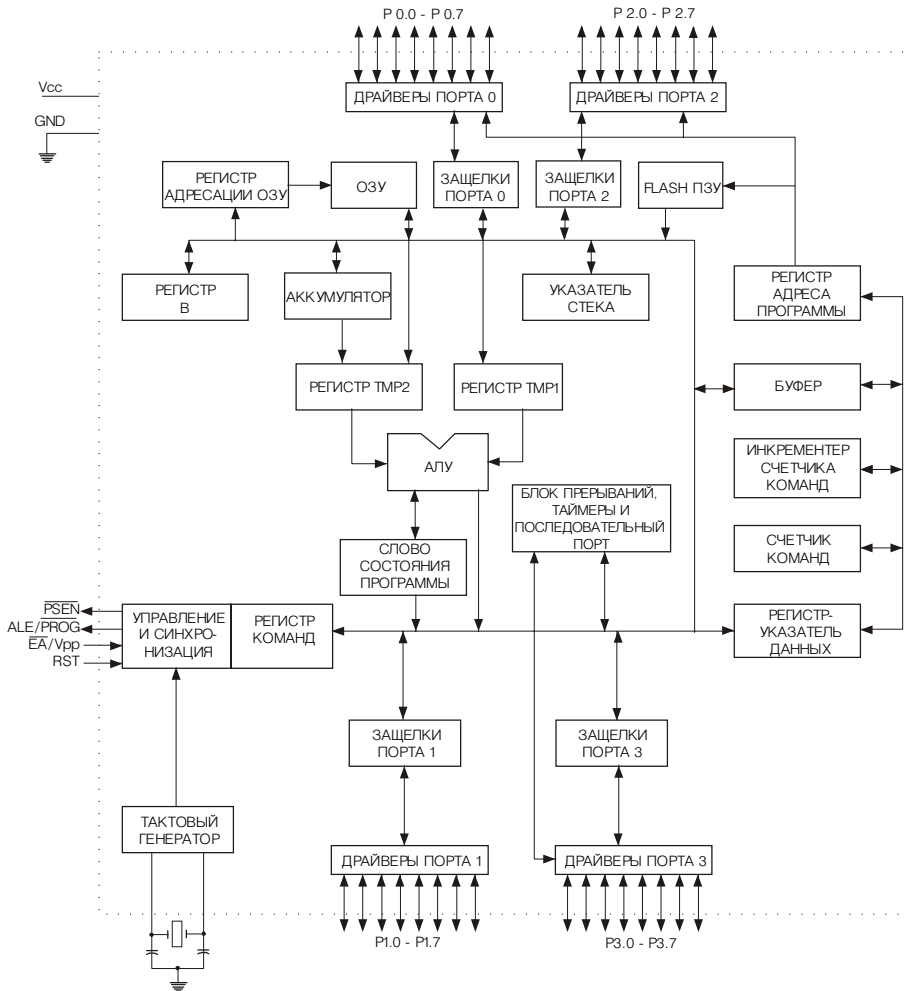


Корпус PDIP/Cerdip



Корпус PLCC/LCC

Блок-схема микроконтроллера AT89C51.



8-разрядный КМОП микроконтроллер с Flash ПЗУ

Отличительные особенности

- Совместимость с приборами семейства MCS-51™
- Емкость перепрограммируемой Flash памяти: 4Кбайт, возможность внутрисистемного перепрограммирования, 1000 циклов стирание/запись
- Диапазон рабочих напряжений от 2,7 В до 6 В
- Полностью статический прибор - диапазон рабочих частот от 0 Гц до 12 МГц.
- Трехуровневая блокировка памяти программ
- ОЗУ емкостью 128 байтов
- 32 программируемых линии ввода/вывода
- Два 16-разрядных таймера/счетчика событий
- Шесть источников сигналов прерывания
- Программируемый последовательный канал UART
- Пассивный (idle) и стоповый (power down) режимы
- Коммерческий (0°C...70°C) диапазон температур.
- 40-выводной корпус PDIP, 44-выводные корпуса TQFP и PQFP и 44-выводной носитель кристалла PLCC

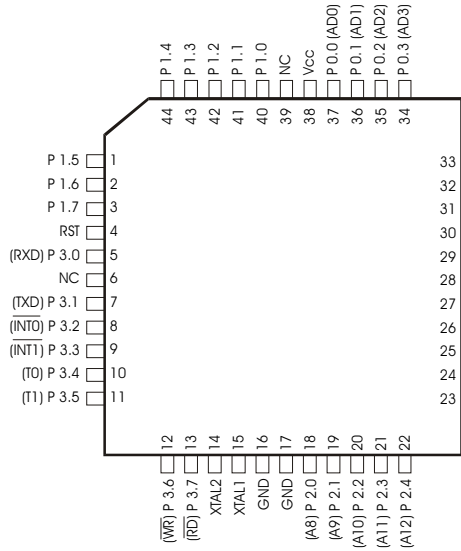
КМОП микроконтроллер AT89LV51, оснащенный Flash программируемым и стираемым ПЗУ, совместим по системе команд и по выводам со стандартными приборами семейства MCS-51™. Микроконтроллер содержит 4 Кбайта Flash ПЗУ, 128 байтов ОЗУ, 32 программируемых линии ввода/вывода, два 16-разрядных таймера/счетчика событий, полнодуплексный последовательный порт (UART), пять векторных двухуровневых прерываний, встроенные генератор и схему формирования тактовой последовательности.

Существуют два варианта микроконтроллеров AT89LV51: с возможностью внутрисистемного программирования с использованием при программировании напряжения 5 В, и программирование с использованием напряжения 12 В, применяемого в большинстве программаторов. Содержимое Flash памяти программ может быть защищено от несанкционированной записи/считывания. Имеется возможность очистки Flash памяти за одну операцию, возможность считывания встроенного кода идентификации.

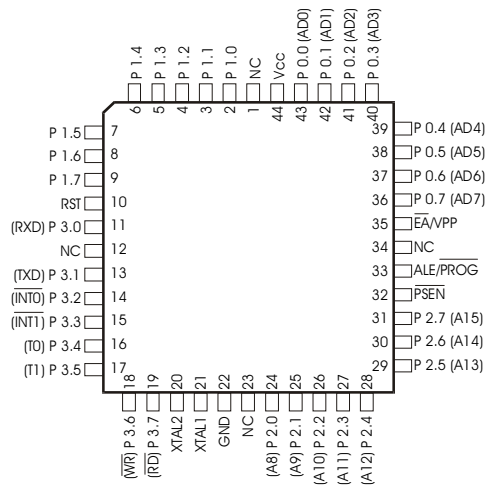
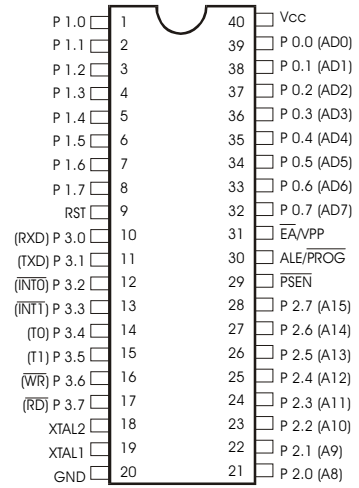
Потребление в активном режиме на частоте 12 МГц не превышает 20 мА и 5,5 мА при напряжении питания 6 В и 3 В, соответственно. При тех же условиях в пассивном режиме, при котором остановлено ЦПУ но система прерываний, ОЗУ, таймер/счетчик событий и последовательный порт остаются активными, потребление не превышает 5 мА и 1 мА. В стоповом режиме потребление не превышает 100 мкА и 20 мкА при напряжении питания 6 В и 3 В, соответственно.

Микроконтроллер AT89LV2051 ориентирован на использование в качестве встроенного управляющего контроллера.

Kopnyc PQFP/TQFP

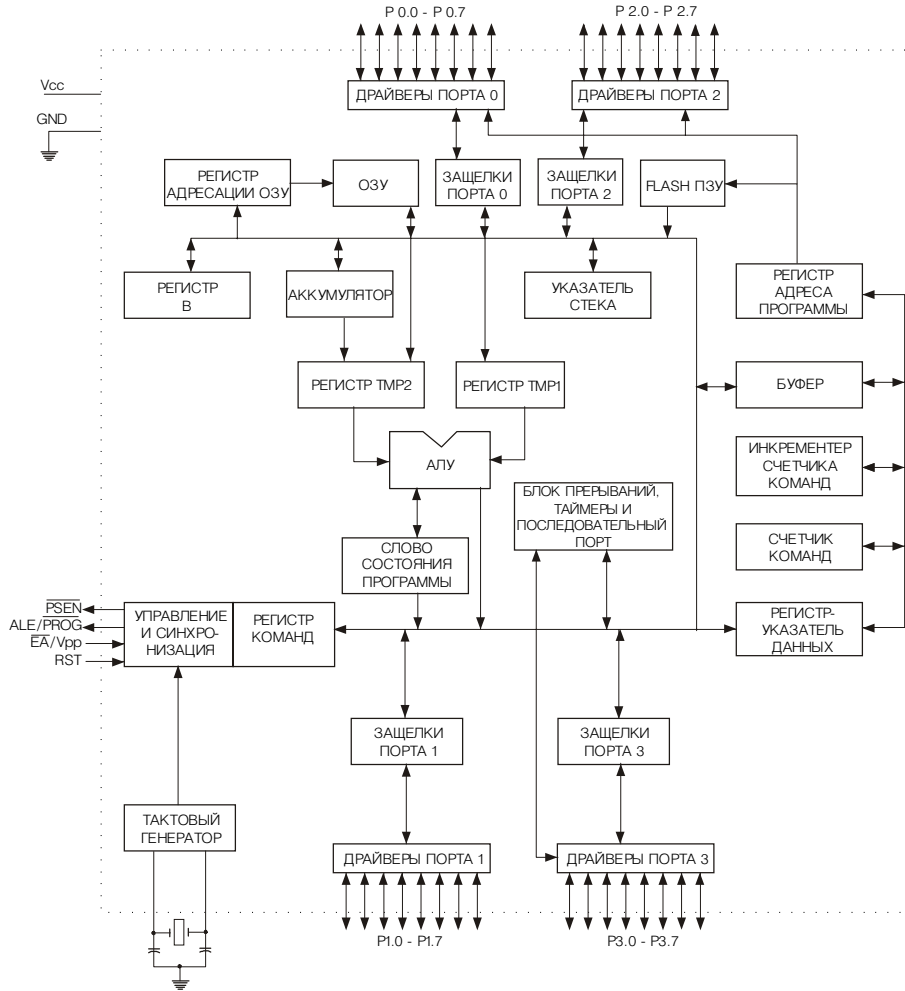


Kopnyc PDIP/Cerdip



Kopnyc PLCC

Блок-схема микроконтроллера AT89LV51.



8-разрядный КМОП микроконтроллер с Flash ПЗУ

Отличительные особенности

- Совместимость с приборами семейства MCS-51™
- Емкость перепрограммируемой Flash памяти: 4Кбайт, возможность внутрисистемного перепрограммирования, 1000 циклов стирание/запись
- Напряжение питания $5\pm 20\%$ В ($5\pm 10\%$ В для приборов Military и Military/883C исполнения)
- Полностью статический прибор - диапазон рабочих частот от 0 Гц до 24 МГц
- Группы по частотам: 12 МГц, 16 МГц, 20 МГц и 24 МГц (Military и Military/883C исполнения только в группе 12 МГц, автомобильное исполнение только в группах 12 МГц и 16 МГц)
- Трехуровневая блокировка памяти программ
- ОЗУ емкостью 256 байтов
- 32 программируемых линии ввода/вывода
- Три 16-разрядных таймера/счетчика событий
- Восемь источников сигналов прерывания
- Программируемый последовательный канал UART
- Пассивный (idle) и стоповый (power down) режимы
- Промышленный (-40°C...85°C), коммерческий (0°C...70°C), автомобильный (-40°C...125°C), Military и Military/883C (-55°C...125°C) диапазоны температур.
- 40-выводной корпус PDIP, 44-выводные корпуса TQFP и PQFP и 44-выводной носитель кристалла PLCC для промышленного коммерческого и автомобильного диапазонов температур.
- 40-выводной корпус Cerdip и 44-выводной носитель кристалла LCC для приборов Military и Military/883C исполнения.

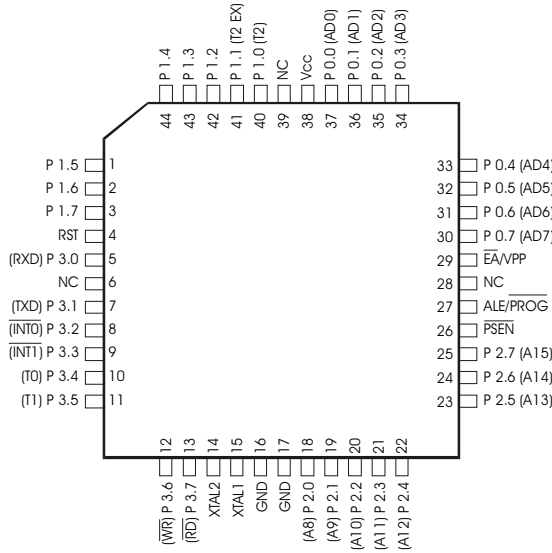
КМОП микроконтроллер AT89C52, оснащенный Flash программируемым и стираемым ПЗУ, совместим по системе команд и по выводам со стандартными приборами семейств 80C51 и 80C52. Микроконтроллер содержит 8 Кбайт Flash ПЗУ, 256 байтов ОЗУ, 32 программируемых линии ввода/вывода, три 16-разрядных таймера/счетчика событий, полнодуплексный последовательный порт (UART), шесть векторных двухуровневых прерываний, встроенные генератор и схему формирования тактовой последовательности.

Существуют два варианта микроконтроллеров AT89C52: с возможностью внутрисистемного программирования с использованием при программировании напряжения 5 В, и программирование с использованием напряжения 12 В, применяемого в большинстве программаторов. Содержимое Flash памяти программ может быть защищено от несанкционированной записи/считывания. Имеется возможность очистки Flash памяти за одну операцию, возможность считывания встроенного кода идентификации.

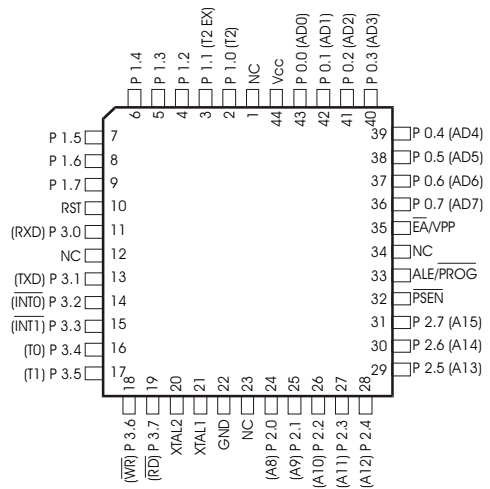
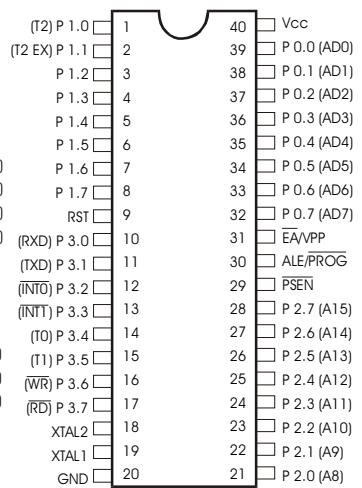
Потребление в активном режиме на частоте 12 МГц не превышает 25 мА и в пассивном режиме, при котором остановлено ЦПУ но система прерываний, ОЗУ, таймеры/счетчики событий и последовательный порт остаются активными, потребление не

превышает 6,5 мА. В стоповом режиме потребление не превышает 100 мкА и 40 мкА при напряжении питания 6 В и 3 В, соответственно. Микроконтроллер AT89C52 ориентирован на использование в качестве встроенного управляющего контроллера.

Корпус PQFP/TQFP

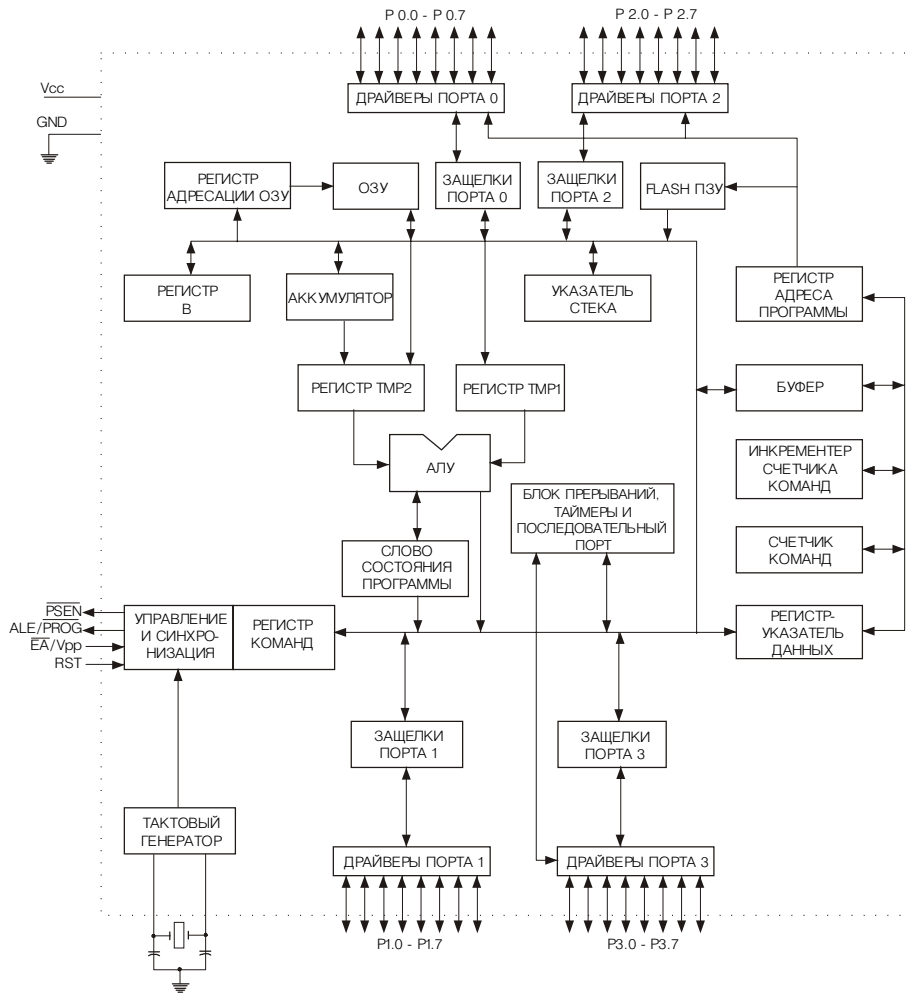


Корпус PDIP/Cerdip



Корпус PLCC/LCC

Блок-схема микроконтроллера AT89C52.



8-разрядный КМОП микроконтроллер с Flash ПЗУ

Отличительные особенности

- Совместимость с приборами семейства MCS-51™.
- Емкость перепрограммируемой Flash памяти: 8Кбайт, возможность внутрисистемного перепрограммирования, 1000 циклов стирание/запись
- Диапазон рабочих напряжений от 2,7 В до 6 В
- Полностью статический прибор - диапазон рабочих частот от 0 Гц до 12 МГц.
- Трехуровневая блокировка памяти программ
- ОЗУ емкостью 256 байтов
- 32 программируемые линии ввода/вывода
- Три 16-разрядных таймера/счетчика событий
- Восемь источников сигналов прерывания
- Программируемый последовательный канал UART
- Пассивный (idle) и стоповый (power down) режимы
- Коммерческий (0°C...70°C) диапазон температур.
- 40-выводной корпус PDIP, 44-выводные корпуса TQFP и PQFP и 44-выводной носитель кристалла PLCC

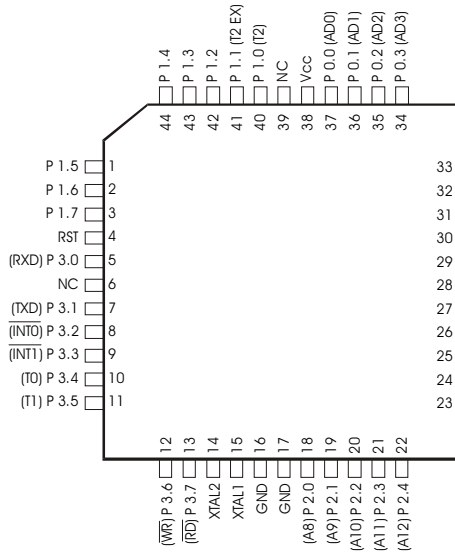
КМОП микроконтроллер AT89LV52, оснащенный Flash программируемым и стираемым ПЗУ, совместим по системе команд и по выводам со стандартными приборами семейств 80C51 и 80C52. Микроконтроллер содержит 8 Кбайт Flash ПЗУ, 256 байт ОЗУ, 32 программируемых линии ввода/вывода, три 16-разрядных таймера/счетчика событий, полнодуплексный последовательный порт (UART), шесть векторных двухуровневых прерываний, встроенные генератор и схему формирования тактовой последовательности.

Существуют два варианта микроконтроллеров AT89LV52: с возможностью внутрисистемного программирования с использованием при программировании напряжения 5 В, и программирование с использованием напряжения 12 В, применяемого в большинстве программаторов. Содержимое Flash памяти программ может быть защищено от несанкционированной записи/считывания. Имеется возможность очистки Flash памяти за одну операцию, возможность считывания встроенного кода идентификации.

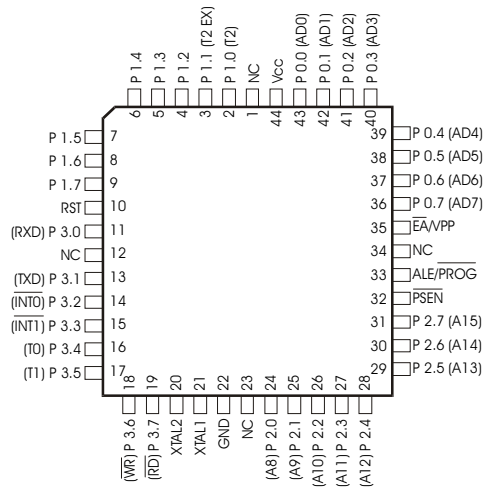
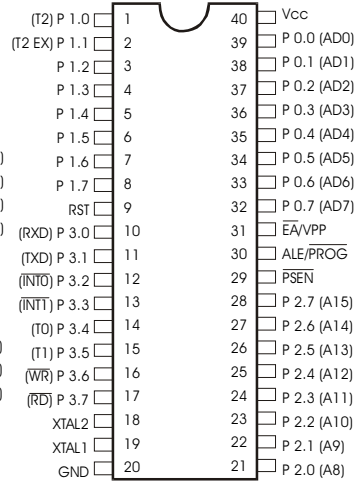
Потребление в активном режиме на частоте 12 МГц не превышает 25 мА и в пассивном режиме, при котором остановлено ЦПУ но система прерываний, ОЗУ, таймеры/счетчики событий и последовательный порт остаются активными, потребление не превышает 6,5 мА. В стоповом режиме потребление не превышает 100 мкА и 40 мкА при напряжении питания 6 В и 3 В, соответственно.

Микроконтроллер AT89LV52 ориентирован на использование в качестве встроенного управляющего контроллера.

Kopnyc PQFP/TQFP

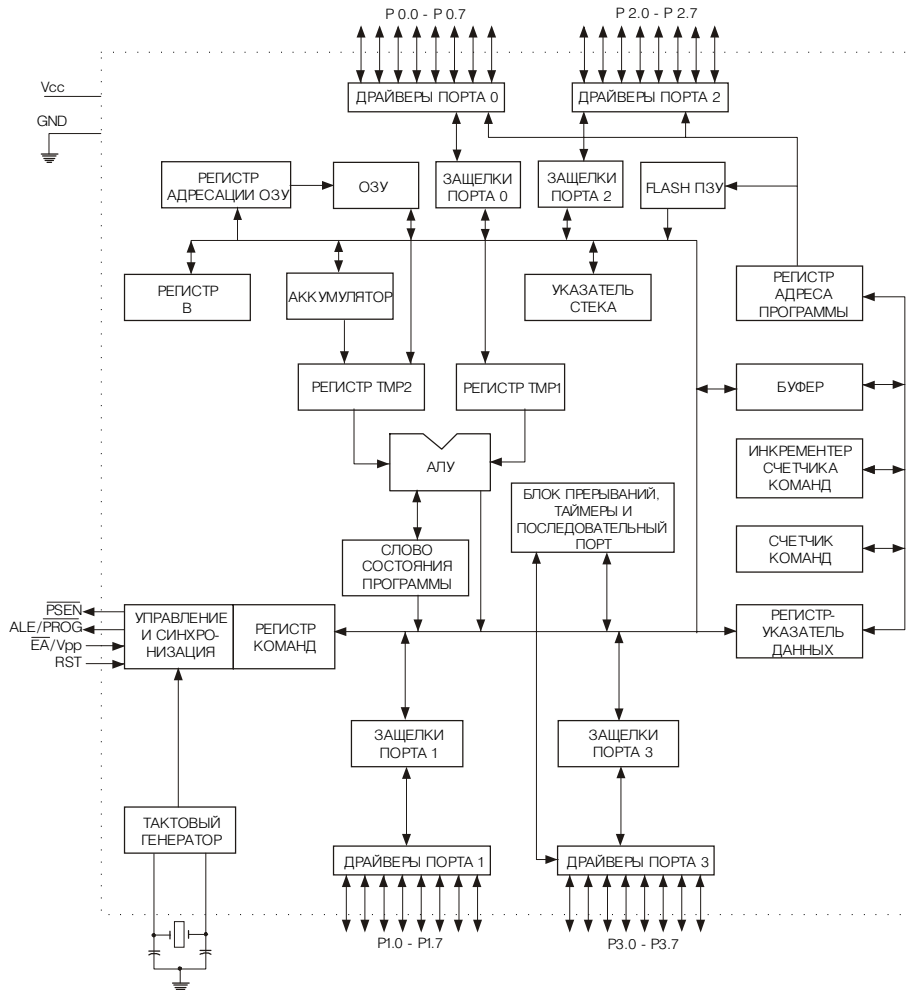


Kopnyc PDIP/Cerdip



Kopnyc PLCC/LCC

Блок-схема микроконтроллера AT89LV52.



8-разрядный КМОП микроконтроллер с Flash ПЗУ

Отличительные особенности

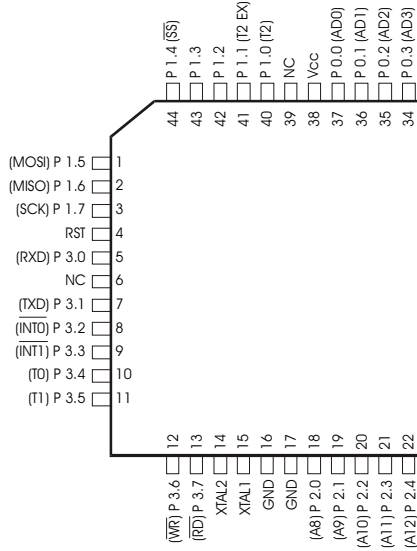
- Совместимость с приборами семейства MCS-51™
- Емкость перепрограммируемой Flash памяти: 8Кбайт
- Возможность внутрисистемного перепрограммирования и загрузки программы через последовательный SPI интерфейс, 1000 циклов стирание/запись
- Встроенное ЭСППЗУ емкостью 2 Кбайта, 100000 циклов стирание/запись
- Версии приборов с напряжением питания $5\pm 20\%$ В и $2,7\pm 10\%$ В
- Полностью статический прибор - диапазон рабочих частот от 0 Гц до 24 МГц
- Рабочие частоты: 12 МГц при напряжении питания $2,7(10\%$ В и 24 МГц при напряжении питания $5\pm 20\%$ В
- Трехуровневая блокировка памяти программ
- ОЗУ емкостью 256 байтов
- Удвоенный указатель данных
- 32 программируемых линии ввода/вывода
- Три 16-разрядных таймера/счетчика событий
- Девять источников сигналов прерывания
- Программируемый последовательный канал UART
- Последовательный SPI интерфейс
- Программируемый сторожевой таймер
- Выход из стопового режима по прерыванию
- Пассивный (idle) и стоповый (power down) режимы
- Промышленный ($-40^{\circ}\text{C}...85^{\circ}\text{C}$) и коммерческий ($0^{\circ}\text{C}...70^{\circ}\text{C}$) диапазоны температур.
- 40-выводной корпус PDIP, 44-выводные корпуса TQFP и PQFP и 44-выводной носитель кристалла PLCC

КМОП микроконтроллер AT89S8252, оснащенный Flash программируемым и стираемым ПЗУ, которое может быть загружено непосредственно в системе через последовательный SPI интерфейс, совместим по системе команд и по выводам со стандартными приборами семейства 80C51. Микроконтроллер содержит 8 Кбайт Flash ПЗУ, 2 Кбайта ЭСППЗУ, 256 байтов ОЗУ, удвоенный указатель данных, 32 программируемых линии ввода/вывода, три 16-разрядных таймера/счетчика событий, полнодуплексный последовательный порт (UART), программируемый сторожевой таймер, семь векторных двухуровневых прерываний, встроенные генератор и схему формирования тактовой последовательности.

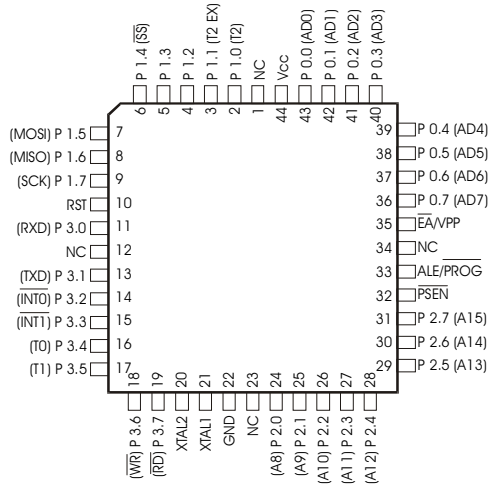
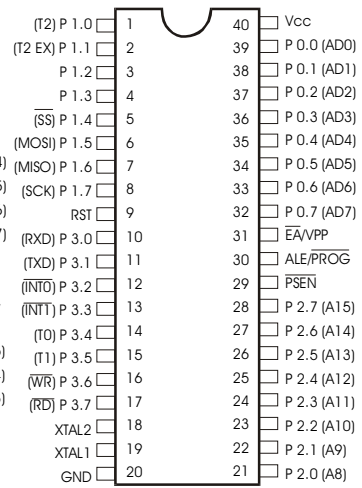
Flash ПЗУ может загружаться по одному байту через последовательный SPI интерфейс. Содержимое Flash памяти программ может быть защищено от несанкционированной записи/считывания.

Микроконтроллер AT89S8252 ориентирован на использование в качестве встроенного управляющего контроллера.

Kopnyc PQFP/TQFP

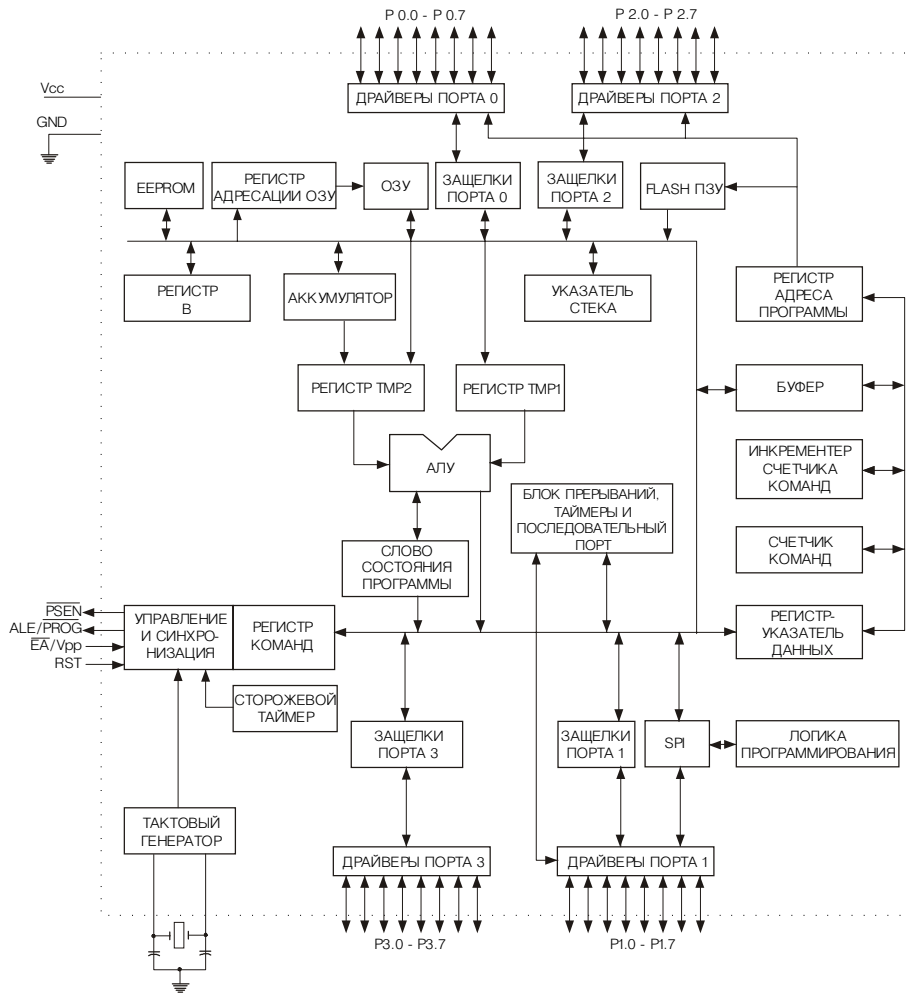


Kopnyc PDIP/Cerdip



Kopnyc PLCC/LCC

Блок-схема микроконтроллера AT89S8252.



Высокопроизводительные 8-разрядные RISC микроконтроллеры семейства AT90S

Отличительные особенности

- Производительность, приближающаяся к 1 MIPS/МГц
- Усовершенствованная AVR RISC архитектура
- Раздельные шины памяти команд и данных, 32 регистра общего назначения
- Flash ПЗУ программ, с возможностью внутрисистемного перепрограммирования и загрузки через SPI последовательный канал, 1000 циклов стирание/запись
- ЭСППЗУ данных, с возможностью внутрисистемной загрузки через SPI последовательный канал, 100000 циклов стирание/запись
- Блокировка режима программирования
- Встроенные аналоговый компаратор, сторожевой таймер, порты SPI и UART, таймеры/счетчики
- Полностью статические приборы - работают при тактовой частоте от 0 Гц до 20 МГц
- Диапазон напряжений питания от 2,7 В до 6,0 В
- Режимы энергосбережения: пассивный (idle) и стоповый (power down)

КМОП микроконтроллеры семейства AT90S выполнены по AVR™ RISC архитектуре с отдельной памятью программ и данных и отдельными шинами для памяти программ и данных (Гарвардская архитектура). AVR ядро объединяет мощную систему команд с 32 регистрами общего назначения и конвейером (в одном цикле одна команда выполняется а другая выбирается) выборки из памяти программ. Все 32 регистра напрямую связаны с АЛУ, что позволяет выполнять обращение к двум независимым регистрам и возвращать результат одной командой, выполняемой в одном тактовом цикле. Шесть регистров могут использоваться как три 16-разрядных указателя адреса данных (кроме прибора AT90S1200). Выполняя команды за один тактовый цикл, прибор обеспечивает производительность, приближающуюся к 1 MIPS на МГц, что на порядок больше, чем у CISC микроконтроллеров. Архитектура эффективно поддерживает как языки высокого уровня, так и программы, написанные на экстремально плотных языках ассемблера.

Микроконтроллеры семейства оснащены встроенной загрузаемой Flash памятью программ, обеспечивающей внутрисистемное перепрограммирование с использованием интерфейса SPI или программирование внешними стандартными программаторами энергонезависимой памяти. Поскольку все команды 16-разрядного или 32-разрядного формата, то Flash память программ имеет 16-разрядную организацию. В качестве памяти данных в микроконтроллерах семейства используется комбинация СОЗУ и ЭСППЗУ (у прибора AT90S1200 только ЭСППЗУ) в которых размещается также и пространство памяти ввода/вывода. Стек, предназначенный для хранения адресов возврата из подпрограмм и прерываний располагается в памяти данных (у прибора AT90S1200 стек выполнен аппаратно). Ряд AVR микроконтроллеров имеют встроенный в арифметическую часть АЛУ перемножитель.

Основные характеристики микроконтроллеров семейства AT90S.

Тип прибора	Объем памяти (организация), бит	Объем ЭСППЗУ (СОЗУ) памяти данных, байт	Тактов. частота, МГц	Таймеры/счетчики, кол-во x разрядность	Другие отличия	Потребл. в активном/пассивном V _{cc} =3В, f=4 МГц, мА	Корпуса промышленного и коммерческого исполнений
AT90S1200	1 К (512x16)	64x8	0...12	1x8	15 линий I/O, аппаратный стек	2/0,5	20PDIP, 20SOIC, 20SSOP
AT90S2313	2 К (1Кx16)	128x8 (128x8)	0...10	1x8, 1x16	15 линий I/O, ШИМ, UART	2,5/0,8	20PDIP, 20SOIC
AT90S2323	2 К (1Кx16)	128x8 (128x8)	0...10	1x8	3 линии I/O	3,0/1,2	8PDIP, 8SOIC
AT90S2343	2 К (1Кx16)	128x8 (128x8)	0...10	1x8	5 линий I/O	3,0/1,2	8PDIP, 8SOIC
AT90S4414	4К (2Кx16)	256x8 (256x8)	0...8	1x8, 1x16	32 линии I/O, ШИМ, UART	3,5/1,0	40PDIP, 44PLCC 44TQFP
AT90S4434	4К (2Кx16)	256x8 (256x8)	0...8	2x8, 1x16	32 линии I/O, 3 ШИМ, UART, 8x10 АЦП, аналог. компаратор	3,0/1,2	40PDIP, 44PLCC, 44TQFP
AT90S8535	8К (4Кx16)	512x8 (512x8)	0...8	2x8, 1x16	32 линии I/O, 3 ШИМ, UART, 8x10 АЦП, аналог. компаратор	3,0/1,2	40PDIP, 44PLCC, 44TQFP
AT90S8515	8К (4Кx16)	256x8 (256x8)	0...8	1x8, 1x16	32 линии I/O, ШИМ, UART	3,5/1,0	40PDIP, 44PLCC 44TQFP
AT mega103	128 К (64Кx16)	4Кx8 (4Кx8)	0...6	3x16	40 линий I/O, 3 ШИМ, UART, 8x10 АЦП, аналог. компаратор	3,0/1,2	64TQFP
AT mega603	64 К (32Кx16)	2Кx8 (4Кx8)	0...6	3x16	40 линий I/O, 3 ШИМ, UART, 8x10 АЦП, аналог. компаратор	3,0/1,2	64TQFP

Все приборы семейства AT90S совместимы по исходным кодам и тактированию. Семейство обеспечено комплектом программ и системами отладки, включающими: макро-ассемблеры, отладчики/симуляторы программ, внутрисхемные эмуляторы, и отладочные устройства.

Микроконтроллеры семейства AT90S поставляются в очищенном состоянии - содержимое и Flash памяти программ и ЭСППЗУ данных находится в состоянии FF и готово к программированию..

Объединение на одном кристалле усовершенствованного 8-разрядного RISC ЦПУ с загружаемым Flash ПЗУ позволило фирме создать мощные микроконтроллеры, обеспечивающие высокую гибкость и экономичность в использовании приборов в качестве встраиваемых контроллеров.

8-разрядный КМОП RISC микроконтроллер с загружаемым Flash ПЗУ

Отличительные особенности

- AVR RISC архитектура - архитектура высокой производительности и малого потребления
- 89 команд, большинство которых выполняется за один машинный цикл
- 1 Кбайт Flash ПЗУ программ, с возможностью внутрисистемного перепрограммирования и загрузки через SPI последовательный канал, 1000 циклов стирание/запись
- 64 байта ЭСППЗУ данных, с возможностью внутрисистемной загрузки через SPI последовательный канал, 100000 циклов стирание/запись
- 32 x 8 бит регистра общего назначения
- 15 программируемых линий ввода/вывода
- Единый 16-разрядный формат команд
- Диапазон напряжений питания от 2,7 В до 6,0 В
- Полностью статический прибор - работает при тактовой частоте от 0 Гц до 12 МГц
- Две группы по рабочей частоте: 4 МГц при $V_{CC}=(2,7...6,0)V$ и 12 МГц при $V_{CC}=(4,0...6,0)V$
- Длительность командного цикла: 83,3 нс, при тактовой частоте 12 МГц
- 8-разрядный таймер/счетчик с отдельным прескалером
- Один внешний и два внутренних источника сигнала прерывания
- Программируемый сторожевой таймер с собственным встроенным генератором
- Встроенный аналоговый компаратор
- Режимы энергосбережения: пассивный (idle) и стоповый (power down)
- Блокировка режима программирования
- Возможность работы без внешних компонентов - от встроенного RC тактового генератора
- Промышленный (-40°C...85°C) и коммерческий (0°C...70°C) диапазоны температур
- 20-выводные корпуса PDIP, SOIC и SSOP

КМОП микроконтроллер AT90S1200 реализован по AVR RISC архитектуре (Гарвардская архитектура с отдельной памятью и отдельными шинами для памяти программ и данных) и совместим по исходным кодам и тактированию с 8-разрядными микроконтроллерами семейства AVR (AT90SXXX). Выполняя команды за один тактовый цикл, прибор обеспечивает производительность, приближающуюся к 1 MIPS/МГц. AVR ядро объединяет мощную систему команд с 32 регистрами общего назначения и конвейерное обращение к памяти программ. Архитектура эффективно поддерживает как языки высокого уровня, так и программы на языках ассемблера.

Микроконтроллер AT90S1200 содержит: 1 Кбайт загружаемого ПЗУ (512x16), 64 байта ЭСППЗУ, 15 линий ввода/вывода общего назначения, 32 регистра общего назначения, 8-разрядный таймер/счетчик с прескалером, систему внутренних и внешних прерываний, программируемый сторожевой таймер с внутренним генератором, последовательный порт с интерфейсом SPI для внутрисистемной загрузки. Программно

управляются два режима энергосбережения. В пассивном режиме (idle) ЦПУ останавливается, но регистры, таймер/счетчик, сторожевой таймер и система прерываний остаются активными. В стоповом режиме (power down) останавливается тактовый генератор и, следовательно, останавливаются все функции пока не поступит сигнал внешнего прерывания или аппаратного сброса, но сохраняется содержимое регистров. Встроенная загрузаемая Flash память обеспечивает внутрисистемное перепрограммирование с использованием интерфейса SPI (в последовательном низковольтном режиме) или с использованием стандартных программаторов энергонезависимой памяти (в 12-вольтовом параллельном режиме).

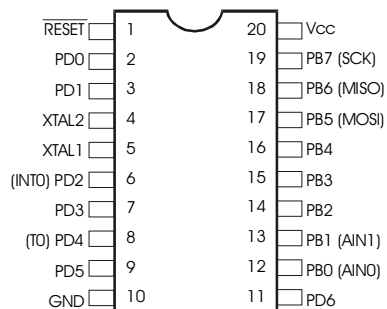
Потребление прибора в активном режиме составляет 2 мА и в пассивном режиме 0,5 мА (при $V_{CC} = 3$ В и $f = 4$ МГц). В стоповом режиме, при работающем сторожевом таймере, потребление не превышает 15 мкА.

Объединение на одном кристалле усовершенствованного 8-разрядного RISC ЦПУ с загрузаемым Flash ПЗУ позволило фирме создать мощный микроконтроллер, обеспечивающий высокую гибкость и экономичность при использовании прибора в качестве встраиваемого контроллера.

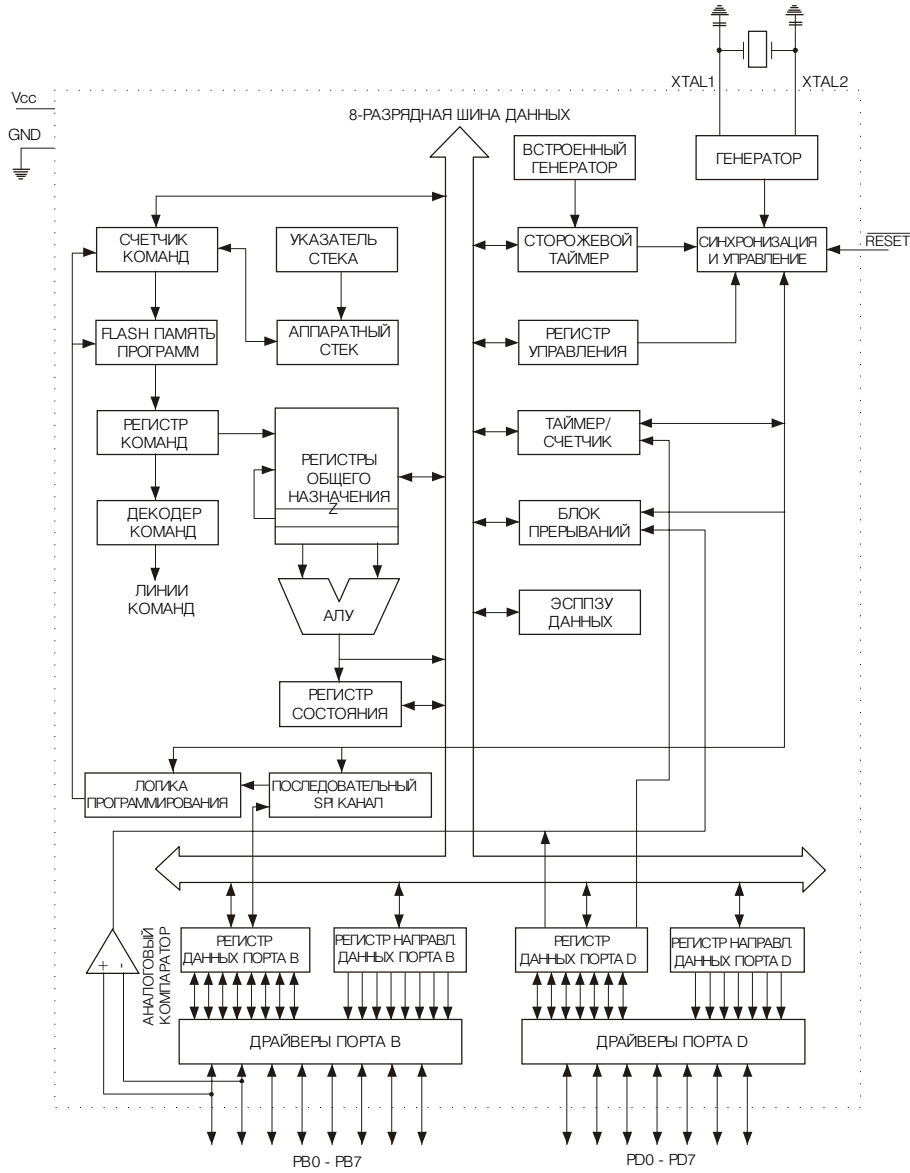
Идентификационные коды:

фирма-изготовитель (1E, по адресу 000), объем Flash памяти 1 Кбайт (90, по адресу 001), тип прибора (01, по адресу 002) если по адресу 001 стоит 90.

Корпус PDIP/SOIC/SSOP



Блок-схема микроконтроллера AT90S1200.



8-разрядный КМОП RISC микроконтроллер с загружаемым Flash ПЗУ

Отличительные особенности

- AVR RISC архитектура - архитектура высокой производительности и малого потребления
- 120 команд, большинство которых выполняется за один машинный цикл
- 2 Кбайта Flash ПЗУ программ, с возможностью внутрисистемного перепрограммирования и загрузки через SPI последовательный канал, 1000 циклов стирание/запись
- 128 байтов ЭСППЗУ данных, с возможностью внутрисистемной загрузки через SPI последовательный канал, 100000 циклов стирание/запись
- 128 байтов встроенного СОЗУ
- 32 x 8 бит регистра общего назначения
- 15 программируемых линий ввода/вывода
- 16-разрядный и 32-разрядный формат команд
- Диапазон напряжений питания от 2,7 В до 6,0 В
- Полностью статический прибор - работает при тактовой частоте от 0 Гц до 10 МГц
- Длительность командного цикла: 100 нс, при тактовой частоте 10 МГц
- 8-разрядный и 16-разрядный (с режимами сравнения и захвата) таймеры/счетчики с общим прескалером
- Функция ШИМ с 8, 9 или 10-разрядным разрешением
- Полный дуплексный UART
- Два внешних и восемь внутренних источников сигнала прерывания
- Программируемый сторожевой таймер с собственным встроенным генератором
- Встроенный аналоговый компаратор
- Режимы энергосбережения: пассивный (idle) и стоповый (power down)
- Блокировка режима программирования
- Промышленный (-40°C...85°C) и коммерческий (0°C...70°C) диапазоны температур
- 20-выводные корпуса PDIP, SOIC

КМОП микроконтроллер AT90S2313 реализован по AVR RISC архитектуре (Гарвардская архитектура с отдельной памятью и отдельными шинами для памяти программ и данных) и совместим по исходным кодам и тактированию с 8-разрядными микроконтроллерами семейства AVR (AT90SXXX). Выполняя команды за один тактовый цикл, прибор обеспечивает производительность, приближающуюся к 1 MIPS/МГц. AVR ядро объединяет мощную систему команд с 32 восьмиразрядными регистрами общего назначения и конвейерное обращение к памяти программ. Шесть из 32 регистров могут использоваться как три 16-разрядных регистра-указателя при косвенной адресации пространства памяти. Выполнение относительных переходов и команд вызова реализуется с прямой адресацией всех 2К адресного пространства. Адреса периферийных функций содержатся в пространстве памяти ввода/вывода. Архитектура эффективно поддерживает как языки высокого уровня, так и программы на языках ассемблера.

КТЦ-МК

Микроконтроллер AT90S2313 содержит: 2 Кбайт загрузаемого ПЗУ (1Кx16), 128 байтов ЭСППЗУ в отдельном пространстве памяти, 128 байтов СОЗУ, 15 линий ввода/вывода общего назначения, 32 восьмиразрядных регистра общего назначения, 8-разрядный таймер/счетчик и 16-разрядный таймер/счетчик с режимами захвата и сравнения, систему внутренних и внешних прерываний, программируемый последовательный UART, программируемый сторожевой таймер с внутренним генератором, последовательный порт с интерфейсом SPI для внутрисистемной загрузки. Программно управляются два режима энергосбережения. В пассивном режиме (idle) ЦПУ останавливается, но СОЗУ, таймеры/счетчики, порт SPI, сторожевой таймер и система прерываний остаются активными. В стоповом режиме (power down) останавливается тактовый генератор и, следовательно, останавливаются все функции, пока не поступит сигнал внешнего прерывания или аппаратного сброса, но сохраняется содержимое регистров.

Встроенная загрузаемая Flash память обеспечивает внутрисистемное перепрограммирование с использованием интерфейса SPI (в последовательном низковольтном режиме) или с использованием стандартных программаторов энергонезависимой памяти (в 12-вольтовом параллельном режиме).

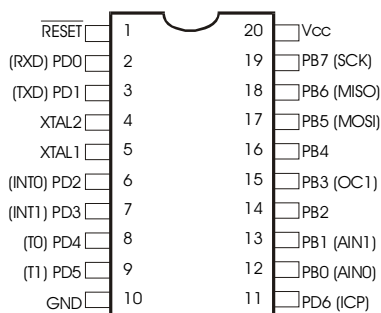
Потребление прибора в активном режиме составляет 2,5 мА и в пассивном режиме 0,8 мА (при $V_{CC} = 3$ В и $f = 4$ МГц). В стоповом режиме, при работающем сторожевом таймере, микроконтроллер потребляет 50 мкА.

Объединение на одном кристалле усовершенствованного 8-разрядного RISC ЦПУ с загрузаемым Flash ПЗУ позволило фирме создать мощный микроконтроллер, обеспечивающий высокую гибкость и экономичность в использовании прибора в качестве встраиваемого контроллера.

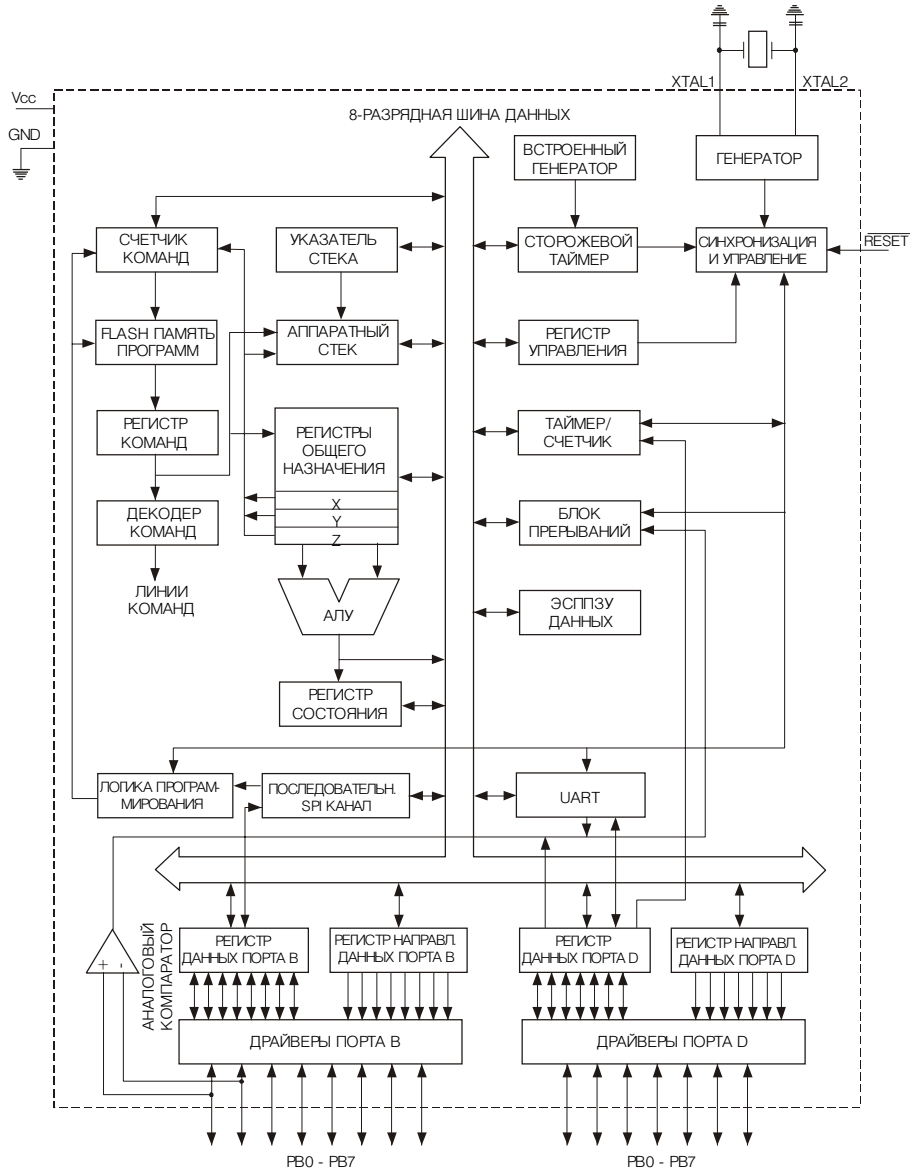
Идентификационные коды:

фирма-изготовитель (1E, по адресу 000), объем Flash памяти 2 Кбайт (91, по адресу 001), тип прибора (01, по адресу 002) если по адресу 001 стоит 90.

Корпус PDIP/SOIC



Блок-схема микроконтроллера AT90S2313.



8-разрядный КМОП RISC микроконтроллер с загружаемым Flash ПЗУ

Отличительные особенности

- AVR RISC архитектура - архитектура высокой производительности и малого потребления
- 120 команд, большинство которых выполняется за один машинный цикл
- 4 Кбайта Flash ПЗУ программ, с возможностью внутрисистемного перепрограммирования и загрузки через SPI последовательный канал, 1000 циклов стирание/запись
- 256 байтов ЭСППЗУ данных, с возможностью внутрисистемной загрузки через SPI последовательный канал, 100000 циклов стирание/запись
- 256 байтов встроенного СОЗУ
- 32 x 8 бит регистра общего назначения
- 32 программируемых линий ввода/вывода
- 16-разрядный и 32-разрядный формат команд
- Диапазон напряжений питания от 2,7 В до 6,0 В
- Полностью статический прибор - работает при тактовой частоте от 0 Гц до 8 МГц
- Длительность командного цикла: 125 нс, при тактовой частоте 8 МГц
- 8-разрядный и 16-разрядный (с режимами сравнения и захвата) таймеры/счетчики с общим прескалером
- Сдвоенный ШИМ с 8, 9 или 10-разрядным разрешением
- Программируемый полный дуплексный UART
- Два внешних и десять внутренних источников сигнала прерывания
- Программируемый сторожевой таймер с собственным встроенным генератором
- Встроенный аналоговый компаратор
- Режимы энергосбережения: пассивный (idle) и стоповый (power down)
- Блокировка режима программирования
- Промышленный (-40°C...85°C) и коммерческий (0°C...70°C) диапазоны температур
- 40-выводной корпус PDIP и 44-выводные корпуса TQFP и PLCC

КМОП микроконтроллер AT90S4414 реализован по AVR RISC архитектуре (Гарвардская архитектура с отдельной памятью и отдельными шинами для памяти программ и данных) и совместим по исходным кодам и тактированию с 8-разрядными микроконтроллерами семейства AVR (AT90SXXX). Выполняя команды за один тактовый цикл, прибор обеспечивает производительность, приближающуюся к 1 MIPS/МГц. AVR ядро объединяет мощную систему команд с 32 8-разрядными регистрами общего назначения и конвейерное обращение к памяти программ. Шесть из 32 регистров могут использоваться как три 16-разрядных регистра-указателя при косвенной адресации пространства памяти. Выполнение относительных переходов и команд вызова реализуется с прямой адресацией всех 2К адресного пространства. Адреса периферийных функций содержатся в пространстве памяти ввода/вывода. Архитектура эффективно поддерживает как языки высокого уровня, так и программы на языках ассемблера.

Микроконтроллер AT90S4414 содержит: 4 Кбайт загрузаемого ПЗУ (2Кx16), 256 байтов СОЗУ и 256 байтов ЭСППЗУ, с возможностью наращивания памяти данных до 64К за счет внешних ИС СОЗУ, 32 линии ввода/вывода общего назначения, 32 регистра общего назначения, 8-разрядный таймер/счетчик и 16-разрядный таймер/счетчик с режимом захвата и сравнения, систему внутренних и внешних прерываний, программируемый последовательный UART, программируемый сторожевой таймер с внутренним генератором, последовательный порт с интерфейсом SPI для внутрисистемной загрузки и для связи с внешними устройствами. Программно управляются два режима энергосбережения. В пассивном режиме (idle) ЦПУ останавливается, но СОЗУ, таймеры/счетчики, порт SPI, сторожевой таймер и система прерываний остаются активными. В стоповом режиме (power down) останавливается тактовый генератор и, следовательно останавливаются все функции, пока не поступит сигнал внешнего прерывания или аппаратного сброса, но сохраняется содержимое регистров.

Встроенная загрузаемая Flash память обеспечивает внутрисистемное перепрограммирование с использованием интерфейса SPI (в последовательном низковольтном режиме) или с использованием стандартных программаторов энергонезависимой памяти (в 12-вольтовом параллельном режиме).

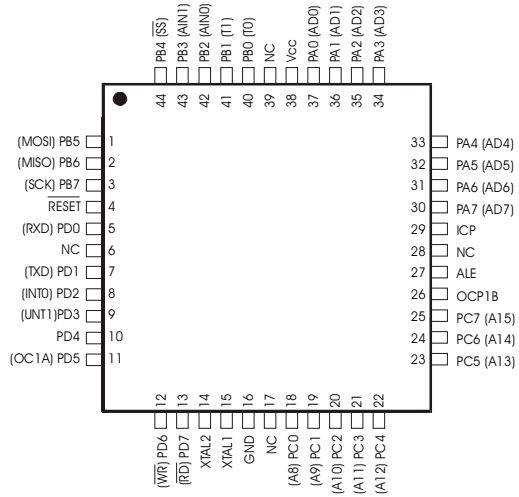
Потребление прибора в активном режиме составляет 3,5 мА и в пассивном режиме 1 мА (при VCC =3 В и f=4 МГц). В стоповом режиме, при работающем сторожевом таймере, микроконтроллер потребляет 50 мкА.

Объединение на одном кристалле усовершенствованного 8-разрядного RISC ЦПУ с загрузаемым Flash ПЗУ позволило фирме создать мощный микроконтроллер, обеспечивающий высокую гибкость и экономичность в использовании прибора в качестве встраиваемого контроллера.

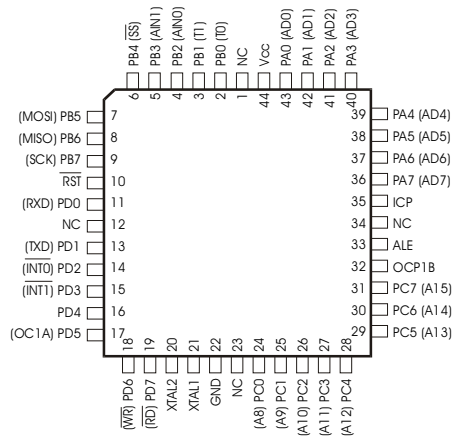
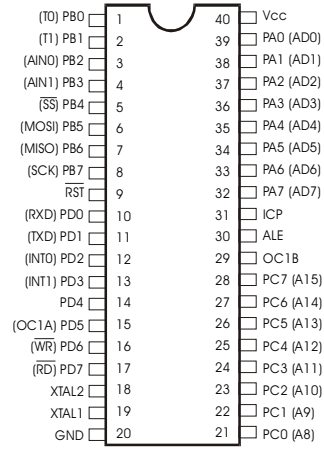
Идентификационные коды:

фирма-изготовитель (1E, по адресу 000), объем Flash памяти 4 Кбайт (92, по адресу 001), тип прибора (01, по адресу 002) если по адресу 001 стоит 92.

Копиыс TQFP

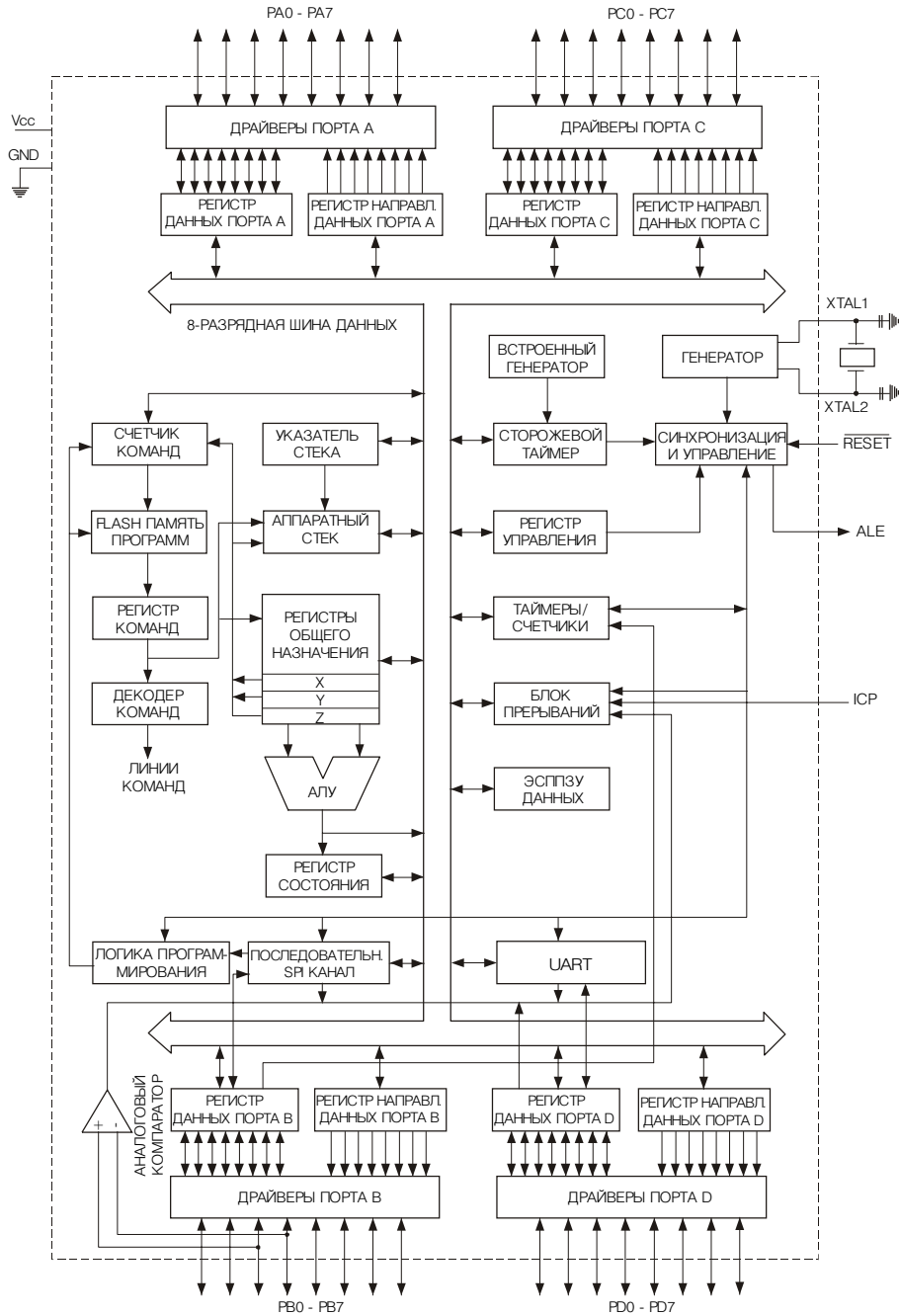


Копиыс PDIP



Копиыс PLCC

Блок-схема микроконтроллера AT90S4414.



AT90S4434
AT90LS4434
AT90S8535
AT90LS8535

8-разрядные КМОП RISC микроконтроллеры с внутрисистемно программируемым Flash ПЗУ

Отличительные особенности

- AVR RISC архитектура - архитектура высокой производительности и малого потребления
- 118 команд, большинство которых выполняется за один машинный цикл
- 4 Кбайта (AT90S/LS4434) и 8 Кбайт (AT90S/LS8535) Flash ПЗУ программ, с возможностью внутрисистемного перепрограммирования и загрузки через SPI последовательный канал, 1000 циклов стирание/запись
- 256 байтов (AT90S/LS4434) и 512 байтов (AT90S/LS8535) ЭСППЗУ данных, с возможностью внутрисистемного перепрограммирования и загрузки через SPI последовательный канал, 100000 циклов стирание/запись
- 256 байтов (AT90S/LS4434) и 512 байтов (AT90S/LS8535) встроенного СОЗУ
- 32 x 8 бит регистра общего назначения
- 32 программируемые линии ввода/вывода
- 8-канальный 10-разрядный аналого-цифровой преобразователь
- 16-разрядный и 32-разрядный формат команд
- Программируемый полный дуплексный UART
- Диапазон напряжений питания от 2,7 В до 6,0 В (AT90LS4434/AT90LS8535) и от 4,0 В до 6,0 В (AT90S4434/AT90S8535)
- Полностью статический прибор - работает при тактовой частоте от 0 Гц до 8 МГц (AT90S4434/AT90S8535) и при тактовой частоте от 0 Гц до 4 МГц (AT90LS4434/AT90LS8535)
- Производительность до 8 MIPS при частоте 8 МГц
- Два 8-разрядных таймера/счетчика с отдельным прескалером и режимом сравнения
- 16-разрядный (с режимами сравнения и захвата) таймер/счетчик с отдельным прескалером
- Три ШИМ канала
- Внешние и внутренние источники сигналов прерывания
- Программируемый сторожевой таймер с собственным встроенным генератором
- Встроенный аналоговый компаратор
- Встроенные часы реального времени с собственным встроенным генератором и режимом счетчика
- Блокировка режима программирования
- Режимы энергосбережения: пассивный (idle), экономичный (power save) и стоповый (power down)
- Встроенная схема сброса по подаче питания
- Промышленный (-40°C...85°C) и коммерческий (0°C...70°C) диапазоны температур
- 40-выводной корпус PDIP и 44-выводные PLCC и TQFP

КМОП микроконтроллеры AT90S/LS4434 и AT90S/LS8535 реализованы по AVR RISC архитектуре (Гарвардская архитектура с отдельной памятью и отдельными шинами для памяти программ и данных). Выполняя команды за один тактовый цикл, приборы обеспечивают производительность, приближающуюся к 1 MIPS/МГц. AVR ядро объединяет мощную систему команд с 32 8-разрядными регистрами общего назначения и конвейерное обращение к памяти программ. Шесть из 32 регистров могут использоваться как три 16-разрядных регистра-указателя при косвенной адресации пространства памяти. Выполнение относительных переходов и команд вызова реализуется с прямой адресацией всего объема (2К/4К) адресного пространства. Адреса периферийных функций содержатся в пространстве памяти ввода/вывода. Архитектура эффективно поддерживает как языки высокого уровня, так и программы на языках ассемблера.

Микроконтроллеры содержат: 4 Кбайт внутрисистемно программируемого Flash ПЗУ (2Кx16), 256 байтов СОЗУ и 256 байтов внутрисистемно программируемого ЭСППЗУ (микроконтроллеры AT90S/LS4434), и 8 Кбайт внутрисистемно программируемого Flash ПЗУ (4Кx16), 512 байтов СОЗУ и 512 байтов внутрисистемно программируемого ЭСППЗУ (микроконтроллеры AT90S/LS8535), 32 линии ввода/вывода общего назначения, 32 регистра общего назначения, два 8-разрядных таймера/счетчика и один 16-разрядный таймер/счетчик с режимом захвата и сравнения, часы реального времени, 8-канальный 10-разрядный аналого-цифровой преобразователь, систему внутренних и внешних прерываний, программируемый последовательный UART, программируемый сторожевой таймер с внутренним генератором, последовательный порт с интерфейсом SPI для внутрисистемной загрузки и для связи с внешними устройствами. Программно управляются три режима энергосбережения. В пассивном режиме (idle) ЦПУ останавливается, но СОЗУ, таймеры/счетчики, порт SPI, сторожевой таймер и система прерываний остаются активными. В стоповом режиме (power down) останавливается тактовый генератор и, следовательно, останавливаются все функции, пока не поступит сигнал внешнего прерывания или аппаратного сброса, но сохраняется содержимое регистров. В экономичном режиме (power save), при остановленном микроконтроллере, продолжает работать генератор таймера, что обеспечивает сохранность временной базы.

Встроенная Flash память программ и ЭСППЗУ данных могут перепрограммироваться непосредственно в системе посредством интерфейса SPI (в последовательном низковольтном режиме) или программироваться стандартными программаторами энергонезависимой памяти (в 12-вольтовом параллельном режиме). Программирование микроконтроллеров ведется по-байтово.

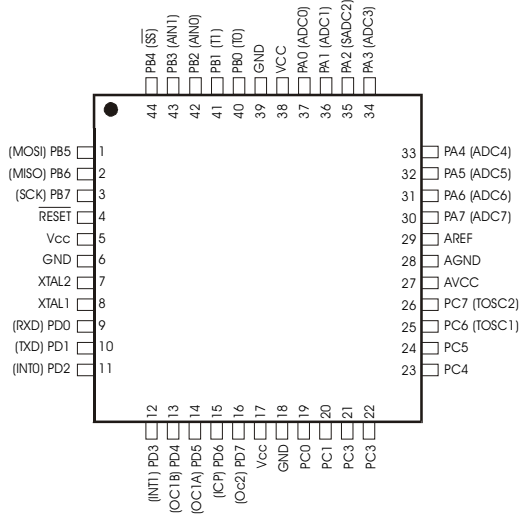
Максимальное потребление приборов в активном режиме составляет 3,0 мА и в пассивном режиме 1,2 мА (при VCC =3 В и f=4 МГц). В стоповом режиме, при работающем сторожевом таймере, микроконтроллер потребляет 15 мкА.

Объединение на одном кристалле усовершенствованного 8-разрядного RISC ЦПУ с загружаемым Flash ПЗУ позволило фирме создать мощный микроконтроллер, обеспечивающий высокую гибкость и экономичность в использовании прибора в качестве встраиваемого контроллера.

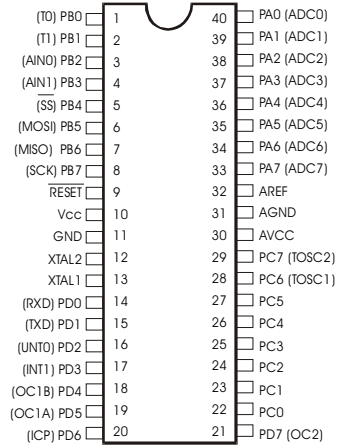
Идентификационные коды:

для прибора 90S4434 - фирма-изготовитель (1Е, по адресу 00), объем Flash памяти 4 Кбайт (92, по адресу 01) и тип прибора (03, по адресу 02) если по адресу 001 стоит 92, для прибора 90S8535 - фирма-изготовитель (1Е, по адресу 00), объем Flash памяти 8 Кбайт (93 по адресу 01) и тип прибора (03, по адресу 002) если по адресу 001 стоит 93.

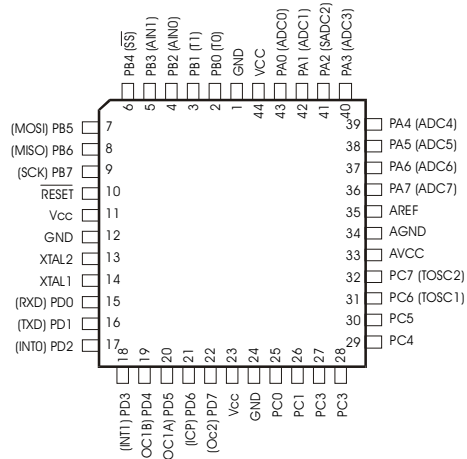
Копыс TQFP



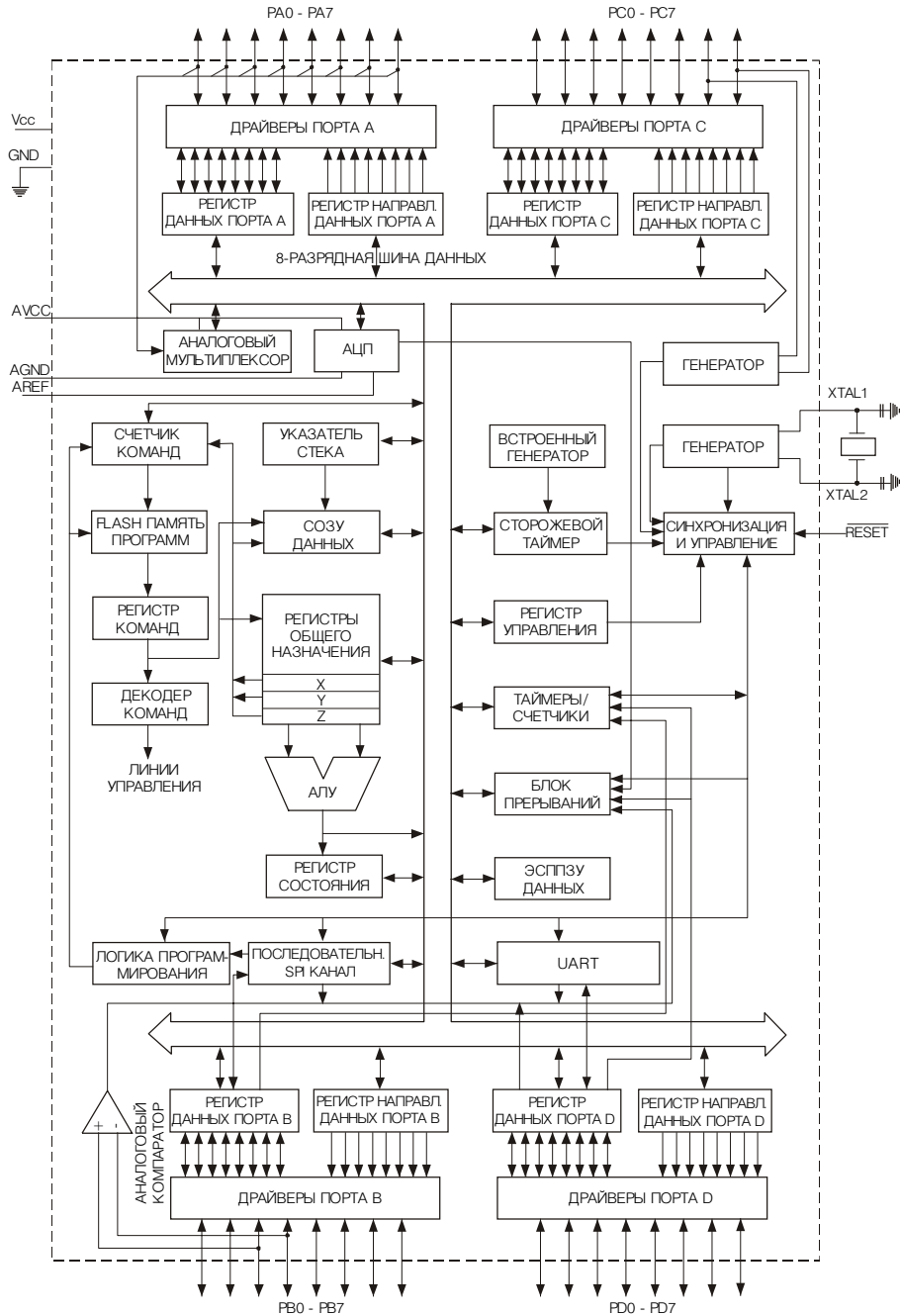
Копыс PDIP



Копыс PLCC



Блок-схема микроконтроллеров AT90S/LS4434 и AT90S/LS8535.



8-разрядный КМОП RISC микроконтроллер с загружаемым Flash ПЗУ

Отличительные особенности

- AVR RISC архитектура - архитектура высокой производительности и малого потребления
- 120 команд, большинство которых выполняется за один машинный цикл
- 8 Кбайта Flash ПЗУ программ, с возможностью внутрисистемного перепрограммирования и загрузки через SPI последовательный канал, 1000 циклов стирание/запись
- 512 байтов ЭСППЗУ данных, с возможностью внутрисистемной загрузки через SPI последовательный канал, 100000 циклов стирание/запись
- 512 байтов встроенного СОЗУ
- 32 x 8 бит регистра общего назначения
- 32 программируемых линий ввода/вывода
- 16-разрядный и 32-разрядный формат команд
- Диапазон напряжений питания от 2,7 В до 6,0 В
- Полностью статический прибор - работает при тактовой частоте от 0 Гц до 8 МГц
- Длительность командного цикла: 125 нс, при тактовой частоте 8 МГц
- 8-разрядный и 16-разрядный (с режимами сравнения и захвата) таймеры/счетчики с общим прескалером
- Сдвоенный ШИМ с 8, 9 или 10-разрядным разрешением
- Программируемый полный дуплексный UART
- Два внешних и десять внутренних источников сигнала прерывания
- Программируемый сторожевой таймер с собственным встроенным генератором
- Встроенный аналоговый компаратор
- Режимы энергосбережения: пассивный (idle) и стоповый (power down)
- Блокировка режима программирования
- Промышленный (-40°C...85°C) и коммерческий (0°C...70°C) диапазоны температур
- 40-выводной корпус PDIP и 44-выводные корпуса TQFP и PLCC

КМОП микроконтроллер AT90S8515 реализован по AVR RISC архитектуре (Гарвардская архитектура с отдельной памятью и отдельными шинами для памяти программ и данных) и совместим по исходным кодам и тактированию с 8-разрядными микроконтроллерами семейства AVR (AT90SXXX). Выполняя команды за один тактовый цикл, прибор обеспечивает производительность, приближающуюся к 1 MIPS/МГц. AVR ядро объединяет мощную систему команд с 32 8-разрядными регистрами общего назначения и конвейерное обращение к памяти программ. Шесть из 32 регистров могут использоваться как три 16-разрядных регистра-указателя при косвенной адресации пространства памяти. Выполнение относительных переходов и команд вызова реализуется с прямой адресацией всех 4К адресного пространства. Адреса периферийных функций содержатся в пространстве памяти ввода/вывода. Архитектура эффективно поддерживает как языки высокого уровня, так и программы на языках ассемблера.

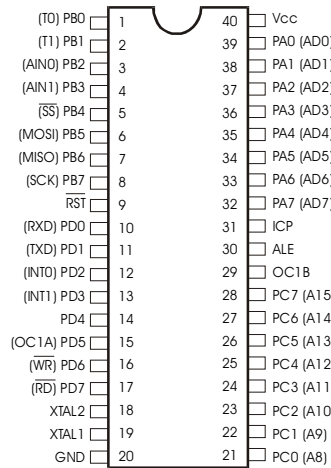
Встроенная загружаемая Flash память обеспечивает внутрисистемное перепрограммирование с использованием интерфейса SPI (в последовательном низковольтном режиме) или с использованием стандартных программаторов энергонезависимой памяти (в 12-вольтовом параллельном режиме).

Потребление прибора в активном режиме составляет 3,5 мА и в пассивном режиме 1 мА (при VCC =3 В и f=4 МГц). В стоповом режиме, при работающем сторожевом таймере, микроконтроллер потребляет 50 мкА.

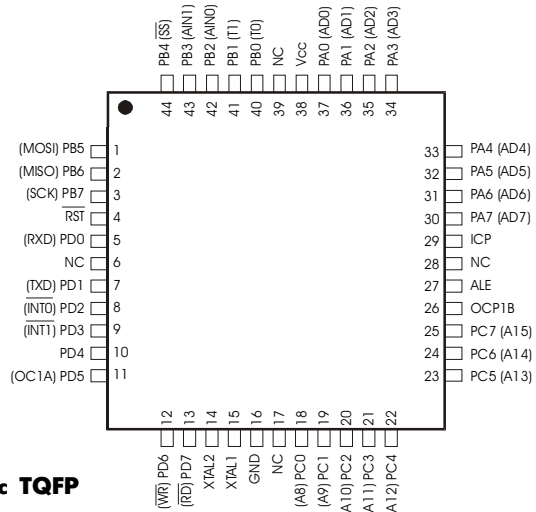
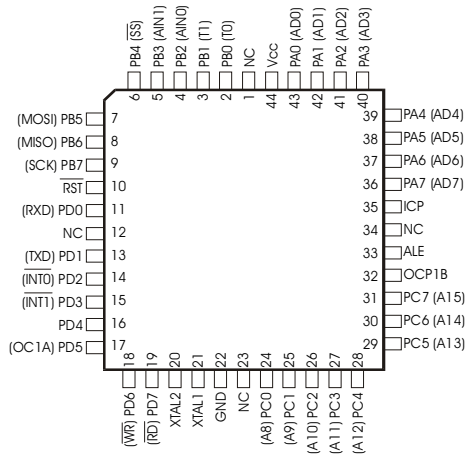
Идентификационные коды:

фирма-изготовитель (1E, по адресу 000), объем Flash памяти 8 Кбайт (93, по адресу 001), тип прибора (01, по адресу 002) если по адресу 001 стоит 93.

Корпус PDIP

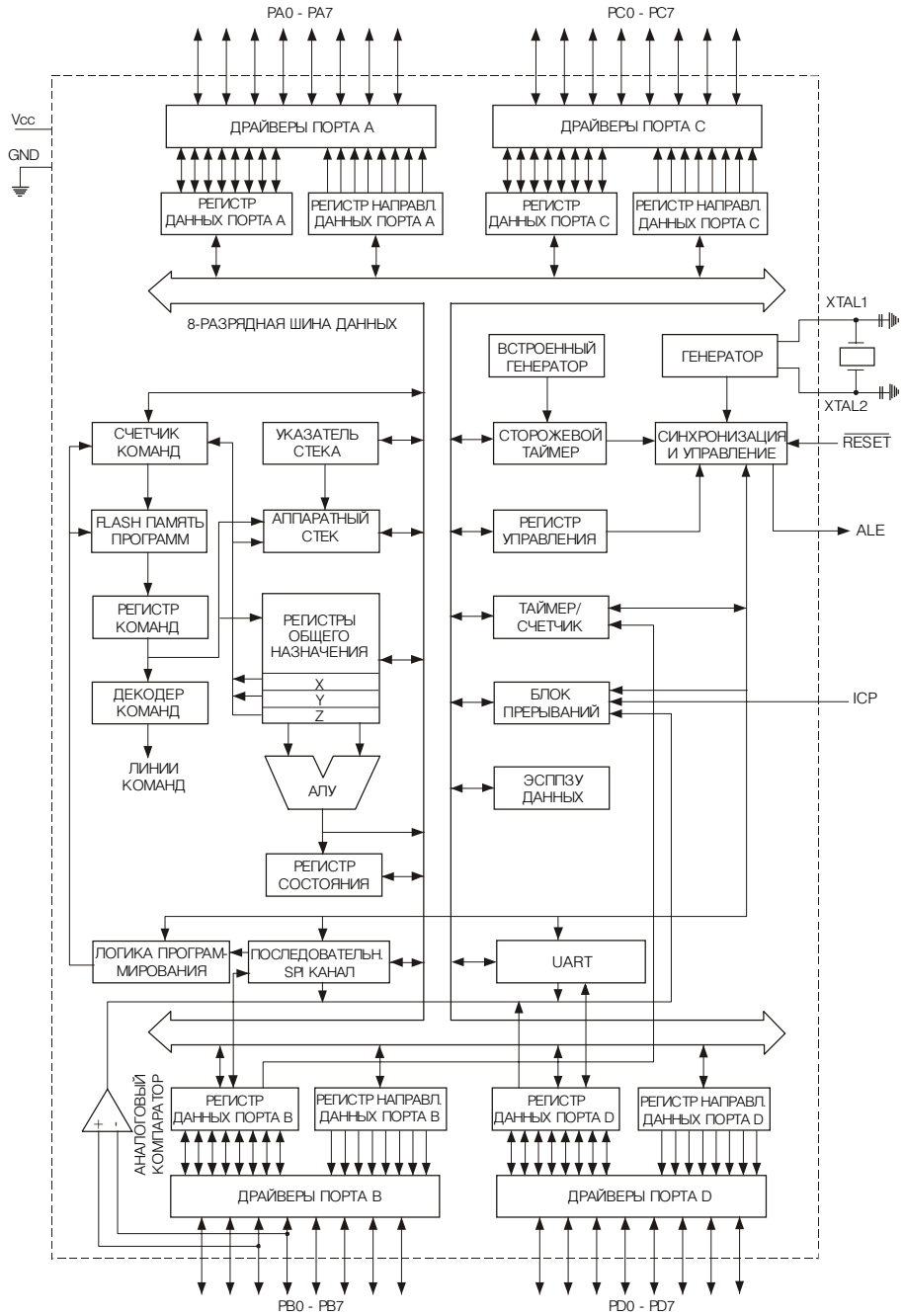


Корпус PLCC



Корпус TQFP

Блок-схема микроконтроллера AT90S8515.



AT90S2323**AT90LS2323****AT90S2343****AT90LS2343**

8-разрядные КМОП RISC микроконтроллеры с внутрисистемно программируемым Flash ПЗУ

Отличительные особенности

- AVR RISC архитектура - архитектура высокой производительности и малого потребления
- 118 команд, большинство которых выполняется за один машинный цикл
- 2 Кбайта Flash ПЗУ программ, с возможностью внутрисистемного перепрограммирования и загрузки через SPI последовательный канал, 1000 циклов стирание/запись
- 128 байтов ЭСППЗУ данных, 100000 циклов стирание/запись
- 128 байтов встроенного ОЗУ
- 32 x 8 бит регистра общего назначения
- 3 программируемые линии I/O (AT90S/LS2323)
- 5 программируемых линий I/O (AT90S/LS2343)
- 16-разрядный и 32-разрядный формат команд
- Диапазон напряжений питания от 2,7 В до 6,0 В (AT90LS2323/ AT90LS2343) и от 4,0 В до 6,0 В (AT90S2323/ AT90S2343)
- Полностью статические приборы - работают при тактовой частоте от 0 Гц до 10 МГц (AT90S2323/ AT90S2343) и от 0 Гц до 4 МГц (AT90LS2323/ AT90LS2343)
- Производительность до 10 MIPS при тактовой частоте 10 МГц
- 8-разрядный таймер/счетчик с отдельным прескалером
- Внешние и внутренние источники сигналов прерывания
- Программируемый сторожевой таймер с собственным встроенным генератором
- Режимы энергосбережения: пассивный (idle) и стоповый (power down)
- Программная блокировка программирования Flash памяти и защиты ЭСППЗУ данных
- Встроенная схема сброса по включению питания
- 8-выводные корпуса PDIP, SOIC
- Промышленный (-40°C...85°C) и коммерческий (0°C...70°C) диапазоны температур

КМОП микроконтроллеры AT90S/LS2323 и AT90S/LS2343 реализованы по AVR RISC архитектуре (Гарвардская архитектура с отдельной памятью и отдельными шинами для памяти программ и данных). Выполняя команды за один тактовый цикл, прибор обеспечивает производительность, приближающуюся к 1 MIPS/МГц. AVR ядро объединяет мощную систему команд с 32 восьмиразрядными регистрами общего назначения и конвейерное обращение к памяти программ. Шесть из 32 регистров могут использоваться как три 16-разрядных регистра-указателя при косвенной адресации пространства памяти. Выполнение относительных переходов и команд вызова реализуется с прямой адресацией всех 2K адресного пространства. Адреса периферийных функций содержатся в пространстве памяти ввода/вывода. Архитектура

КТЦ-МК

эффективно поддерживает как языки высокого уровня, так и программы на языках ассемблера.

Микроконтроллеры AT90S/LS2323 и AT90S/LS2343 содержат: 2 Кбайт Flash ПЗУ (1Кx16), 128 байтов ЭСППЗУ в отдельном пространстве памяти, 128 байтов СОЗУ, 3 линии (AT90S/LS2323) или 5 линий (AT90S/LS2343) ввода/вывода общего назначения, 32 восьмиразрядных регистра общего назначения, 8-разрядный таймер/счетчик, систему внутренних и внешних прерываний, программируемый сторожевой таймер с внутренним генератором, последовательный порт с интерфейсом SPI для внутрисистемной загрузки. Программно управляются два режима энергосбережения. В пассивном режиме (idle) ЦПУ останавливается, но СОЗУ, таймер/счетчик, порт SPI, сторожевой таймер и система прерываний остаются активными. В стоповом режиме (power down) останавливается тактовый генератор и, следовательно, останавливаются все функции, пока не поступит сигнал внешнего прерывания или аппаратного сброса, но сохраняется содержимое регистров.

Встроенная Flash память обеспечивает внутрисистемное перепрограммирование с использованием интерфейса SPI (в последовательном низковольтном режиме) или в 12-вольтовом специальном режиме. Программирование ведется по-байтово и в том и в другом режиме.

Потребление приборов в активном режиме составляет 3,0 мА и в пассивном режиме 1,2 мА (при VCC =3 В и f=4 МГц). В стоповом режиме, при работающем сторожевом таймере, микроконтроллер потребляет 15 мкА.

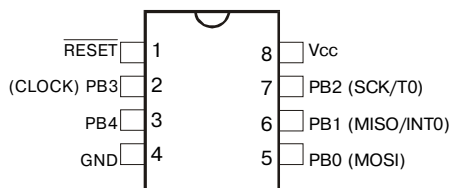
Объединение на одном кристалле усовершенствованного 8-разрядного RISC ЦПУ с загружаемым Flash ПЗУ позволило фирме создать мощный микроконтроллер, обеспечивающий высокую гибкость и экономичность при использовании прибора в качестве встраиваемого контроллера.

Идентификационные коды:

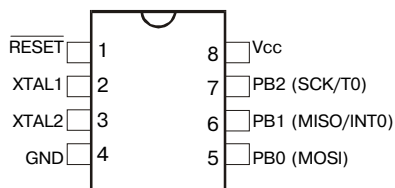
фирма-изготовитель (1E, по адресу 00), объем Flash памяти 2 Кбайт (91, по адресу 01), тип прибора AT90S/LS2343 (03, по адресу 02) если по адресу 01 стоит 97 и AT90S/LS2323 (04, по адресу 02) если по адресу 01 стоит 97

PDIP/SOIC

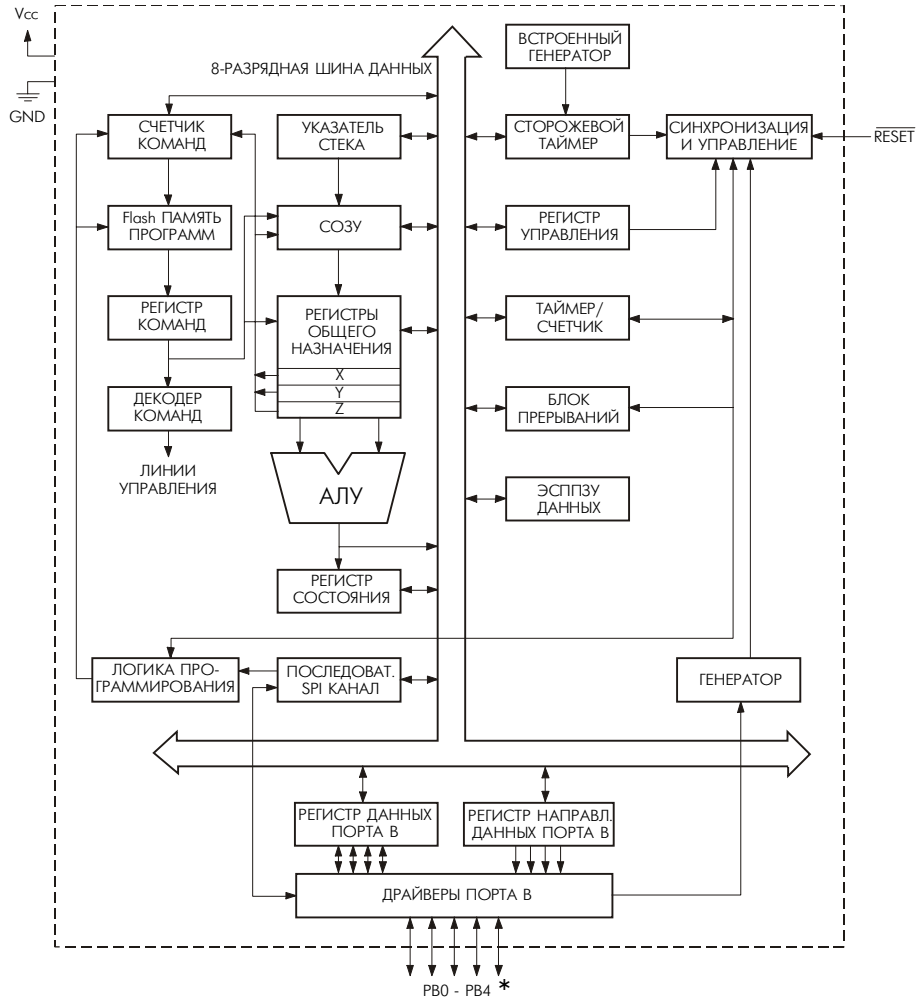
Корпус AT90S/LS2343



Корпус AT90S/LS2323



Блок-схема микроконтроллеров AT90S/LS2323 и AT90S/LS2343.



* PB0 - PB2 у микроконтроллеров AT90S/LS2323

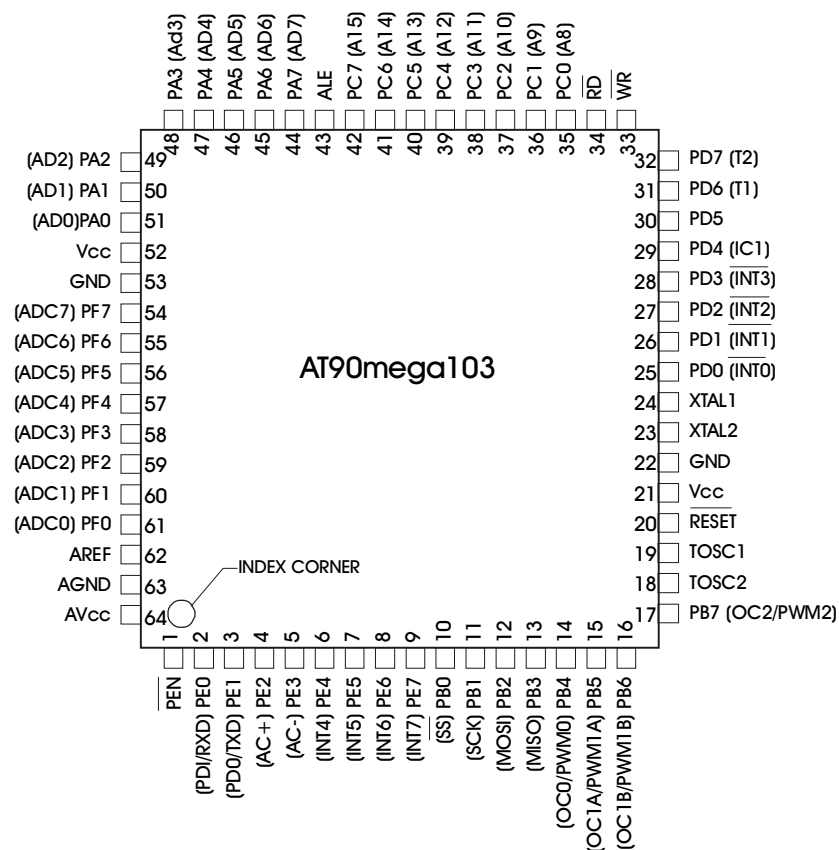
**ATmega603,
ATmega603L,
ATmega103,
ATmega103L**

**8-разрядные микроконтроллеры
с Flash памятью емкостью 64/128Кбайт**

Отличительные особенности

- Использована AVR расширенная RISC архитектура
- Мощный набор из 121 команды, большинство которых выполняется за один машинный цикл
- Емкость внутрисистемно программируемой Flash памяти 64 Кбайт (ATmega603/L) и 128 Кбайт (ATmega103/L), 1000 циклов стирания/записи
- SPI интерфейс внутрисистемного программирования
- Емкость встроенной EEPROM 2 Кбайт (ATmega603/L) и 4 Кбайт (ATmega103/L), 100000 циклов стирания/записи
- Встроенная RAM емкостью 4 Кбайт
- 32 8-разрядных регистра общего назначения, набор регистров управления периферией
- 32 программируемых линии I/O, 8 линий выхода, 8 линий входа
- Программируемые последовательные UART и SPI интерфейсы
- Диапазон напряжений питания от 2,7 В до 6,0 В (ATmega603L/ ATmega103L) и от 4,0 В до 6,0 В (ATmega603/ ATmega103)
- Диапазон тактовых частот от 0 до 4 МГц (ATmega603L/ ATmega103L) и от 0 до 6 МГц (ATmega603/ ATmega103)
- Производительность до 6 MIPS при частоте 6 МГц
- Встроенная система реального времени с отдельным генератором
- Два 8-разрядных таймера/счетчика с отдельным предделителем и ШИМ
- 16-разрядный таймер/счетчик с отдельным предделителем, режимами захвата/сравнения и двойным ШИМ с разрядностью 8, 9 или 10 разрядов
- Программируемый сторожевой таймер с встроенным генератором
- Встроенный аналоговый компаратор
- 8-канальный 10-разрядный аналого-цифровой преобразователь
- Режимы энергосбережения Idle, Power Save и Power Down
- Программная установка тактовой частоты
- Программная блокировка защиты программных средств

Разводка выводов



Описание

Приборы ATmega603/103 являются 8-разрядными CMOS микроконтроллерами с AVR усовершенствованной RISC архитектурой. Выполняя большинство команд за один тактовый цикл, микроконтроллеры ATmega603/103 обеспечивают производительность 1 MIPS на каждый мегагерц тактовой частоты, что позволяет разработчикам оптимизировать потребление, зависящее в основном от тактовой частоты.

AVR ядро базируется на усовершенствованной RISC архитектуре, с регистровым файлом быстрого доступа, содержащим 32 регистра общего назначения, непосредственно связанных с арифметико-логическим устройством (ALU), и мощной системой команд. За один тактовый цикл из регистрового файла извлекаются два операнда, выполняется команда и результат записывается в регистр назначения. Такая высокоэффективная архитектура обеспечивает производительность почти в десять раз большую, чем стандартные CISC микроконтроллеры.

Микроконтроллеры АТmega603/103 располагают следующими возможностями: 64/128 Кбайт внутрисистемно программируемой Flash памяти программ, 2/4 Кбайт EEPROM данных, 4 Кбайт SRAM данных, 32 линии I/O общего назначения, 8 линий входа, 8 линий выхода, 32 рабочих регистра общего назначения, 4 гибких таймера/счетчика с режимами сравнения, PWM и UART, программируемый сторожевой таймер с встроенным собственным генератором, последовательный SPI порт и три программно устанавливаемых режима энергосбережения. В режиме Idle останавливается центральный процессор, но продолжают работать SRAM, таймеры/счетчики, порт SPI и система прерываний. В режиме Power Down сохраняется содержимое регистров, но останавливается тактовый генератор и до поступления сигнала прерывания или аппаратного сброса запрещается выполнение всех функций микроконтроллера. В режиме Power Save все устройства находятся в режиме «сна», но генератор таймера продолжает работать, обеспечивая сохранность временной базы

Приборы изготавливаются по технологии энергонезависимой памяти фирмы Atmel. Встроенная ISP Flash память программ может быть перепрограммирована непосредственно в системе, с использованием последовательного SPI интерфейса, или с помощью обычных программаторов энергонезависимой памяти. Объединив 8-разрядное RISC CPU с внутрисистемно программируемой Flash памятью большого объема, фирма создала семейство мощных микроконтроллеров, обеспечивающих реализацию недорогих и очень удобных решений для большого количества встраиваемых применений. Семейство АТmega603/103 поддерживается большим количеством средств разработки программ и систем, включающих: С-компиляторы, макроассемблеры, отладчики/симуляторы программ, внутрисхемные эмуляторы и отладочные устройства.

СРАВНЕНИЕ АТmega603 и АТmega103

Микроконтроллер АТmega603 оснащен внутрисистемно программируемой Flash памятью емкостью 64 Кбайт, 2 Кбайт EEPROM и 4 Кбайт SRAM и не выполняет команду ELPM. Микроконтроллер АТmega103 оснащен внутрисистемно программируемой Flash памятью емкостью 128 Кбайт, 4 Кбайт EEPROM и 4 Кбайт SRAM. В систему команд этого микроконтроллера включена команда ELPM, необходимая для обеспечения непрерывного табличного поиска в старшей половине адресов Flash памяти. В Табл. 1 представлены отличия в объеме памяти этих двух приборов.

Таблица 1

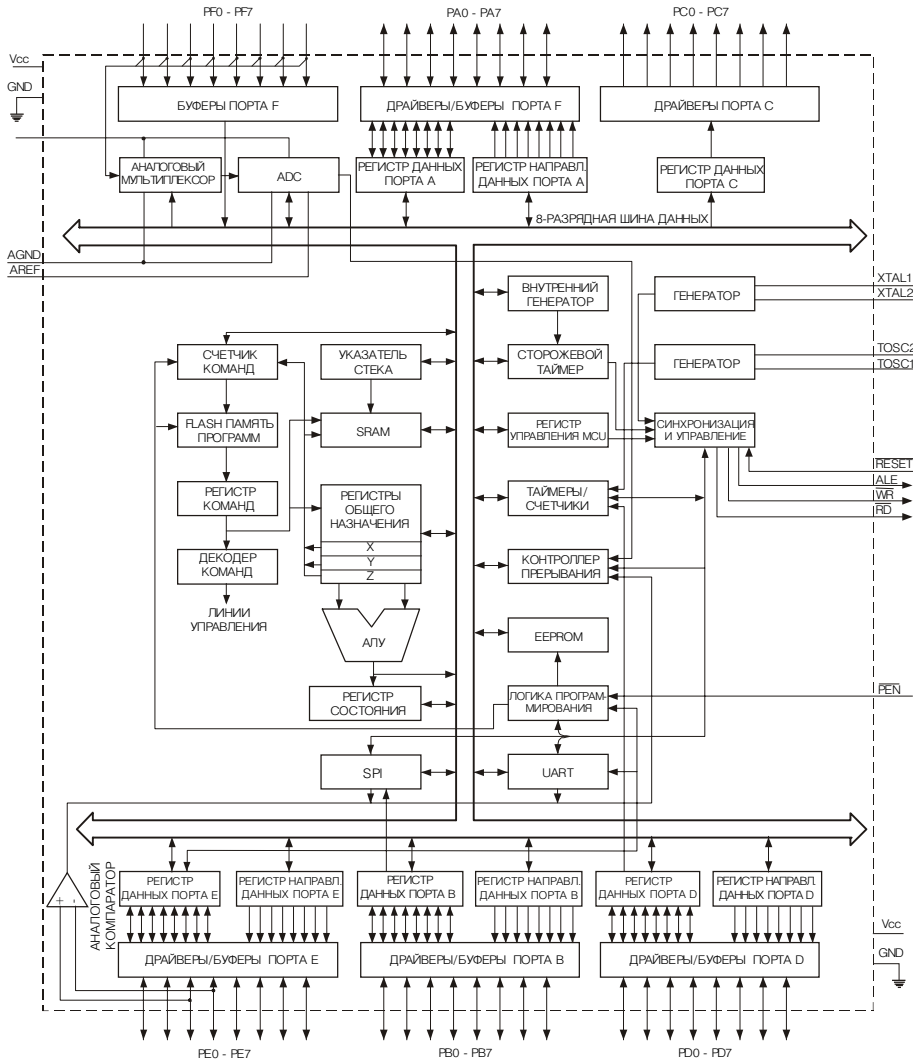
Тип прибора	Объем Flash памяти	Объем EEPROM	Объем SRAM
АТmega603	64 Кбайт	2 Кбайт	4 Кбайт
АТmega103	128 Кбайт	4 Кбайт	4 Кбайт

Назначение выводов

VCC Напряжение питания

GND Земля

Port A (PA7..PA0) 8-разрядный двунаправленный порт I/O. К выходам порта могут быть подключены встроенные нагрузочные резисторы (отдельно к каждому разряду). Выходные буферы обеспечивают протекающий ток 20 мА и способны напрямую управлять LED индикатором. При использовании выводов порта в качестве входов и установке внешним сигналом в низкое состояние, ток будет вытекать только при подключенных встроенных нагрузочных резисторах.



Блок схема микроконтроллеров ATmega603/103.

Порт А, при наличии внешней SRAM, используется в качестве мультиплексируемой шины адреса/данных.

Port B (PB7. .PB0) 8-разрядный двунаправленный порт I/O со встроенными нагрузочными резисторами. Выходные буферы обеспечивают втекающий ток 20 мА. При использовании выводов порта в качестве входов и установке внешним сигналом в низкое состояние, ток будет вытекать только при подключенных встроенных нагрузочных резисторах. Порт В используется также при реализации различных специальных функций.

Port C (PC7. .PC0) 8-разрядный порт выхода. Выходные буферы обеспечивают втекающий ток 20 мА.

Порт С используется также как выходы адреса при использовании внешней SRAM.

Port D (PD7..PД0) 8-разрядный двунаправленный порт I/O со встроенными нагрузочными резисторами. Выходные буферы обеспечивают втекающий ток 20 мА. При использовании выводов порта в качестве входов и установке внешним сигналом в низкое состояние, ток будет вытекать только при подключенных встроенных нагрузочных резисторах.

Port E (PE7..PE0) 8-разрядный двунаправленный порт I/O со встроенными нагрузочными резисторами. Выходные буферы обеспечивают втекающий ток 20 мА. При использовании выводов порта в качестве входов и установке внешним сигналом в низкое состояние, вытекающий через них ток обеспечивается только при подключенных встроенных нагрузочных резисторах.

Port F (PF7..PF0) 8-разрядный порт входа. Входы порта используются также как аналоговые входы аналого-цифрового преобразователя.

RESET Вход сброса. Для выполнения сброса необходимо удерживать низкий уровень на входе в течение двух машинных циклов.

XTAL1 Вход инвертирующего усилителя генератора и вход схемы встроенного генератора тактовой частоты.

XTAL2 Выход инвертирующего усилителя генератора.

TOSC1 Вход инвертирующего усилителя генератора таймера/счетчика.

TOSC2 Выход инвертирующего усилителя генератора таймера/счетчика.

WR Строб записи внешней SRAM.

RD Строб чтения внешней SRAM.

ALE Строб разрешения фиксации адреса, используемый для разрешения внешней памяти. Строб ALE используется для фиксации младшего байта адреса в защелках адреса в течение первого цикла обращения, в течение второго цикла обращения, при обращении к данным, используются выводы AD0- AD 7.

AVCC Напряжение питания аналого-цифрового преобразователя. Вывод подсоединяется к внешнему VCC через низкочастотный фильтр. Подробности см. на рис. 47.

AREF Вход аналогового напряжения сравнения для аналого-цифрового преобразователя. На этот вывод, для обеспечения работы аналого-цифрового преобразователя, подается напряжение в диапазоне между AGND и AVCC.

AGND Этот вывод должен быть подсоединен к отдельной аналоговой земле, если плата оснащена ею. В ином случае вывод подсоединяется к общей земле.

PEN Вывод разрешения программирования в низковольтном последовательном режиме программирования. При удержании этого вывода на низком уровне во время сброса по включении питания, прибор перейдет в режим программирования по последовательному каналу.

Тактовый генератор

XTAL1 и XTAL2 являются входом и выходом, соответственно, инвертирующего усилителя, который с использованием кварцевого кристалла или керамического резонатора работает как встроенный генератор, как показано на Рис. 2, При использовании внешнего источника тактовой частоты вывод XTAL2 должен остаться свободным, сигнал подается на вывод XTAL1, как показано на Рис. 3.

Кварцевый кристалл генератора таймера подсоединяется непосредственно к выводам OSC1 и OSC2. Внешние конденсаторы не требуются. Генератор оптимизирован под часовой кварц с частотой 32,768 КГц. Внешний тактовый сигнал,

подаваемый на эти выводы, поступает на усилитель с полосой пропускания 256 КГц. Таким образом частота внешнего сигнала должна находиться в диапазоне от 0 до 256 КГц.

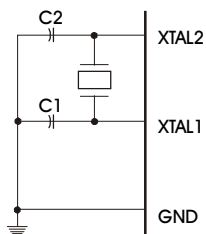


Рис. 2 Подсоединение тактового генератора.

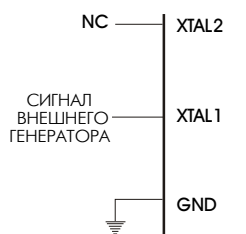


Рис. 3 Подсоединение внешнего источника тактового сигнала

Архитектура микроконтроллеров ATmega603/103

Файл регистров быстрого доступа, содержит 32 8-разрядных рабочих регистра общего назначения связанных непосредственно с ALU. За один тактовый цикл из файла регистров выбираются два операнда, выполняется операция и результат вновь возвращается в файл регистров.

Шесть из 32 регистров могут быть использованы как три 16-разрядных регистра указателя косвенной адресации адресного пространства данных, обеспечивающие эффективное вычисление адресов. Один из этих указателей адреса используется, также, как указатель адреса для функции непрерывного просмотра таблиц. Эти 16-разрядные дополнительные регистры обозначаются X-регистр, Y-регистр и Z-регистр.

ALU поддерживает арифметические и логические операции между регистрами или между константой и регистром. Выполняются в ALU и операции с отдельными регистрами. На Рис. 4 показана AVR расширенная RISC архитектура микроконтроллеров ATmega603/103.

В дополнение к операциям с регистрами, регистровый файл может использоваться и для обычной адресации памяти. Это объясняется тем, что файл регистров располагается по 32 самыми младшими адресами пространства данных, и к ним можно обращаться как к обычным ячейкам памяти.

Пространство памяти I/O содержит 64 адреса периферийных функций CPU таких как: регистры управления, таймеры/счетчики, аналого-цифровые преобразователи и другие I/O функции. К памяти I/O можно обращаться непосредственно или как к ячейкам пространства памяти соответствующим адресам регистра файлов \$20 - \$5F. В микроконтроллерах AVR использованы принципы Гарвардской архитектуры - отдельные память и шины для программ и данных. При работе с памятью программ используется одноуровневый конвейер - в то время, как одна команда выполняется, следующая команда выбирается из памяти программ. Такой прием позволяет выполнять команду в каждом тактовом цикле. Памятью программ является внутрисистемно программируемая Flash память. За малым исключением AVR команды имеют формат одного 16-разрядного слова, в связи с чем каждый адрес памяти программ содержит одну 16-разрядную команду.

В процессе обработки прерываний и вызовов подпрограмм адрес возврата счетчика команд (PC) сохраняется в стеке. Стек размещается в SRAM данных и, следовательно размер стека ограничен только общим размером SRAM и уровнем

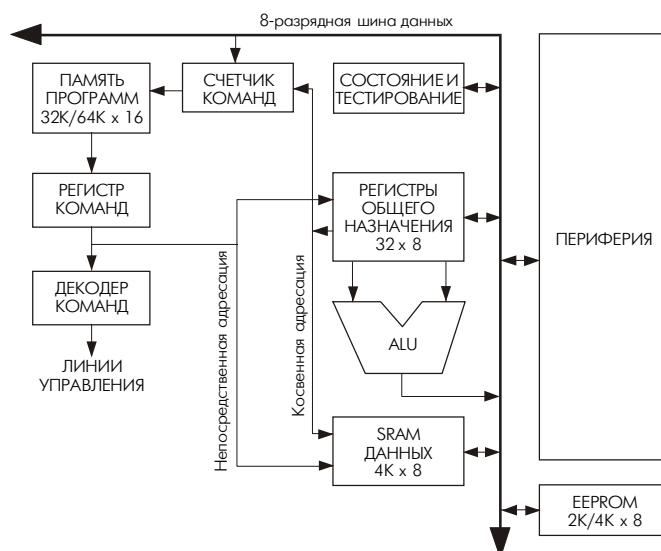


Рис. 4 AVR Enhanced RISC архитектура микроконтроллеров ATmega603/103

ее использования. Все пользовательские программы в подпрограммах возврата (прежде, чем подпрограммы или прерывания будут выполняться) должны инициализировать указатель стека (SP). 16-разрядный указатель стека, с возможностью чтения/записи располагается в пространстве I/O.

AVR архитектура поддерживает пять различных режимов адресации 4000 байт SRAM данных.

Гибкий модуль обработки прерываний имеет в пространстве I/O свой управляющий регистр с дополнительным битом разрешения глобального прерывания в регистре статуса. Все прерывания имеют свои векторы прерывания в таблице векторов прерывания, располагаемой в начале памяти программ. Приоритеты прерываний соответствуют положению векторов прерываний - прерывание с наименьшим адресом вектора имеет наивысший приоритет.

Все пространства памяти AVR архитектуры линейны и регулярны.

Файл регистров общего назначения

На Рис 5 представлена структура 32 регистров общего назначения.

Все регистровые команды обращаются непосредственно к регистрам в течение одного тактового цикла. Исключением являются пять логических и арифметических операций с константами (SBCI, SUBI, CPI и ANDI) и операция ORI между константой и содержимым регистра, и команда непосредственной загрузки константы LDI. Эти команды используют вторую половину регистров регистрового файла - R16..R31.

Самые общие команды SBC, SUB, CP, AND и OR и все прочие операции между двумя регистрами или с одним регистром используют для записи результата регистровый файл.

Как показано на Рис. 5, каждому регистру соответствует адрес памяти данных, отображающий их в первых 32 ячейках пользовательского пространства данных. Хотя

	7	0	Addr.	
	R0		\$00	
	R1		\$01	
	R2		\$02	
	...			
	R13		\$0D	
	R14		\$0E	
	R15		\$0F	
РЕГИСТРЫ ОБЩЕГО НАЗНАЧЕНИЯ	R16		\$10	
	R17		\$11	
	...			
	R26		\$1A	Младший байт регистра X
	R27		\$1B	Старший байт регистра X
	R28		\$1C	Младший байт регистра Y
	R29		\$1D	Старший байт регистра Y
	R30		\$1E	Младший байт регистра Z
	R31		\$1F	Старший байт регистра Z

Рис. 5 Регистры общего назначения CPU микроконтроллеров AVR

они не используются как физические ячейки SRAM, такая организация памяти обеспечивает гибкое обращение к регистрам, поскольку X, Y и Z регистры могут быть использованы для индексации любого регистра в файле.

SRAM данных имеет объем 4 Кбайт и занимает адресное пространство от \$0060 до \$0FFF.

РЕГИСТР X, РЕГИСТР Y И РЕГИСТР Z

Шесть регистров (с R26 по R31) регистрового файла, кроме обычной для прочих регистров функций, выполняют функцию 16-разрядных регистров указателей адреса при косвенной адресации SRAM. Эти три регистра косвенной адресации определяются как регистры X, Y и Z.

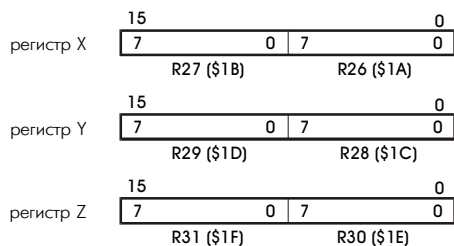


Рис. 6 Регистры X, Y и Z.

В различных режимах адресации эти регистры выполняют функции фиксированного смещения, автоматического инкремента и декремента (см. описания команд).

ALU - Арифметико-логическое устройство

Высокопроизводительное AVR ALU соединено непосредственно со всеми 32 быстродействующими регистрами общего назначения. За один тактовый цикл ALU выполняет операцию между регистрами этого регистрового файла. Операции ALU подразделяются на три основные категории: арифметические, логические и операции над битами.

Внутрисистемно программируемая Flash память программ

Коды программ микроконтроллеров ATmega603/103 записываются в 64/128 Кбайт встроенной внутрисистемно программируемой Flash памяти. Поскольку все команды имеют формат одного или двух 16-разрядных слов, то и память программ имеет организацию 32/64Кx16. Flash память обеспечивает не менее 1000 циклов стирания/записи.

Таблицы констант могут быть размещены в любом месте всего пространства памяти программ (см. описания команд LPM (Load Program Memory) - Загрузить байт памяти программ и ELPM (Extended Load Program Memory) - Загрузить байт памяти программ в расширенном режиме).

Конфигурация памяти

Микроконтроллеры ATmega603/103 поддерживают две конфигурации, как показано в таблице 2.

Таблица 2 Конфигурация памяти

Конфигурация	Встроенная SRAM данных	Внешняя SRAM данных
A	4000 байт	Нет
B	4000 байт	До 64Кбайт (1)

Примечание 1. Из 64 Кбайт внешней памяти будут доступны 60 Кбайт.

По первым 4096 адресам памяти данных размещаются регистровый файл, пространство памяти I/O и встроенная SRAM данных. Из них первые 96 адресов занимают регистровый файл и пространство памяти I/O, в следующих 4000 адресов размещается встроенная SRAM.



Рис. 7. Конфигурация памяти

Микроконтроллеры конфигурации В позволяют использовать дополнительную внешнюю память данных. Внешняя память будет адресоваться оставшимся до 64К пространством адресов, т.е. оно будет начинаться следом за пространством адресов встроенной SRAM. При использовании внешней SRAM емкостью 64К будут потеряны 4К внешней памяти, поскольку адреса этого объема будут заняты встроенной памятью. При обращении по адресам памяти данных за пределами встроенной SRAM используются те же команды, что для обращения к встроенной SRAM. При обращении к встроенной памяти данных выходы стробов управления внешней памятью данных (\overline{RD} и \overline{WR}) остаются неактивными во время всего цикла обращения.

Работа внешней SRAM разрешается установкой бита SRE в регистре MCUCR. По сравнению с обращением к встроенной памяти данных, обращение к внешней памяти данных требует дополнительного цикла на каждый байт. Это означает, что для выполнения команд LD, ST, LDS, STS, PUSH и POP требуется дополнительный тактовый цикл. Если стек размещен во внешней SRAM, то прерывания, вызов подпрограмм и возвраты потребуют два дополнительных цикла, поскольку в стеке будет опускаться и подниматься содержимое двухбайтового счетчика команд. Если интерфейс с внешней SRAM используется с состоянием ожидания, то на каждый байт необходимо еще два дополнительных тактовых цикла. Это приводит к следующему эффекту. Командам пересылки данных необходимо два дополнительных тактовых цикла, тогда как при обработке прерывания, вызове подпрограммы и при возврате из подпрограмм потребуется на четыре тактовых цикла больше, чем это указано в описании системы команд.

При адресации памяти данных используются пять режимов адресации: непосредственная адресация, косвенная со смещением, косвенная, косвенная с преддекрементом и косвенная с постдекрементом. Регистры с R26 по R31 регистрового файла работают как X, Y и Z регистры указатели косвенной адресации.

Косвенной адресации со смещением доступны 63 адреса относительно базовых адресов, находящихся в регистрах Y или Z. При использовании косвенной адресации с автоматическим преддекрементом и постдекрементом автоматически декрементируются и инкрементируются адреса записанные в регистры X, Y и Z. Всеми этими режимами перекрывается все адресное пространство данных, включая 32 регистра общего назначения и 64 регистра I/O. Подробное описание всех режимов адресации приведено в следующем разделе.

Режимы адресации памяти программ и данных

При обращении к Flash памяти программ и памяти данных (SRAM, регистровому файлу и памяти I/O) AVR Enhanced RISC микроконтроллерами ATmega603/103 используются мощные и эффективные режимы адресации. В данном разделе описываются режимы адресации, поддерживаемые AVR архитектурой. На рисунках OP обозначает часть слова команды, соответствующую операционному коду.

НЕПОСРЕДСТВЕННАЯ АДРЕСАЦИЯ, ОДИНОЧНЫЙ РЕГИСТР Rd

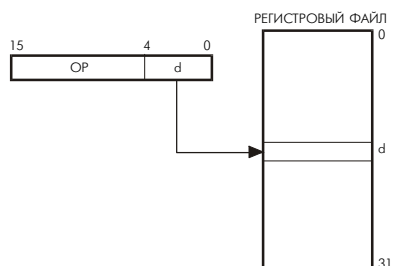


Рис.8 Непосредственная адресация одного регистра. Операнд содержится в регистре d (Rd).

НЕПОСРЕДСТВЕННАЯ АДРЕСАЦИЯ, ДВА РЕГИСТРА Rd и Rr

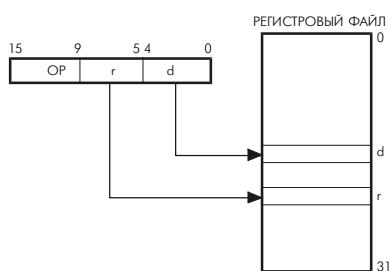


Рис.9 Непосредственная регистровая адресация двух регистров. Операнды содержатся в регистрах r (Rr) и d (Rd). Результат сохраняется в регистре d (Rd).

НЕПОСРЕДСТВЕННАЯ АДРЕСАЦИЯ I/O

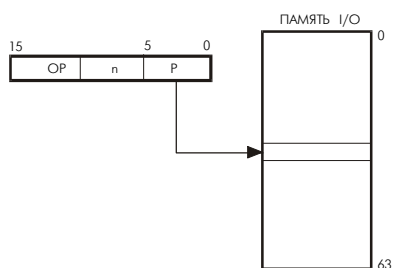


Рис. 10 Непосредственная адресация I/O. Адрес операнда содержится в 6 битах слова команды. Величина n определяет адрес регистра источника или регистра назначения.

НЕПОСРЕДСТВЕННАЯ АДРЕСАЦИЯ ДАННЫХ

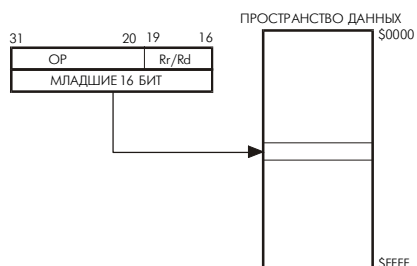


Рис. 11 Непосредственная адресация данных
 16-разрядный адрес данных содержится в 16 младших разрядах 32-разрядной команды. Rd/Rr определяют регистр источник или регистр назначения.

КОСВЕННАЯ АДРЕСАЦИЯ ДАННЫХ СО СМЕЩЕНИЕМ

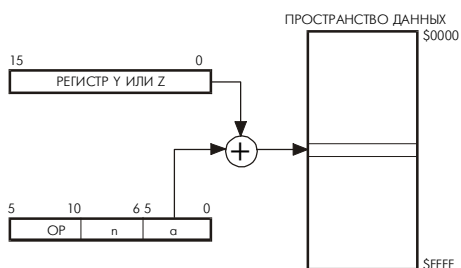


Рис. 12 Косвенная адресация данных со смещением
 Адрес операнда вычисляется суммированием содержимого регистра Y или Z с 6 битами адреса, содержащимися в слове команды.

КОСВЕННАЯ АДРЕСАЦИЯ ДАННЫХ

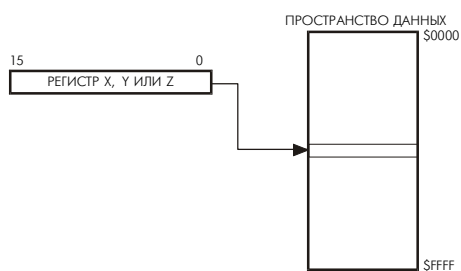


Рис. 13 Косвенная адресация данных
 Адрес операнда содержится в регистре X, Y или Z

КОСВЕННАЯ АДРЕСАЦИЯ ДАННЫХ С ПРЕДЕКРЕМЕНТОМ

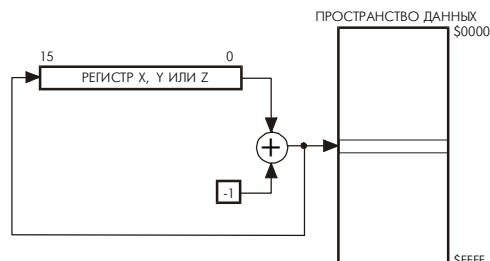


Рис. 14 Косвенная адресация данных с преддекрементом
 Перед выполнением операции регистр X, Y или Z декрементируется. Декрементированное содержимое регистра X, Y или Z является адресом операнда.

КОСВЕННАЯ АДРЕСАЦИЯ ДАННЫХ С ПОСТИНКРЕМЕНТОМ

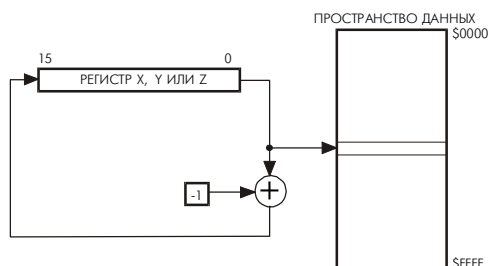


Рис. 15 Косвенная адресация данных с постинкрементом
 После выполнения операции регистр X, Y или Z инкрементируется. Адресом операнда является содержимое X, Y или Z регистра предшествовавшее инкрементированию.

АДРЕСАЦИЯ КОНСТАНТЫ С ИСПОЛЬЗОВАНИЕМ КОМАНД LPM И ELPМ

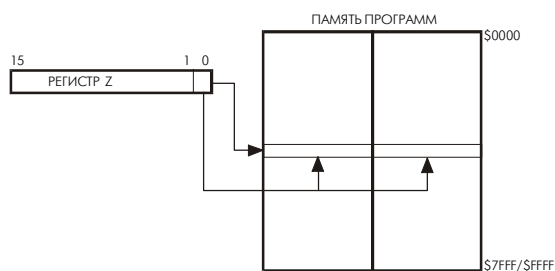


Рис. 16 Адресация константы кода памяти
 Адрес байта константы определяется содержимым регистра Z. Старшие 15 битов определяют слово адреса (от 0 до 32К). Состояние младшего бита определяет выбор младшего байта (LSB = 0) или старшего байта (LSB = 1). При использовании команды ELPМ младший бит (RAM Page) регистра Z - RAMPZ используется для выбора страницы памяти (RAMPZ0 = 0:младшая страница, RAMPZ0 = 1:старшая страница). Команда ELPМ не используется микроконтроллером ATmega603.

НЕПОСРЕДСТВЕННАЯ АДРЕСАЦИЯ ПАМЯТИ ПРОГРАММ, КОМАНДЫ JMP и CALL

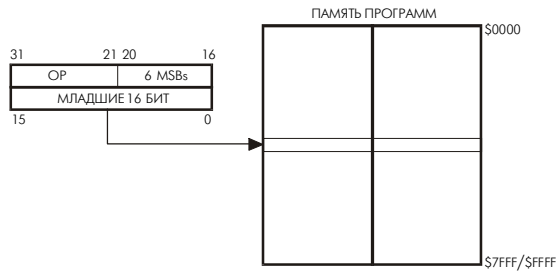


Рис. 17 Непосредственная адресация памяти программ
 Выполнение программы продолжается с адреса, записанного непосредственно в адресе команды.

КОСВЕННАЯ АДРЕСАЦИЯ ПАМЯТИ ПРОГРАММ, КОМАНДЫ IJMP и ICALL

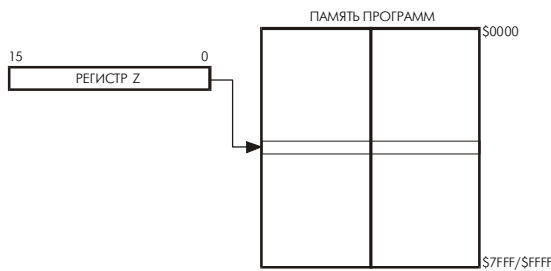


Рис. 18 Косвенная адресация памяти программ
 Выполнение программы продолжается с адреса, содержащегося в регистре Z (т.е. счетчик команд загружается содержимым регистра Z).

ОТНОСИТЕЛЬНАЯ АДРЕСАЦИЯ ПАМЯТИ ПРОГРАММ, КОМАНДЫ RJMP и RCALL

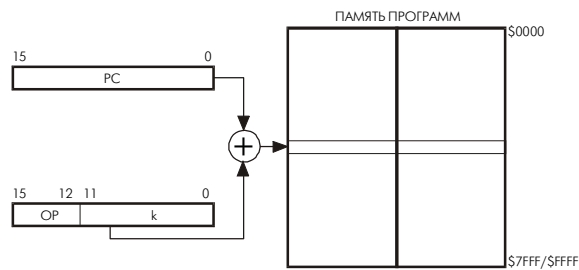


Рис. 19 Относительная адресация памяти программ
 Выполнение программы продолжается с адреса PC + k + 1. Значение относительного адреса может быть от -2048 до 2047.

EEPROM память данных

EEPROM память данных организована как отдельное пространство данных с возможностью считывания и записи отдельного байта. EEPROM обеспечивает 100000 циклов стирания/записи. Взаимодействие между EEPROM и CPU определяется регистром адреса EEPROM, регистром данных EEPROM и регистром управления EEPROM.

Время обращения к памяти и тактирование выполнения команд

В данном разделе описаны основные принципы тактирования обращений при выполнении команд и обращений к встроенной памяти.

AVR CPU тактируется системным тактовым сигналом System Clock \emptyset , формируемым посредством внешнего кварцевого кристалла. Внутреннее деление не используется.

На Рис.20 представлен процесс параллельных выборки и выполнения команд, обеспечиваемые Гарвардской архитектурой, и концепция регистрового файла быстрого доступа. Это базовый принцип конвейерной обработки, обеспечивающий удельную производительность 1 MIPS/МГц при соответствующих результатах стоимости функции, количества функций на один такт и количества функций на единицу потребляемой мощности.

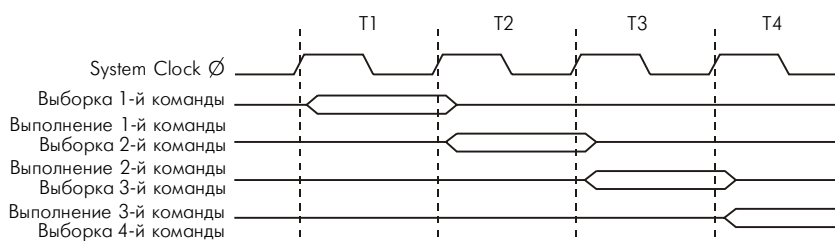


Рис. 20 Параллельные выборка и выполнение команд

На Рис. 21 представлен принцип внутреннего тактирования регистрового файла. В течение одного тактового цикла выполнения операции ALU использует два операнда регистров и результат возвращает в регистр назначения

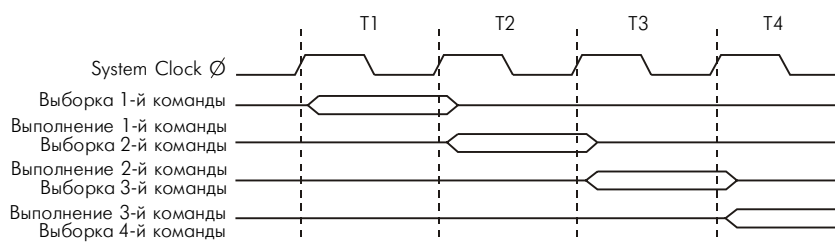


Рис.21 Одноцикловая работа ALU

На Рис. 22 показано обращение к встроенной SRAM данных за два тактовых цикла.

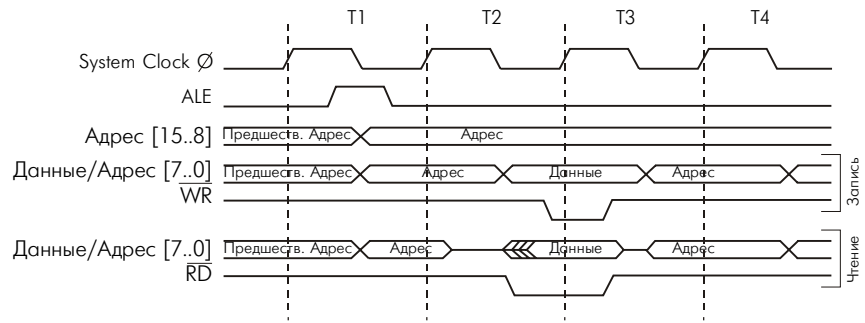


Рис. 22 Циклы обращения к внешней SRAM данных без состояния ожидания (Wait State)

На Рис. 23 показано обращение к внешней SRAM данных при установленном бите состояния ожидания (Wait State active).

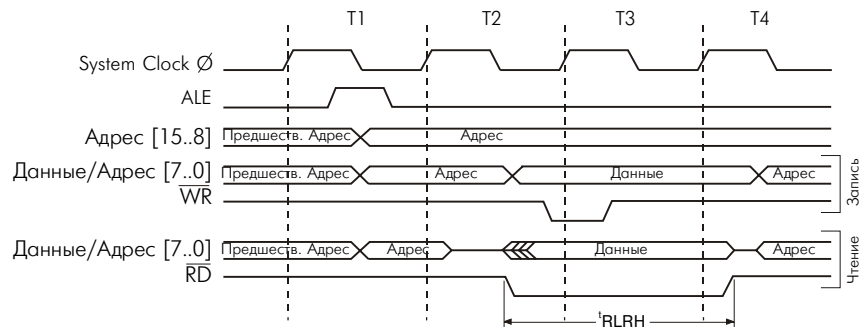


Рис. 23 Циклы обращения к внешней SRAM данных с состоянием ожидания

Память ввода/вывода (I/O)

Описание пространства I/O микроконтроллеров ATmega603/103 представлено в Табл. 3

Таблица 3. Пространство I/O микроконтроллеров ATmega603/103

Адрес I/O (адрес SRAM)	Обозначение	Функция
\$3F (\$5F)	SREG	Регистр статуса (Status REGISTER)
\$3E (\$5E)	SPH	Верхний байт указателя стека (Stack Pointer High)
\$3D (\$5D)	SPL	Нижний байт указателя стека (Stack Pointer Low)
\$3C (\$5C)	XDIV	Регистр управления делением тактовой частоты (XTAL Divide Control Register)
\$3B (\$5B)	RAMPZ	Регистр выбора страницы Z RAM (RAM Page Z Select Register)
\$3A (\$5A)	EICR	Регистр управления внешними прерываниями (External Interrupt Control Register)
\$39 (\$59)	EIMSK	Регистр масок внешних прерываний (External Interrupt MaSK register)
\$38 (\$58)	EIFR	Регистр флагов внешних прерываний (External Interrupt Flag Register)
\$37 (\$57) Counter	TIMSK	Регистр масок прерываний по таймерам/счетчикам (Timer/Interrupt MaSK register)
\$36 (\$56)	TIFR	Регистр флагов прерывания по таймерам/счетчикам (Timer/Counter Interrupt Flag register)
\$35 (\$55)	MCUCR	Регистр управления MCU (MCU General Control Register)
\$34 (\$54)	MCUSR	Регистр статуса MCU (MCU Status Register)
\$33 (\$53)	TCCR0	Регистр управления таймером/счетчиком 0 (Timer/Counter0 Control Register)
\$32 (\$52)	TCNT0	Таймер/счетчик0 (Timer/Counter0 (8-bit))
\$31 (\$51)	OCR0	Регистр сравнения выхода таймера/счетчика 0 (Timer/Counter0 Output Compare Register)
\$30 (\$50)	ASSR	Регистр статуса асинхронного режима (Asynchronous Mode Status Register)
\$2F (\$4F)	TCCR1A	Управляющий регистр А таймера/счетчика 1 (Timer/Counter1 Control Register A)
\$2E (\$4E)	TCCR1B	Управляющий регистр В таймера/счетчика 1 (Timer/Counter1 Control Register A)
\$2D (\$4D)	TCNT1H	Старший байт таймера/счетчика 1 (Timer/Counter1 High Byte)
\$2C (\$4C)	TCNT1L	Младший байт таймера/счетчика 1 (Timer/Counter1 Low Byte)
\$2B (\$4B)	OCR1AH	Старший байт регистра А сравнения выхода таймера/счетчика 1 (Timer/Counter1 Output Compare Register A High Byte)
\$2A (\$4A)	OCR1AL	Младший байт регистра А сравнения выхода таймера/счетчика 1 (Timer/Counter1 Output Compare Register A Low Byte)
\$29 (\$49)	OCR1BH	Старший байт регистра В сравнения выхода таймера/счетчика 1 (Timer/Counter1 Output Compare Register B High Byte)
\$28 (\$48)	OCR1BL	Младший байт регистра В сравнения выхода таймера/счетчика 1 (Timer/Counter1 Output Compare Register B Low Byte)
\$27 (\$47)	ICR1H	Старший байт регистра захвата таймера/счетчика 1 (Timer/Counter1 Input Capture Register High Byte)

\$26 (\$46)	ICR1L	Младший байт регистра захвата таймера/счетчика 1 (Timer/Counter1 Input Capture Register Low Byte)
\$25 (\$45)	TCCR2	Регистр управления таймером/счетчиком 2 (Timer/Counter2 Control Register)
\$24 (\$44)	TCNT2	Таймер/счетчик 2 (Timer/Counter2 (8-bit))
\$23 (\$43)	OCR2	Регистр сравнения выхода таймера/счетчика 2 (Timer/Counter2 Output Compare Register)
\$21 (\$41)	WDTCR	Регистр управления сторожевым таймером (Watchdog Timer Control Register)
\$1F (\$3F)	EEARH	Старший байт регистра адреса EEPROM (EEPROM Address Register High)
\$1E (\$3E)	EEARL	Младший байт регистра адреса EEPROM (EEPROM Address Register Low)
\$1D (\$3D)	EEDR	Регистр данных EEPROM (EEPROM Data Register)
\$1C (\$3C)	EECR	Регистр управления EEPROM (EEPROM Control Register)
\$1B (\$3B)	PORTA	Регистр данных порта A (Data Register, Port A)
\$1A (\$3A)	DDRA	Регистр направления данных порта A (Data Direction Register, Port A)
\$19 (\$39)	PINA	Выводы входов порта A (Input Pins, Port A)
\$18 (\$38)	PORTB	Регистр данных порта B (Data Register, Port B)
\$17 (\$37)	DDRB	Регистр направления данных порта B (Data Direction Register, Port B)
\$16 (\$36)	PINB	Выводы входов порта B (Input Pins, Port B)
\$15 (\$35)	PORTC	Регистр данных порта C (Data Register, Port C)
\$12 (\$32)	PORTD	Регистр данных порта D (Data Register, Port D)
\$11 (\$31)	DDRD	Регистр направления данных порта D (Data Direction Register, Port D)
\$10 (\$30)	PIND	Выводы входов порта D (Input Pins, Port D)
\$0F (\$2F)	SPDR	Регистр данных SPI I/O (SPI I/O Data Register)
\$0E (\$2E)	SPSR	Регистр статуса SPI (SPI Status Register)
\$0D (\$2D)	SPCR	Регистр управления SPI (SPI Control Register)
\$0C (\$2C)	UDR	Регистр данных UART I/O (UART I/O Data Register)
\$0B (\$2B)	USR	Регистр статуса UART (UART Status Register)
\$0A (\$2A)	UCR	Регистр управления UART (UART Control Register)
\$09 (\$29)	UBRR	Регистр управления скоростью UART (UART Baud Rate Register)
\$08 (\$28)	ACSR	Регистр статуса и управления аналогового компаратора (Analog Comparator Control and Status Register)
\$07 (\$27)	ADMUX	Регистр выбора мультиплектора ADC (ADC Multiplexer Select Register)
\$06 (\$26)	ADCSR	Регистр статуса и управления ADC (ADC Control and Status Register)
\$05 (\$25)	ADCH	Старший байт регистра данных ADC (ADC Data Register High)
\$04 (\$24)	ADCL	Младший байт регистра данных ADC (ADC Data Register Low)
\$03 (\$23)	PORTE	Регистр данных порта E (Data Register, Port E)
\$02 (\$22)	DDRE	Регистр направления данных порта E (Data Direction Register, Port E)
\$01 (\$21)	PINE	Выводы входов порта E (Input Pins, Port E)
\$00 (\$20)	PINF	Выводы входов порта F (Input Pins, Port F)

КТЦ-МК

Примечания: Зарезервированные и не используемые ячейки в таблице не показаны. Все средства I/O и периферии микроконтроллеров ATmega603/103 размещены в пространстве I/O. При использовании IN и OUT используются адреса регистров I/O с \$00 по \$3F. Поскольку регистры I/O представлены в адресном пространстве SRAM, то к ним можно адресоваться как к обычным ячейкам SRAM с адресами с \$20 по \$5F. Адрес SRAM получается простым добавлением \$20 к непосредственному адресу I/O. Адрес SRAM, по всему документу, приведен в круглых скобках после непосредственного адреса I/O. Регистры I/O, в пределах адресов от \$00 (\$20) до \$1F (\$3F), по-битово адресуются командами SBI и CBI. Состояние каждого отдельного бита этих регистров может быть проверено командами SBIS и SBIC. Более подробное описание приведено в разделе Система команд.

Различия в регистрах управления I/O и периферией подробно рассмотрены в следующем разделе.

РЕГИСТР СТАТУСА - SREG

Регистр статуса - SREG - размещен в пространстве I/O по адресу \$3F (\$5F) и его биты определяются как:

Биты	7	6	5	4	3	2	1	0	
\$3F (\$5F)	I	T	H	S	V	N	Z	C	REG
Чтение/Запись	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное состояние	0	0	0	0	0	0	0	0	

• **Bit 7 - I: Global Interrupt Enable - Разрешение глобального прерывания**

Бит разрешения глобального прерывания для разрешения прерывания должен быть установлен в состояние 1. Управление разрешением конкретного прерывания выполняется регистрами маски прерывания GIMSK и TIMSK. Если бит глобального прерывания очищен (в состоянии 0), то ни одно из разрешений конкретных прерываний, установленных в регистрах GIMSK и TIMSK, не действует. Бит I аппаратно очищается после прерывания и устанавливается для последующего разрешения глобального прерывания командой RETI.

• **Bit 6 - T: Bit Copy Storage - Бит сохранения копии**

Команды копирования бита BLD (Bit Load) и BST (Bit Store) используют бит T как бит источник и бит назначения при операциях с битами. Командой BST бит регистра регистрового файла копируется в бит T, командой BLD бит T копируется в регистр регистрового файла.

• **Bit 5 - H: Half Carry Flag - Флаг полупереноса**

Флаг полупереноса указывает на полуперенос в ряде арифметических операций. Более подробная информация приведена в описании системы команд.

• **Bit 4 - S: Sign Bit, S = N ⊕ V - Бит знака**

Бит S всегда находится в состоянии, определяемом логическим исключающим ИЛИ (exclusive OR) между флагом отрицательного значения N и дополнением до двух флага переполнения V. Более подробная информация приведена в описании системы команд.

• **Bit 3 - V: Two's Complement Overflow Flag - Дополнение до двух флага переполнения**

Дополнение до двух флага V поддерживает арифметику дополнения до двух. Более подробная информация приведена в описании системы команд.

• **Bit 2 - N: Negative Flag - Флаг отрицательного значения**

Флаг отрицательного значения N указывает на отрицательный результат ряда

арифметических и логических операций. Более подробная информация приведена в описании системы команд.

• **Bit 1 - Z: Zero Flag - Флаг нулевого значения**

Флаг нулевого значения Z указывает на нулевой результат ряда арифметических и логических операций. Более подробная информация приведена в описании системы команд.

• **Bit 0 - C: Carry Flag - Флаг переноса**

Флаг переноса C указывает на перенос в арифметических и логических операциях. Более подробная информация приведена в описании системы команд.

УКАЗАТЕЛЬ СТЕКА - Stack Pointer - SP

Микроконтроллеры AVR оснащены 16-разрядным указателем стека, размещенным в двух регистрах пространства I/O по адресам \$3E (\$5E) и \$3D (\$5D). Поскольку микроконтроллеры ATmega603/103 поддерживают объем SRAM до 64 Кбайт, то используются все 16 разрядов указателя стека.

Указатель стека указывает на область в SRAM данных, в которой размещаются стеки подпрограмм и прерываний. Объем стека в SRAM данных должен задаваться программой перед каждым вызовом подпрограммы и обработкой разрешенного прерывания. Указатель стека декрементируется на единицу, при каждом занесении командой PUSH данных в стек, и на две единицы при занесении данных в стек подпрограммой CALL и прерыванием.

Указатель стека инкрементируется на единицу, при извлечении данных из стека командой POP, и на две единицы при извлечении данных из стека при возврате из подпрограммы (RET) или возврате из прерывания (IRET).

Биты	15	14	13	12	11	10	9	8	
\$3E (\$5E)	SP15	SP14	SP13	SP12	SP11	SP10	SP9	SP8	SPH
\$3D (\$5D)	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0	SPL
	7	6	5	4	3	2	1	0	
Чтение/Запись	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное состояние	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	

РЕГИСТР ВЫБОРА СТРАНИЦЫ Z RAM -

The RAM Page Z Select Register - RAMPZ

Биты	7	6	5	4	3	2	1	0	
\$3B (\$5B)	-	-	-	-	-	-	-	RAMPZ0	RAMPZ
Чтение/Запись	R	R	R	R	R	R	R	R/W	
Начальное состояние	0	0	0	0	0	0	0	0	

Регистр RAMPZ используется обычно для определения к какой странице RAM, емкостью 64К, возможно обращение посредством указателя Z. Поскольку микроконтроллеры ATmega603/103 не поддерживают SRAM с объемом свыше 64К, этот регистр используется только для выбора страницы в памяти программ при использовании команды ELPM. Различные установки бита RAMPZ0 оказывают следующий эффект:

КТЦ-МК

RAMPZ0 = 0: Команде ELPM доступна память программ с адресами от \$0000 до \$7FFF (младшие 64 Кбайт)

RAMPZ0 = 1: Команде ELPM доступна память программ с адресами от \$8000 до \$FFFF (старшие 64 Кбайт).

Отметим, что на LPM не воздействует установка RAMPZ.

Микроконтроллер АТмега603 не содержит регистра RAMPZ и не имеет команды ELPM. Команда LPM способна перекрыть все пространство памяти программ микроконтроллера АТмега603.

РЕГИСТР УПРАВЛЕНИЯ MCU - MCU Control Register - MCUCR

Биты регистра управления MCU управляют выполнением основных функций MCU.

Биты	7	6	5	4	3	2	1	0	
\$3B (\$5B)	SRE	SRW	SE	SM1	SM0	-	-	-	MCUCR
Чтение/Запись	R/W	R/W	R/W	R/W	R/W	R	R	R	
Начальное состояние	0	0	0	0	0	0	0	0	

• Bit 7 - SRE: External SRAM Enable - Разрешение внешней SRAM

Установленный в 1 бит SRE разрешает обращение к внешней SRAM данных и переводит работу выводов AD0-7 (Порт A), A8-15 (Порт C), \overline{WR} и \overline{RD} на выполнение альтернативной функции. Затем бит SRE перенастраивает установки направлений любых выводов в соответствующих регистрах направления данных. Очистка бита SRE (установка в 0) запрещает обращение к внешней SRAM и восстанавливает нормальные установки направлений выводов и данных.

• Bit 6 - SRW: External SRAM Wait State - Режим ожидания внешней SRAM

При установленном в 1 бите SRW к циклу обращения к внешней SRAM добавляется один цикл ожидания. При сброшенном в 0 бите SRW обращение к внешней SRAM выполняется по трехцикловой схеме. См. Рис. 22 Циклы обращения к внешней SRAM данных без состояния ожидания (Wait State) и Рис. 23 Циклы обращения к внешней SRAM данных с состоянием ожидания (Wait State active).

• Bit 5 - SE: Sleep Enable - Разрешение режима Sleep

Установленный в 1 бит SE разрешает перевод MCU в режим sleep по команде SLEEP. Чтобы исключить перевод MCU в незапрограммированный режим sleep, рекомендуется устанавливать бит SE непосредственно перед выполнением команды SLEEP.

• Bits 4,3 - SM1/SM0: Sleep Mode Select bits 1 and 0 - Биты выбора режима Sleep

Данные биты позволяют выбрать один из трех возможных режимов sleep, как показано в таблице 4.

Таблица 4 Выбор режима Sleep

SM1	SM0	Sleep Mode
0	0	Режим Idle
0	1	Зарезервировано
1	0	Режим Power Down
1	1	Режим Power Save

Bits 2..0 - Res: Reserved bits - Зарезервированные биты

Эти биты зарезервированы и при считывании всегда будут показывать состояние 0.

РЕГИСТР УПРАВЛЕНИЯ КОЭФФИЦИЕНТОМ ДЕЛЕНИЯ ЧАСТОТЫ КВАРЦЕВОГО ГЕНЕРАТОРА - XTAL Divide Control Register - XDIV

Регистр XDIV используется для установления коэффициента деления частоты кварцевого генератора в диапазоне от 1 до 129.

Биты	7	6	5	4	3	2	1	0	
\$3C (\$5C)	XDIVEN	XDIV6	XDIV5	XDIV4	XDIV3	XDIV2	XDIV1	XDIV0	XDIV
Чтение/Запись	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное состояние	0	0	0	0	0	0	0	0	

• Bit 7 - XDIVEN: XTAL Divide Enable - Разрешение деления частоты XTAL

При установленном в 1 бите XDIVEN тактовая частота CPU и всей периферии делится в соответствии с установленными битами XDIV6 - XDIV0 коэффициентом деления. Такая возможность может быть использована для снижения потребления, при условии допустимости снижения вычислительной мощности. Этот бит может быть установлен и очищен в процессе выполнения программы тогда, когда это определяется применением.

• Bits 6..0 - XDIV6..XDIV0: XTAL Divide Select Bits 6 - 0 - Биты выбора коэффициента деления

Эти биты устанавливают коэффициент деления тактовой частоты при установленном бите XDIVEN. Если десятичное значение этих семи битов обозначить через d, то результирующая тактовая частота CPU будет определяться по формуле

$$f_{clk} = \frac{XTAL}{(129-d)}$$

Состояния этих битов можно изменить только когда бит XDIVEN сброшен (в состоянии 0). При установленном бите XDIVEN, записанное одновременно в биты XDIV6..XDIV0 значение будет определять коэффициент деления. При сбросе бита XDIVEN записанные в биты XDIV6..XDIV0 значения игнорируются. Поскольку делитель делит тактовую частоту поступающую на MCU, то и на периферийные устройства поступает тактовая частота с тем же коэффициентом деления.

Обработка прерываний и сброса

Микроконтроллеры ATmega603/103 используют 23 источника прерывания. Эти прерывания и вектор сброса располагают отдельными программными векторами в пространстве памяти программ. Каждому прерыванию присвоен свой бит разрешения который должен быть установлен совместно с битом I регистра статуса. Младшие адреса пространства памяти программ автоматически определяются как векторы сброса и прерываний.

Полный перечень векторов представлен в Таблице 5. Перечень представляет также уровень приоритета для каждого прерывания. Прерывания с младшими адресами имеют больший уровень приоритета. RESET имеет наивысший уровень приоритета, следующим является INTO - Запрос внешнего прерывания 0 и т.д.

Таблица 5 Векторы сброса и прерываний

Vector No.	Program Address	Source	Interrupt Definition
1	\$0000	RESET	Сброс по выводу и сторожевому таймеру (Hardware Pin and Watchdog Reset)
2	\$0002	INT0	Запрос внешнего прерывания 0 (External Interrupt Request 0)
3	\$0004	INT1	Запрос внешнего прерывания 1 (External Interrupt Request 1)
4	\$0006	INT2	Запрос внешнего прерывания 2 (External Interrupt Request 2)
5	\$0008	INT3	Запрос внешнего прерывания 3 (External Interrupt Request 3)
6	\$000A	INT4	Запрос внешнего прерывания 4 (External Interrupt Request 4)
7	\$000C	INT5	Запрос внешнего прерывания 5 (External Interrupt Request 5)
8	\$000E	INT6	Запрос внешнего прерывания 6 (External Interrupt Request 6)
9	\$0010	INT7	Запрос внешнего прерывания 7 (External Interrupt Request 7)
10	\$0012	TIMER2 COMP	Совпадение при сравнении таймера/счетчика 2 (Timer/Counter2 Compare Match)
11	\$0014	TIMER2 OVF	Переполнение таймера/счетчика 2 (Timer/Counter2 Overflow)
12	\$0016	TIMER1 CAPT	Захват таймера/счетчика 1 (Timer/Counter1 Capture Event)
13	\$0018	TIMER1 COMPA	Совпадение А при сравнении таймера/счетчика 1 (Timer/Counter1 Compare Match A)
14	\$001A	TIMER1 COMPB	Совпадение В при сравнении таймера/счетчика 1 (Timer/Counter1 Compare Match B)
15	\$001C	TIMER1 OVF	Переполнение таймера/счетчика 1 (Timer/Counter1 Overflow)
16	\$001E	TIMER0 COMP	Совпадение при сравнении таймера/счетчика 0 (Timer/Counter0 Compare Match)
17	\$0020	TIMER0 OVF	Переполнение таймера/счетчика 0 (Timer/Counter0 Overflow)
18	\$0022	SPI, STC	Завершение пересылки SPI (SPI Serial Transfer Complete)
19	\$0024	UART, RX	Завершение приема UART (UART, Rx Complete)
20	\$0026	UART, UDRE	Регистр данных UART пуст (UART Data Register Empty)
21	\$0028	UART, TX	Завершение передачи UART (UART, Tx Complete)
22	\$002A	ADC	Завершение ADC преобразования (ADC Conversion Complete)
23	\$002C	EE READY	Готовность EEPROM (EEPROM Ready)
24	\$002E	ANALOG COMP	Срабатывание аналогового компаратора (Analog Comparator)

Наиболее часто используемые программные установки адресов векторов сброса и прерываний:

Адрес	Метка	Код	Комментарий
\$0000		jmp RESET	; Обработчик сброса (Reset Handler)
\$0002		jmp EXT_INT0	; Обработчик прерывания IRQ0 (IRQ0Handler)
\$0004		jmp EXT_INT1	; Обработчик прерывания IRQ1 (IRQ1 Handler)
\$0006		jmp EXT_INT2	; Обработчик прерывания IRQ2 (IRQ2 Handler)
\$0008		jmp EXT_INT3	; Обработчик прерывания IRQ3 (IRQ3 Handler)
\$000A		jmp EXT_INT4	; Обработчик прерывания IRQ4 (IRQ4 Handler)
\$000C		jmp EXT_INT5	; Обработчик прерывания IRQ5 (IRQ5 Handler)
\$000E		jmp EXT_INT6	; Обработчик прерывания IRQ6 (IRQ6 Handler)
\$0010		jmp EXT_INT7	; Обработчик прерывания IRQ7 (IRQ7 Handler)
\$0012		jmp TIM2_COMP	; Обработчик сравнения таймера 2 (Timer2 Compare Handler)
\$0014		jmp TIM2_OVF	; Обработчик переполнения таймера 2 (Timer2 Overflow Handler)
\$0016		jmp TIM1_CAPT	; Обработчик захвата таймера 1 (Timer1 Capture Handler)
\$0018		jmp TIM1_COMP_A	; Обработчик сравнения А таймера 1 (Timer1 CompareA Handler)
\$001A		jmp TIM1_COMP_B	; Обработчик сравнения В таймера 1 (Timer1 CompareB Handler)
\$001C		jmp TIM1_OVF	; Обработчик переполнения таймера 1 (Timer1 Overflow Handler)
\$001E		jmp TIM0_COMP	; Обработчик сравнения таймера 0 (Timer0 Compare Handler)
\$0020		jmp TIM0_OVF	; Обработчик переполнения таймера 0 (Timer0 Overflow Handler)
\$0022		jmp SPI_STC	; Обработчик завершения пересылки SPI (SPI Transfer Complete Handler)
\$0024		jmp UART_RXC	; Обработчик завершения приема UART (UART RX Complete Handler)
\$0026		jmp UART_DRE	; Обработчик пустого регистра данных (UDR Empty Handler)
\$0028		jmp UART_TXC	; Обработчик завершения передачи UART (UART TX Complete Handler)
\$002A		jmp ADC	; Обработчик завершения преобразования ADC (ADC Conversion Complete Handler)
\$002C		jmp EE_RDY	; Обработчик готовности EEPROM (EEPROM Ready Handler)
\$002E		jmp ANA_COMP	; Обработчик срабатывания аналогового компаратора (Analog Comparator Handler)
\$0030	MAIN:	<instr> xxx	; Начало основной программы (Main program start)
...

ИСТОЧНИКИ СБРОСА

Микроконтроллеры ATmega603/103 располагают тремя источниками сигнала сброса:

- Сброс по включению питания (Power-On Reset). MCU сбрасывается при подключении питания к выводам VCC и GND.
- Внешний сброс (External Reset). MCU сбрасывается если низкий уровень присутствует на входе более двух циклов XTAL.
- Сброс по сторожевому таймеру (Watchdog Reset). MCU сбрасывается если истекает период сторожевого таймера и сторожевой таймер разрешен.

В течение сброса все регистры I/O, за исключением регистра статуса MCU, устанавливаются в их начальные состояния и программа начинает работу с адреса \$0000. По этому адресу должна находиться команда JMP - команда абсолютного перехода к подпрограмме обработки сброса. Если программа никогда не разрешает прерывания, то векторы прерываний не используются и по этим адресам могут располагаться коды программы.

Таблица 6. Временные и электрические параметры схемы сброса (V_{CC} = 5 В).

Обозн.	Параметр	Условия	Мин	Тип	Макс	Ед. измерения
V _{POT}	Граничное напряжение сброса по включению питания		1,8	2	2,2	V
V _{RST}	Граничное напряжение сброса по выводу RESET			V _{CC} /2		V
V _{BO}	Сброс по снижению напряж. питания			2.5		V
T _{TOUT}	Период задержки сигнала сброса	SUT1/0 = 00		5		Циклов CPU
		SUT1/0 = 01	0.4	0.5	0.6	мс
		SUT1/0 = 10	3.2	4.0	48	мс
		SUT1/0 = 11	128	160	192	мс

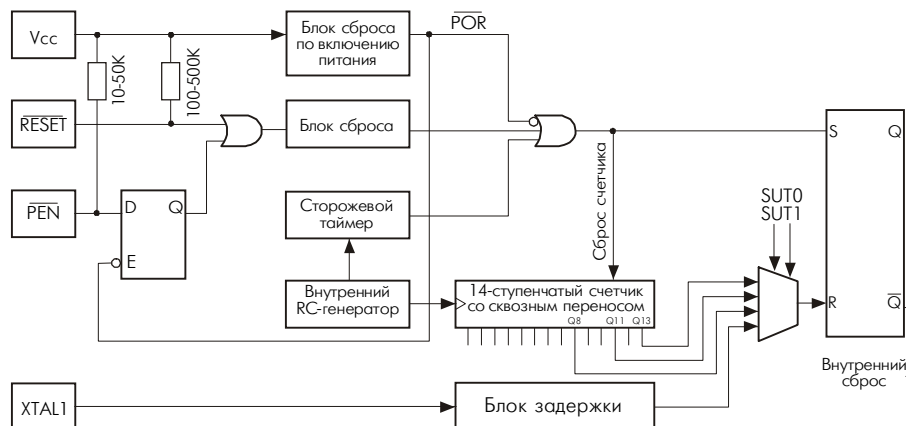


Рис. 24 Логика сброса

СБРОС ПО ВКЛЮЧЕНИЮ ПИТАНИЯ

Схема сброса по включению питания (Power-On Reset - POR) обеспечивает запуск микроконтроллера только по достижении напряжением V_{cc} безопасного уровня. Как показано на Рис.24, встроенный таймер, тактируемый встроенным генератором сторожевого таймера, удерживает запуск MCU на некоторое время после достижения граничного напряжения включения питания V_{pot} , не зависящее от скорости нарастания напряжения V_{cc} (см. Рис. 26).

В Таблице 6 показаны установки битов SUT1 и SUT0 использующихся для установки длительности периода задержки процедуры запуска. Пользователю предоставляется возможность выбора задержки времени запуска. Установка SUT 1/0 = 00, при которой MCU запускается через 5 тактовых циклов, используется при использовании внешнего тактового сигнала, подаваемого на вывод XTAL1. Такая установка обеспечивает быстрый запуск из режимов power down или power save, при условии наличия тактового сигнала в этих режимах. Подробности в разделе *Программирование*.

Если встроенная задержка запуска достаточна, то \overline{RESET} может быть подсоединен к V_{cc} непосредственно или через внешний нагрузочный резистор. Удержанием вывода на низком уровне, во время подачи напряжения, период сброса по включению питания может быть увеличен. Пример такого тактирования приведен на. Рис. 27.

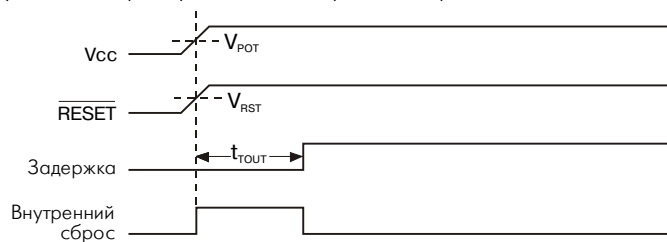


Рис. 25 Начальный запуск MCU. Вывод \overline{RESET} подключен к V_{cc} , быстрое нарастание V_{cc}

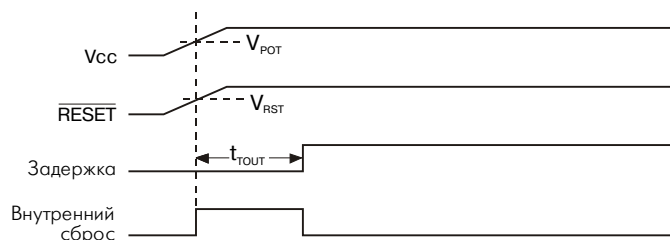


Рис.26 Начальный запуск MCU. Вывод $\overline{\text{RESET}}$ подключен к Vcc, медленное нарастание Vcc

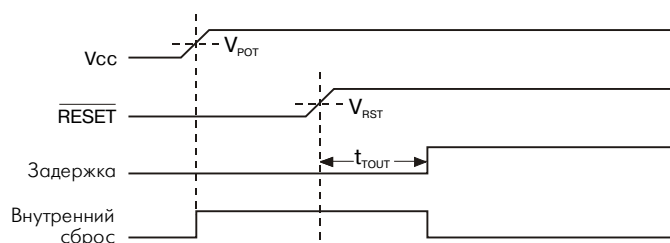


Рис. 27 Начальный запуск MCU. Внешнее управление состоянием вывода $\overline{\text{RESET}}$

ВНЕШНЕЕ УПРАВЛЕНИЕ СБРОСОМ

Внешний сброс формируется подачей низкого уровня на вывод RESET на время не меньше двух тактовых циклов кварцевого генератора. При достижении напряжением на выводе RESET уровня VRST запускается таймер, задерживающий запуск MCU на время t_{TOUT}.

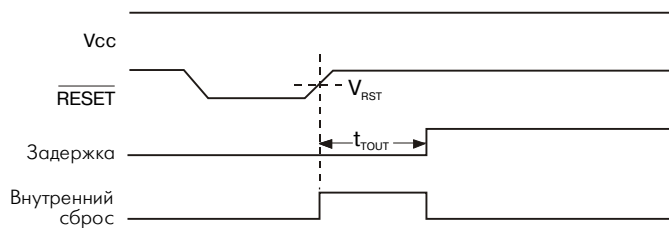


Рис. 28 Внешний сброс во время работы микроконтроллера

СБРОС ПО СТОРОЖЕВОМУ ТАЙМЕРУ

По окончании времени, определяемого сторожевым таймером, таймер формирует короткий, длительностью в один цикл XTAL, импульс сброса. По падающему фронту этого импульса таймер задержки начинает отсчет t_{TOUT}. В разделе *Сторожевой таймер (Watchdog Timer)* приводится подробное описание работы сторожевого таймера.

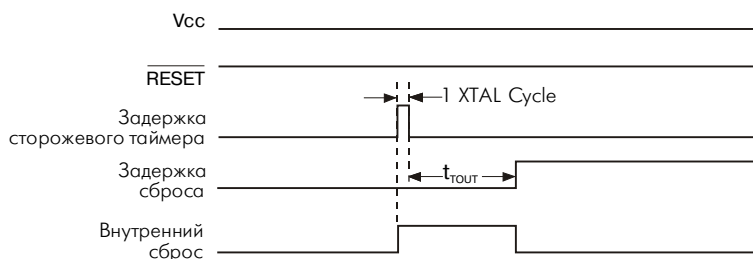


Рис. 29 Сброс по сторожевому таймеру во время работы микроконтроллера

РЕГИСТР СТАТУСА MCU - MCUSR - (MCU Status Register)

Регистр статуса MCU выдает информацию об источнике, вызвавшем сброс MCU.

Бит	7	6	5	4	3	2	1	0	
\$34(\$54)	-	-	-	-	-	-	EXTRF	PORF	MCUSR
Чтение/Запись	R	R	R	R	R	R	R/W	R/W	
Начальное значение	0	0	0	0	0	0	см. описание битов		

• **Bits 7..2 - Res: Reserved Bits - Зарезервированные биты**

Эти биты зарезервированы и при считывании всегда покажут состояние 0.

• **Bit 1 - EXTRF: External Reset Flag - Флаг внешнего сброса**

После подачи напряжения питания состояние этого бита неопределенно (X). Бит устанавливается в состояние 1 внешним сбросом. Сброс по сторожевому таймеру оставляет этот бит неизменным.

• **Bit 0 - PORF: Power On Reset Flag - Флаг включения питания**

Данный бит устанавливается сбросом по подаче питания. Сброс по сторожевому таймеру или по внешнему сбросу оставляют этот бит неизменным.

В таблице показано влияние трех режимов сброса на состояния битов PORF и EXTRF.

Таблица 7 Состояния PORF и EXTRF битов после сброса

Источник сброса	PORF	EXTRF
Сброс по подаче питания	1	Неопределенное состояние
Внешний сброс	Не меняется	1
Сброс по сторожевому таймеру	Не меняется	Не меняется

Для определения режима сброса, на основании состояния данных битов PORF и EXTRF, пользовательское программное обеспечение должно сбрасывать биты PORF и EXTRF как только это позволит программа. Прежде, чем их очистить необходимо провести проверку этих битов. Если бит был очищен до внешнего сброса или сброса по сторожевому таймеру, то источник сброса может быть определен по следующей таблице истинности:

Таблица 8 Определение источника сброса

PORF	EXTRF	Источник сброса
0	0	Сброс по сторожевому таймеру
0	1	Внешний сброс
1	0	Сброс по подаче питания
1	1	Сброс по подаче питания

ОБРАБОТКА ПРЕРЫВАНИЙ

Микроконтроллеры ATmega603/103 содержат два специальных 8-разрядных регистра масок прерываний: регистр масок внешних прерываний EIMSK (External Interrupt Mask) и регистр масок прерываний по таймеру/счетчику TIMSK (Timer/Counter Interrupt Mask). Кроме того, в регистрах управления периферией могут быть организованы и другие биты разрешения и биты масок.

При возникновении прерывания бит I разрешения глобального прерывания (Global Interrupt Enable) очищается и все прочие прерывания запрещаются. Пользовательское ПО, с тем, чтобы разрешить вложенные прерывания, может установить бит I внутри подпрограммы обработки прерывания. Выход из подпрограммы обработки прерывания происходит по команде RETI, при этом бит I устанавливается в состояние 1. Когда счетчик команд указывает вектор подпрограммы обработки прерывания, соответствующий флаг, вызвавший прерывание, аппаратно очищается. Некоторые флаги прерываний можно очистить, записав в соответствующий бит(ы) очищаемого флага логическую единицу.

РЕГИСТР МАСОК ВНЕШНИХ ПРЕРЫВАНИЙ - EIMSK - (External Interrupt Mask Register)

Бит	7	6	5	4	3	2	1	0	
\$39 (\$59)	INT7	INT6	INT5	INT4	INT3	INT2	INT1	INT0	EIMSK
Чтение/Запись R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное значение	0	0	0	0	0	0	0	0	

• **Bits 7..4 - INT7 - INT4: External Interrupt Request 7-4 Enable - Разрешение запроса внешних прерываний с 4 по 7**

При установленных битах INT7- INT4 и установленном бите I регистра статуса (SREG) разрешаются прерывания по соответствующим выводам входов сигналов прерываний. Бит управления опознаванием прерывания регистра управления внешними прерываниями EICR (External Interrupt Control Register) определяет срабатывание по нарастающему или падающему фронту или по логическому уровню. Активация любого из этих выводов вызовет запрос прерывания даже если вывод будет разрешен как выход. Это обеспечивает возможность организации программного прерывания.

• **Bits 3..0 - INT3 - INT0: External Interrupt Request 3-0 Enable - Разрешение запроса внешних прерываний с 0 по 3**

При установленных битах INT3- INT0 и установленном бите I регистра статуса (SREG) разрешаются прерывания по соответствующим входам прерываний. Внешние прерывания всегда вызывают прерывание низким уровнем. Активация любого из этих выводов вызовет запрос прерывания даже если вывод будет разрешен как выход. Это обеспечивает возможность организации программного прерывания. Запрос прерывания

КТЦ-МК

по логическому уровню, если он разрешен, будет генерировать запрос прерывания до тех пор, пока на входе будет находиться низкий уровень.

РЕГИСТР ФЛАГОВ ВНЕШНИХ ПЕРЕРЫВАНИЙ - EIFR (External Interrupt Flag Register)

Бит	7	6	5	4	3	2	1	0	
\$38 (\$58)	INTF7	INTF6	INTF5	INTF4	-	-	-	-	EIFR
Чтение/Запись	R/W	R/W	R/W	R/W	R	R	R	R	
Начальное значение	0	0	0	0	0	0	0	0	

• **Bits 7..4 - INTF7 - INTF4: External Interrupt 7-4 Flags - Флаги внешних прерываний с 4 по 7**

В случае поступления запроса на прерывание на какой либо из выводов INT7 - INT4, будет установлен в 1 соответствующий флаг прерывания (INTF7 - INTF4). Если бит I регистра SREG и соответствующий бит разрешения (INT7 - INT4) в EIMSK будут установлены, то MCU перейдет к вектору прерывания. По завершению подпрограммы прерывания флаг очищается. Кроме того, его можно очистить, записав в него логическую 1.

• **Bits 3..0 - Res: Reserved Bits - Зарезервированные биты**

Эти биты зарезервированы и при считывании всегда покажут состояние 0.

РЕГИСТР УПРАВЛЕНИЯ ВНЕШНИМИ ПЕРЕРЫВАНИЯМИ - EICR (External Interrupt Control Register)

Бит	7	6	5	4	3	2	1	0	
\$3A (\$5A)	ISC71	ISC70	ISC61	ISC60	ISC51	ISC50	ISC41	ISC40	EICR
Чтение/Запись	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное значение	0	0	0	0	0	0	0	0	

• **Bits 7..0 - ISCX1, ISCX0: External Interrupt 7-4 Sense Control bits - Биты управления опознаванием внешних прерываний с 4 по 7**

Внешние прерывания 7 - 4 активируются по выводам INT7 - INT4, если установлен флаг I в SREG и установлена соответствующая маска в EIMSK. Запрос прерывания по логическому уровню или фронтам определяется в следующей таблице:

Таблица 9 Управление опознаванием прерывания

ISCX1	ISCX0	Описание
0	0	Запрос прерывания генерируется низким уровнем на INTX.
0	1	Зарезервирован
1	0	Запрос прерывания генерируется падающим фронтом на INTX.
1	1	Запрос прерывания генерируется нарастающим фронтом на INTX.

Примечание: X может быть равен 7, 6, 5 или 4.

При изменении битов ISCX1/ISCX0 прерывание должно быть запрещено путем очистки бита разрешения в регистре GIMSK. В ином случае может произойти прерывание. Запрос прерывания по логическому уровню, если он разрешен, будет генерировать запрос прерывания до тех пор, пока на входе будет находиться низкий уровень.

**РЕГИСТР МАСОК ПЕРЕРЫВАНИЯ ПО ТАЙМЕРАМ/СЧЕТЧИКАМ-
TIMSK (Timer/Counter Interrupt Mask Register)**

Бит	7	6	5	4	3	2	1	0	
\$37 (\$57)	OCIE2	TOIE2	TICIE1	OCIE1A	OCIE1B	TOIE1	OCIE0	TOIE0	TIMSK
Чтение/Запись	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное значение	0	0	0	0	0	0	0	0	

• Bit 7 - OCIE2: Timer/Counter2 Output Compare Interrupt Enable - Разрешение прерывания по совпадению таймера/счетчика2

При установленном бите OCIE2 и установленном бите I регистра статуса разрешается прерывание по совпадению содержимого регистра сравнения и состояния таймера/счетчика2. Соответствующее прерывание (с вектором \$0012) выполняется если произойдет совпадение при сравнении содержимого регистра сравнения и состояния таймера/счетчика2. В регистре флагов прерывания TIFR (Timer/Counter Interrupt Flag Register) устанавливается флаг совпадения таймера/счетчика2.

• Bit 6 - TOIE2: Timer/Counter2 Overflow Interrupt Enable - Разрешение прерывания по переполнению таймера/счетчика2

При установленном бите TOIE2 и установленном бите I регистра статуса разрешается прерывание по переполнению таймера/счетчика2. Соответствующее прерывание (с вектором \$0014) выполняется если произойдет переполнение таймера/счетчика2. В регистре флагов прерывания TIFR (Timer/Counter Interrupt Flag Register) устанавливается флаг переполнения таймера/счетчика2.

• Bit 5 - TICIE1: Timer/Counter1 Input Capture Interrupt Enable - Разрешение прерывания по захвату таймера/счетчика1

При установленном бите TICIE1 и установленном бите I регистра статуса разрешается прерывание по захвату таймера/счетчика1. Соответствующее прерывание (с вектором \$0016) выполняется если произойдет запуск захвата по выводу 29, PD4(IC1). В регистре флагов прерывания TIFR (Timer/Counter Interrupt Flag Register) устанавливается флаг захвата таймера/счетчика1.

• Bit 4 - OCIE1A: Timer/Counter1 Output CompareA Match Interrupt Enable - Разрешение прерывания по совпадению регистра A с таймером/счетчиком1

При установленном бите OCIE1A и установленном бите I регистра статуса разрешается прерывание по совпадению регистра A с состоянием таймера/счетчика1. Соответствующее прерывание (с вектором \$0018) выполняется если произойдет совпадение содержимого регистра A сравнения выхода с состоянием таймера/счетчика1. В регистре флагов прерывания TIFR (Timer/Counter Interrupt Flag Register) устанавливается флаг совпадения регистра A с таймером/счетчиком1.

• Bit 3 - OCIE1B: Timer/Counter1 Output CompareB Match Interrupt Enable - Разрешение прерывания по совпадению регистра B с таймером/счетчиком1

При установленном бите OCIE1B и установленном бите I регистра статуса разрешается прерывание по совпадению регистра B с состоянием таймера/счетчика1. Соответствующее прерывание (с вектором \$001A) выполняется если произойдет совпадение содержимого регистра B сравнения выхода с состоянием таймера/счетчика1. В регистре флагов прерывания TIFR (Timer/Counter Interrupt Flag Register) устанавливается флаг совпадения регистра B с таймером/счетчиком1.

• Bit 2 - TOIE1: Timer/Counter1 Overflow Interrupt Enable - Разрешение прерывания по переполнению таймера/счетчика1

При установленном бите OCIE1B и установленном бите I регистра статуса разрешается

КТЦ-МК

прерывание по переполнению таймера/счетчика1. Соответствующее прерывание (с вектором \$001С) выполняется если произойдет переполнение таймера/счетчика1. В регистре флагов TIFR (Timer/Counter Interrupt Flag Register) устанавливается флаг переполнения таймера/счетчика1. При нахождении таймера/счетчика1 в PWM режиме флаг переполнения счетчика устанавливается когда счетчик изменит направление счета при \$0000.

• **Bit 1 - OCIE0: Timer/Counter0 Output Compare Interrupt Enable - Разрешение прерывания по совпадению таймера/счетчика0**

При установленном бите OCIE0 и установленном бите I регистра статуса разрешается прерывание по совпадению содержимого регистра сравнения и состояния таймера/счетчика0. Соответствующее прерывание (с вектором \$001E) выполняется если произойдет совпадение при сравнении содержимого регистра сравнения и состояния таймера/счетчика0. В регистре флагов прерывания TIFR (Timer/Counter Interrupt Flag Register) устанавливается флаг совпадения таймера/счетчика0.

• **Bit 0 - TOIE0: Timer/Counter0 Overflow Interrupt Enable - Разрешение прерывания по переполнению таймера/счетчика0**

При установленном бите TOIE0 и установленном бите I регистра статуса разрешается прерывание по переполнению таймера/счетчика0. Соответствующее прерывание (с вектором \$0020) выполняется если произойдет переполнение таймера/счетчика0. В регистре флагов TIFR (Timer/Counter Interrupt Flag Register) устанавливается флаг переполнения таймера/счетчика0.

РЕГИСТР ФЛАГОВ ПРЕРЫВАНИЙ ПО ТАЙМЕРАМ/СЧЕТЧИКАМ TIFR (Timer/Counter Interrupt Flag Register)

Бит	7	6	5	4	3	2	1	0	
\$36 (\$56)	OCF2	TOV2	ICF1	OCF1A	OCF1B	TOV1	OCF0	TOV0	TIFR
Чтение/Запись	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное значение	0	0	0	0	0	0	0	0	

• **Bit 7 - OCF2: Output Compare Flag 2:- Флаг 2 совпадения таймера/счетчика2 и данных OCR2**

Бит OCF2 устанавливается при совпадении состояния таймера/счетчика2 и содержимого регистра OCR2 (Output Compare Register 2). Бит OCF2 аппаратно очищается при обработке соответствующего вектора прерывания. Возможна очистка бита записью во флаг логической 1. При установленном бите I в регистре SREG, установленных OCIE2 (Timer/Counter2 Output Compare Interrupt Enable) и OCF2 выполняется прерывание по совпадению выхода таймера/счетчика2.

• **Bit 6 - TOV2: Timer/Counter2 Overflow Flag - Флаг переполнения таймера/счетчика2**

Бит TOV2 устанавливается при переполнении таймера/счетчика2. Он аппаратно очищается при обработке соответствующего вектора прерывания. Возможна очистка бита записью во флаг логической 1. При установленном бите I в регистре SREG, установленных OCIE2 (Timer/Counter2 Overflow Interrupt Enable) и TOV2 выполняется прерывание по переполнению таймера/счетчика2. В режиме PWM этот бит устанавливается при смене направления счета при \$00.

• **Bit 5 - ICF1: Input Capture Flag 1 - Флаг 1 захвата входа**

Бит ICF1 устанавливается в случае захвата входа, показывающего, что состояние таймера/счетчика1 переслано в входной регистр захвата ICR1. Бит очищается аппаратно при

обработке соответствующего вектора прерывания. Возможна очистка бита записью во флаг логической 1.

• **Bit 4 - OCF1A: Output Compare Flag 1A - Флаг 1A совпадения выхода**

Бит OCF1A устанавливается при совпадении состояния таймера/счетчика1 и содержимого регистра OCR1A (Output Compare Register 1A). Бит OCF1A аппаратно очищается при обработке соответствующего вектора прерывания. Возможна очистка бита записью во флаг логической 1. При установленном бите 1 в регистре SREG, установленных OCIE1A (Timer/Counter1 Compare Interrupt Enable) и OCF1A выполняется прерывание по совпадению выхода таймера/счетчика1.

• **Bit 3 - OCF1B: Output Compare Flag 1B - Флаг 1B совпадения выхода**

Бит OCF1B устанавливается при совпадении состояния таймера/счетчика1 и содержимого регистра OCR1B (Output Compare Register 1B). Бит OCF1B аппаратно очищается при обработке соответствующего вектора прерывания. Возможна очистка бита записью во флаг логической 1. При установленном бите 1 в регистре SREG, установленных OCIE1B (Timer/Counter1 Compare Interrupt Enable) и OCF1B выполняется прерывание по совпадению выхода таймера/счетчика1.

• **Bit 2 - TOV1: Timer/Counter1 Overflow Flag - Флаг переполнения таймера/счетчика1**

Бит TOV1 устанавливается при переполнении таймера/счетчика1. Он аппаратно очищается при обработке соответствующего вектора прерывания. Возможна очистка бита записью во флаг логической 1. При установленном бите 1 в регистре SREG, установленных TOIE1 (Timer/Counter1 Overflow Interrupt Enable) и TOV1 выполняется прерывание по переполнению таймера/счетчика1. В режиме PWM этот бит устанавливается при смене таймером/счетчиком1 направления счета при \$0000.

• **Bit 1 - OCF0: Output Compare Flag 0 - Флаг 0 совпадения выхода**

Бит OCF0 устанавливается при совпадении состояния таймера/счетчика0 и содержимого регистра OCR0 (Output Compare Register 0). Бит OCF0 аппаратно очищается при обработке соответствующего вектора прерывания. Возможна очистка бита записью во флаг логической 1. При установленном бите 1 в регистре SREG, установленных OCIE0 (Timer/Counter0 Output Compare Interrupt Enable) и OCF0 выполняется прерывание по совпадению выхода таймера/счетчика1.

• **Bit 0 - TOV0: Timer/Counter0 Overflow Flag - Флаг переполнения таймера/счетчика0**

Бит TOV0 устанавливается при переполнении таймера/счетчика0. Он аппаратно очищается при обработке соответствующего вектора прерывания. Возможна очистка бита записью во флаг логической 1. При установленном бите 1 в регистре SREG, установленных TOIE0 (Timer/Counter0 Overflow Interrupt Enable) и TOV0 выполняется прерывание по переполнению таймера/счетчика0. В режиме PWM этот бит устанавливается при смене таймером/счетчиком1 направления счета при \$00.

ВРЕМЯ ОТКЛИКА НА ПРЕРЫВАНИЕ

Отклик на выполнение всех разрешенных прерываний AVR составляет минимум 4 тактовых цикла. В течение 4 тактовых циклов после установки флага прерывания выполняется переход по адресу вектора прерывания для выполнения подпрограммы прерывания. В течение этих 4 циклов содержимое счетчика команд (2 байта) опускаются в стек и указатель стека декрементируется на 2. Вектор указывает переход в подпрограмму обработки прерывания и этот переход занимает 3 тактовых цикла. Если прерывание возникнет во время выполнения многоциклового команды, то команда завершается до начала обслуживания прерывания.

Возврат из подпрограммы обработки прерывания (как и вызов подпрограммы) занимает

4 тактовых цикла. В течение этих 4 циклов состояние счетчика команд (2 байта) извлекается из стека и указатель стека инкрементируется на 2. Когда AVR выходит из прерывания, он всегда возвращается в основную программу и выполняет еще одну команду, прежде, чем начать обслуживание какого либо отложенного прерывания.

Отметим, что регистр статуса SREG не обрабатывается аппаратными средствами AVR, ни для прерываний, ни для подпрограмм.

При обработке подпрограмм прерываний, требующих сохранения в SREG, запись должна выполняться программными средствами пользователя. Для прерываний, запускаемых статическими событиями (например совпадение содержимого регистра сравнения IА с состоянием таймера/счетчика1) флаг прерывания устанавливается в момент наступления события. Если флаг очищен, но условия возникновения прерывания продолжают существовать, флаг не будет устанавливаться до тех пор, пока это событие не наступит вновь.

Режимы энергосбережения (Sleep Modes)

Для перевода в любой из трех режимов энергосбережения бит SE в регистре MCUCR должен быть установлен в состояние 1. Биты SM1 и SM0 регистра MCUCR определяют какой из режимов Idle, Power Down или Power Save будет запущен командой SLEEP.

При возникновении разрешенного прерывания во время нахождения MCU в режиме энергосбережения, MCU активируется, выполняет подпрограмму обработки прерывания и продолжает работу до следующей команды SLEEP. Если во время режима энергосбережения происходит сброс, MCU активируется и начинает работу по вектору сброса. Содержимое регистрового файла, SRAM и памяти I/O в процессе активации не изменяется. В Таблице 4 представлен выбор режима энергосбережения.

Отметим, что если для возврата из режима энергосбережения Power Down или Power Save, используется запуск прерывания по уровню, то низкий уровень должен удерживаться несколько дольше, чем время задержки сброса t_{TOUT} , иначе микроконтроллер не активируется.

РЕЖИМ Idle

Если биты SM1/SM0 находятся в состоянии 00 команда SLEEP переводит MCU в режим Idle, останавливая CPU но оставляя активными таймеры/счетчики, сторожевой таймер и систему прерываний. Это обеспечивает активацию MCU внешними прерываниями и такими внутренними прерываниями, как переполнение таймера и завершение приема UART. Если активация по аналоговому компаратору не требуется, то аналоговый компаратор может быть отключен установкой бита ACD в регистре управления и статуса аналогового компаратора ACSR. Это позволит дополнительно снизить потребление в Idle режиме. При активации MCU из Idle режима CPU начинает выполнять программу незамедлительно.

РЕЖИМ Power Down

При установке битов SM1/SM0 в состояние 10 команда SLEEP переводит MCU в режим Power Down. В этом режиме останавливается внешний генератор. Пользователь может разрешить работу сторожевого таймера. Если сторожевой таймер разрешен, то активация MCU произойдет по завершении установленного в сторожевом таймере периода времени. Если внешний источник тактового сигнала подключен к выводу XTAL1, то активация MCU из режима Power Down может происходить без

задержки, обычно необходимой для стабилизации XTAL генератора. Режим такой активации разрешается программированием перемычек SUT0/SUT1 в Flash памяти. См. раздел *Режим параллельного программирования*.

РЕЖИМ Power Save

При установке битов SM1/SM0 в состояние 11 команда SLEEP переводит MCU в режим Power Save. Этот режим, за одним исключением, аналогичен режиму Power Down. Если таймер/счетчик0 тактируется асинхронно, т.е. бит AS0 в регистре ASSR установлен, таймер/счетчик0 будет работать в режиме Power Save.

MCU будет активироваться прерываниями по переполнению или совпадению выхода таймера/счетчика0.

Таймеры/счетчики

Микроконтроллеры ATmega603/103 оснащены тремя таймерами/счетчиками общего назначения - двумя 8-разрядными и одним 16-разрядным. Таймер/счетчик0, в дополнение к обычному режиму, может тактироваться асинхронно от внешнего генератора. Этот генератор оптимизирован под использование кварцевого кристалла на частоту 32768 кГц, что позволяет использовать таймер/счетчик0 как часы реального времени (Real Time Clock - RTC).

Таймер/счетчик0 оснащен своим собственным предварительным делителем. Таймеры/счетчики 1 и 2 используют выходы ступеней деления общего 10-разрядного предварительного делителя. Эти два таймера/счетчика можно использовать как таймеры с встроенной временной базой или как счетчики, переключаемые по состоянию на внешнем выводе.

Предварительные делители таймеров/счетчиков

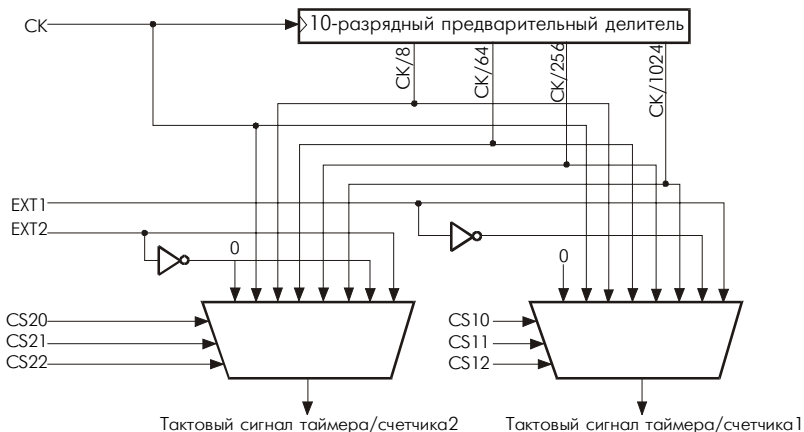


Рис. 30 Предварительный делитель таймера/счетчика1 и таймера/счетчика2

Предварительный делитель таймеров/счетчиков 1и 2 содержит четыре ступени деления: СК/8, СК/64, СК/256 и СК/1024, где СК входной тактовый сигнал. Кроме того, в качестве источников тактовых сигналов могут быть использованы сигналы от внешних источников, тактовый сигнал СК и нулевой тактовый сигнал (stop).

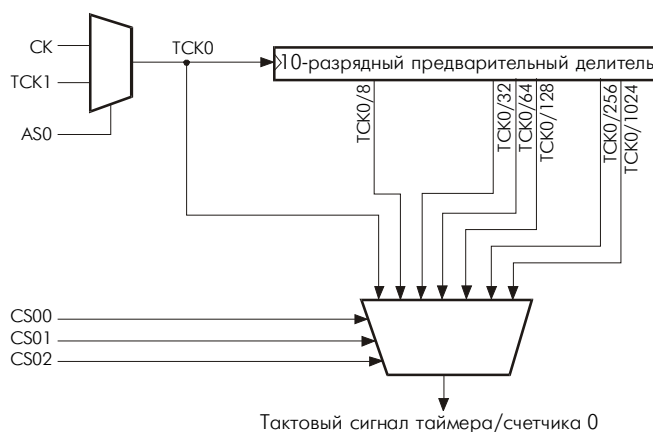


Рис. 31 Предварительный делитель таймера/счетчика0

Тактовый сигнал таймера/счетчика0 обозначен TCK0. Этот тактовый сигнал по умолчанию подключен к основному тактовому сигналу системы CK. При установке бита AS0 в регистре ASSR таймер/счетчик0 будет асинхронно тактироваться сигналом с вывода TOSC1, что позволяет использовать таймер/счетчик0 в качестве часов реального времени (RTC). Генератор оптимизирован под использование кварцевого кристалла с частотой 32768 кГц, подсоединяемого между выводами TOSC1 и TOSC2.

8-разрядные таймеры/счетчики T/C0 и T/C2

8-разрядный таймер/счетчик0 получает тактовый сигнал или непосредственно от TCK0 или после прохождения его через предварительный делитель.

8-разрядный таймер/счетчик2 получает тактовый сигнал непосредственно от CK, после прохождения его через предварительный делитель или от внешнего вывода. Оба таймера/счетчика могут быть остановлены, как это показано в описании регистров управления таймерами/счетчиками TCCR0 и TCCR2.

В регистре флагов прерывания таймеров/счетчиков TIFR хранятся различные флаги состояния регистров (переполнения, совпадения при сравнении и захвата события). Установки управляющих сигналов хранятся в регистрах управления таймерами/счетчиками TCCR0 и TCCR2. Установка разрешения/запрещения прерываний производится в регистре масок прерываний таймеров/счетчиков TIMSK.

При тактировании таймера/счетчика2 внешним тактовым сигналом этот сигнал синхронизируется с тактовой частотой CPU. Для обеспечения правильной синхронизации внешнего сигнала необходимо, чтобы минимальное время между двумя входящими тактовыми циклами было не менее одного цикла внутреннего тактового сигнала CPU. Внешний тактовый сигнал синхронизируется нарастающим фронтом внутреннего тактового сигнала CPU.

Точность и разрешение 8-разрядных таймеров/счетчиков растет с уменьшением коэффициента предварительного деления. Аналогичным образом высокий коэффициент предварительного деления удобно использовать при реализации функций с низким быстродействием или точной синхронизации редко происходящих действий.

Оба таймера/счетчика поддерживают две функции сравнения выхода, используя регистры сравнения выхода OCR0 и OCR2 как источники данных, сравниваемых с содержимым таймеров/счетчиков. В функции сравнения выхода входит и опция очистки счетчика при совпадении и формирование, при совпадении, сигнала на выводах сравнения выхода - PB4(OC0/PWM0) и PB7(OC2/PWM2).

Таймеры/счетчики 0 и 2 можно использовать как 8-разрядные широтно-импульсные модуляторы (PWM). В этом режиме таймер/счетчик, совместно с регистром совпадения выхода работают как автономный ШИМ с центрированными импульсами и без ложных выбросов. Подробнее эта функция описана в разделе *Таймеры/счетчики 0 и 2 в ШИМ режиме*.

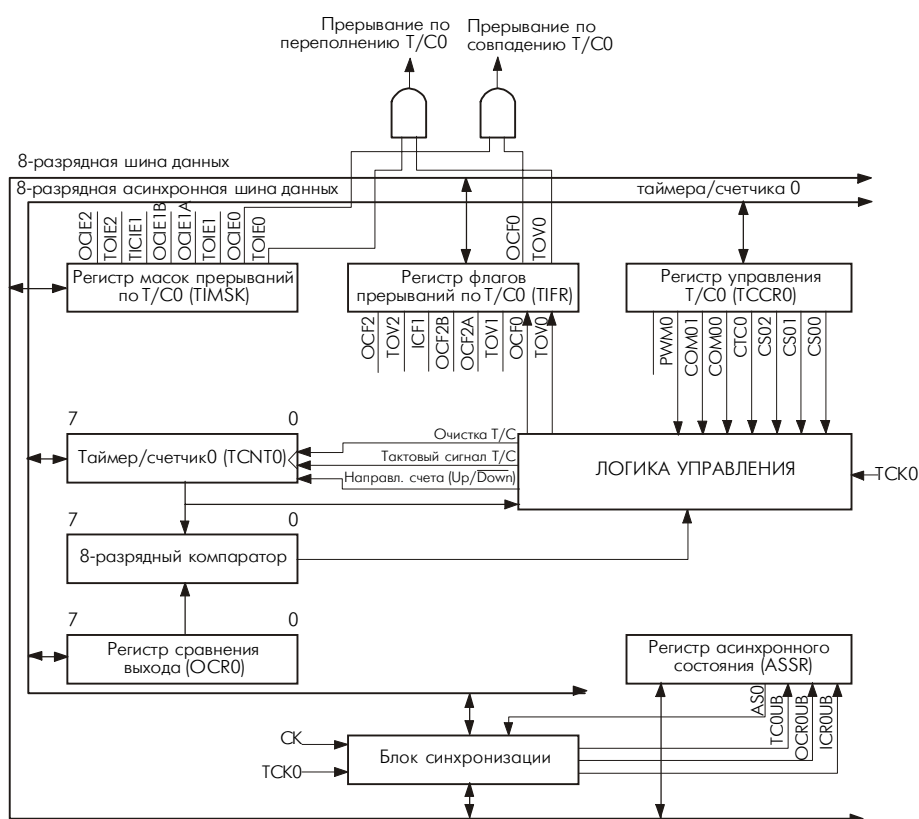


Рис. 32 Блок-схема таймера/счетчика0

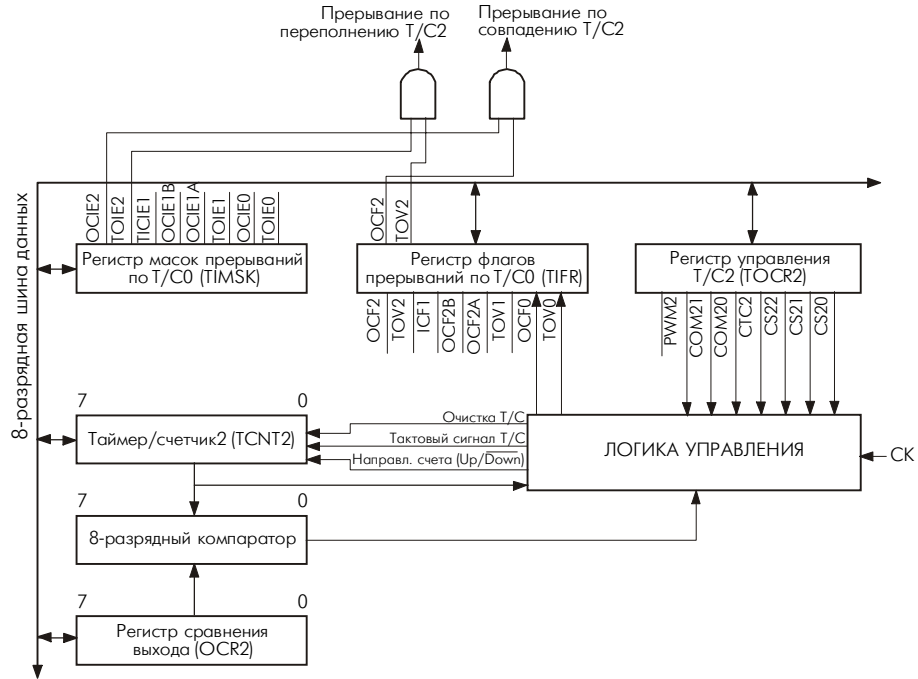


Рис. 33 Блок-схема таймера/счетчика2

РЕГИСТР УПРАВЛЕНИЯ ТАЙМЕРОМ/СЧЕТЧИКОМ0 - TCCR0 - (The Timer/Counter0 Control Register)

Биты	7	6	5	4	3	2	1	0	
\$33 (\$53)	-	PWM0	COM01	COM00	CTC0	CS02	CS01	CS00	TCCR0
Чтение/Запись	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное состояние	0	0	0	0	0	0	0	0	

РЕГИСТР УПРАВЛЕНИЯ ТАЙМЕРОМ/СЧЕТЧИКОМ2 - TCCR2 - (The Timer/Counter2 Control Register)

Биты	7	6	5	4	3	2	1	0	
\$25 (\$45)	-	PWM2	COM21	COM20	CTC2	CS22	CS21	CS20	TCCR2
Чтение/Запись	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное состояние	0	0	0	0	0	0	0	0	

• **Bit 7 - Res: Reserved Bit - Зарезервированный бит**

Данный бит в микроконтроллерах ATmega603/103 зарезервирован и при считывании всегда покажет состояние 0.

• **Bit 6 - PWM0 / PWM2: Pulse Width Modulator Enable - Разрешение широтно-импульсного модулятора**

Установленный в состояние 1 бит разрешает режим ШИМ для таймеров/счетчиков 0 или 2. Режим подробно описан в разделе *Таймеры/счетчики 0 и 2 в ШИМ режиме*.

• **Bits 5,4 - COM01, COM00 / COM21, COM20: Compare Output Mode, bits 1 and 0 - Режим сравнения выхода, биты 1 и 0**

Управляющие биты COMn1 и COMn0 устанавливают состояние выходных выводов PB4(OC0/PWM0) или PB7(OC2/PWM2), после совпадения в таймере/счетчике2. Поскольку это альтернативная функция выводов порта I/O, то соответствующий бит направления вывода должен быть установлен в состояние 1. Управляющая конфигурация показана в таблице 10.

Таблица 10. Выбор режима сравнения

COMn1	COMn0	Описание
0	0	Таймер/счетчик отсоединен от выходного вывода OCn/PWMn
0	1	Переключение выходной линии OCn/PWMn.
1	0	Очистка выходной линии OCn/PWMn (установка в состояние 0).
1	1	Установка выходной линии OCn/PWMn (установка в состояние 1).

Примечания: n = 0 или 2.

В ШИМ режиме функции этих битов отличаются. Подробное описание приведено в таблице 13. При изменении битов COMn1/COMn0 прерывание по сравнению выхода должно быть запрещено очисткой его бита разрешения прерывания в регистре TIMSK. В противном случае при изменении состояния бита может произойти прерывание.

• **Bit 3 - CTC0 / CTC2: Clear Timer/Counter on Compare match - Очистить таймер/счетчик при совпадении**

При установленном в состояние 1 бите CTC0 или CTC2 таймер/счетчик сбрасывается в состояние \$00 в течение одного тактового цикла CPU после наступления совпадения. Если бит управления сброшен, то таймер продолжает считать и не используется в процедуре сравнения. Поскольку факт совпадения детектируется в тактовом цикле CPU следующем за совпадением, то эта функция будет вести себя несколько по-другому, если коэффициент предварительного деления будет больше 1. Если используется коэффициент предварительного деления равный 1 и в регистр сравнения A установлено содержимое C, то таймер будет продолжать счет так как это делается при установленном CTC0/2.

... | C-1 | C | C+1 | 0 | 1 | ...

Если установлен коэффициент деления 8, таймер будет считать аналогично следующей последовательности:

... | C-1, C-1, C-1, C-1, C-1, C-1, C-1, C-1 | C, C, C, C, C, C, C, C | C+1, 0, 0, 0, 0, 0, 0, 0 | 1, 1, 1, ...

В ШИМ режиме состояние этого бита значения не имеет.

• **Bits 2,1,0 - CS02, CS01, CS00 / CS22, CS21, CS20: Clock Select bits 2,1 and 0 - Биты выбора тактовой частоты**

Биты 2,1 и 0 выбора тактовой частоты таймера/счетчика2 подключают выход определенной ступени предварительного делителя.

Таблица 11. Выбор коэффициента деления предварительного делителя таймера/счетчика0

CS02	CS01	CS00	Описание
0	0	0	Таймер/счетчик0 остановлен.
0	0	1	ТСК0
0	1	0	ТСК0 / 8
0	1	1	ТСК0 / 32
1	0	0	ТСК0 / 64
1	0	1	ТСК0 / 128
1	1	0	ТСК0 / 256
1	1	1	ТСК0 / 1024

Таблица 12. Выбор коэффициента деления предварительного делителя таймера/счетчика2

CS22	CS21	CS20	Описание
0	0	0	Таймер/счетчик0 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 PD7(T2), падающий фронт
1	1	1	Внешний вывод PD7(T2), нарастающий фронт

Условие Stop обеспечивает реализацию функции разрешения/запрещения таймера.

Режим деления СК реализуется непосредственным делением тактовой частоты СК. Если для тактирования таймера/счетчика 2 используется внешний источник, то переключения на выводе PD7(T2) будут воздействовать на счетчик, даже если этот вывод сконфигурирован как выход.

ТАЙМЕР/СЧЕТЧИК0 - TCNT0 - (Timer/Counter0)

Биты	7	6	5	4	3	2	1	0	
\$32 (\$42)	MSB							LSB	TCNT0
Чтение/Запись	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное состояние	0	0	0	0	0	0	0	0	

ТАЙМЕР/СЧЕТЧИК2 - TCNT2 - (Timer/Counter2)

Биты	7	6	5	4	3	2	1	0	
\$24 (\$44)	MSB							LSB	TCNT2
Чтение/Запись	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное состояние	0	0	0	0	0	0	0	0	

Содержимое этих 8-разрядных регистров является состоянием таймеров/счетчиков. Оба таймера/счетчика реализованы как счетчики по нарастанию или реверсивные (в ШИМ режиме) счетчики с возможностью чтения/записи. Если в таймер/счетчик записано некоторое значение и выбран источник тактового сигнала, то он продолжит счет с записанного значения с тактовой частотой счетчика.

**РЕГИСТР СРАВНЕНИЯ ВЫХОДА ТАЙМЕРА/СЧЕТЧИКА0 -
OCR0 - (Timer/Counter0 Output Compare Register)**

Биты	7	6	5	4	3	2	1	0	
\$31 (\$51)	MSB							LSB	OCR0
Чтение/Запись	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное состояние	0	0	0	0	0	0	0	0	

**РЕГИСТР СРАВНЕНИЯ ВЫХОДА ТАЙМЕРА/СЧЕТЧИКА2 -
OCR2 - (Timer/Counter2 Output Compare Register)**

Биты	7	6	5	4	3	2	1	0	
\$23 (\$43)	MSB							LSB	OCR2
Чтение/Запись	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное состояние	0	0	0	0	0	0	0	0	

Регистры сравнения выходов являются 8-разрядными регистрами с возможностью чтения/записи. Выполнение процедуры сравнения определяется регистрами TCCR0 и TCCR2. Совпадение при сравнении произойдет только тогда, когда таймер/счетчик досчитает до значения содержимого OCR. Программная запись одного и того же значения в таймер/счетчик и в регистр сравнения выхода не приведет к формированию совпадения при сравнении.

Совпадение при сравнении приведет к установке флага прерывания по совпадению в течение тактового цикла CPU следующего за совпадением. Необходимо принимать меры предосторожности при работе таймера/счетчика0 в асинхронном режиме, т.е. устанавливать в состояние 1 бит AC0 в регистре ASSR. При записи в регистр OCR0 значение, пересылается в регистр по TCK0 такту, следующему за операцией записи.

**ТАЙМЕРЫ/СЧЕТЧИКИ 0 и 2 в ШИМ РЕЖИМЕ -
Timer/Counter 0 and 2 in PWM mode**

При установленном ШИМ режиме таймер/счетчик и регистр сравнения выхода (OCR0 или OCR2) формируют 8-разрядный, без ложных выбросов и с правильной фазой ШИМ сигнал с выходом через PB4(OC0/PWM0) или PB7(OC2/PWM2) выходы. Таймер/счетчик работает как реверсивный счетчик, считающий от \$00 до \$FF, после чего он считает в обратную сторону до нуля и только после этого начинает новый цикл. Когда состояние счетчика совпадает с содержимым регистра сравнения выхода выходы PB4(OC0/PWM0) или PB7(OC2/PWM2) устанавливаются или очищаются, в соответствии с установленными, в регистрах управления таймерами/счетчиками TCCR0 и TCCR2, битами COM01/COM00 или COM21/COM20. См. таблицу 13.

В ШИМ режиме при записи содержимое регистра сравнения выхода пересылается на временное хранение. Содержимое фиксируется при достижении таймером/счетчиком состояния \$FF. Такой прием предохраняет от появления ШИМ импульсов увеличенной ширины (ложных выбросов) в случае несинхронной записи OCR0 или OCR2. Пример см. на рис.34.

В промежуток времени между операциями записи и фиксации считывание из OCR0 или OCR2 приведет к считыванию из места временного хранения. Это означает, что чаще всего при чтении значения уставки считывание будет производиться из OCR0/2. При состоянии регистра OCR \$00 или \$FF выход ШИМ будет удерживаться

Таблица 13. Выбор режима сравнения в ШИМ режиме

COMn1	COMn0	Эффект, оказываемый на вывод Compare/PWM
0	0	Не подсоединен
0	1	Не подсоединен
1	0	Очистка при совпадении, счет по нарастанию. Установка при совпадении, счет по убыванию (неинвертирующий ШИМ).
1	1	Очистка при совпадении, счет по убыванию. Установка при совпадении, счет по нарастанию (инвертирующий ШИМ).

Примечания: n = 0 или 2

на низком или высоком уровне, в зависимости от установок COM21/COM20 или COM11/COM10. См. таблицу 14.

В ШИМ режиме флаг переполнения таймера (TOV0 или TOV2) устанавливается при смене направления счета при \$00. Прерывания по переполнению таймеров 0 и 2 работают так же, как и в нормальном режиме таймеров/счетчиков, т.е. они срабатывают когда TOV0 или TOV2 установлены, и разрешены прерывания по переполнению таймера и глобальному прерыванию. Это относится также к флагам сравнения выхода таймеров и прерываниям.

Частота ШИМ будет соответствовать тактовой частоте таймера деленной на 510.

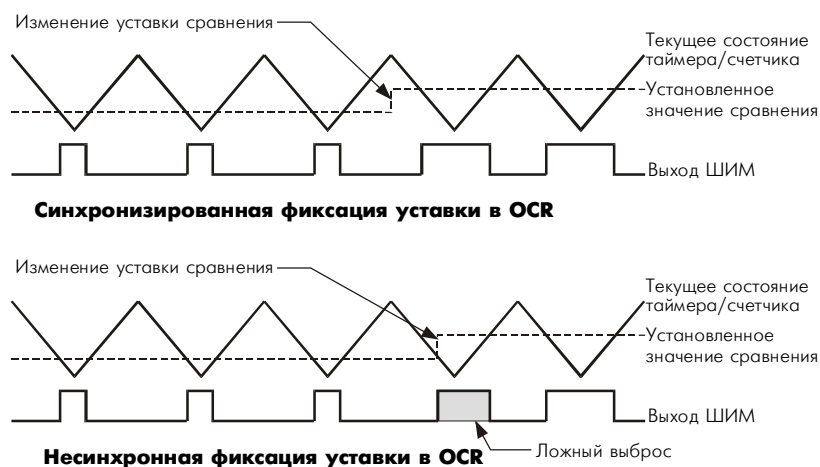


Рис. 34 Эффект несинхронной фиксации OCR

Таблица 14. Состояния ШИМ выходов при OCRn = \$00 или \$FF

COMn1	COMn0	OCRn	Выход PWMn
1	0	\$00	L - низкий уровень
1	0	\$FF	H - высокий уровень
1	1	\$00	H - высокий уровень
1	1	\$FF	L - низкий уровень

Примечания: n = 0 или 2

АСИНХРОННАЯ РАБОТА ТАЙМЕРА/СЧЕТЧИКА

При синхронной работе таймера/счетчика все операции и тактирование идентичны работе таймера/счетчика. Однако асинхронная работа имеет некоторые особенности.

- Это важно! При переключении между асинхронным и синхронным тактированием таймера/счетчика регистры таймера TCNT0, OCR0 и TCCR0 могут быть повреждены. Безопасное переключение выполняется следующей последовательностью действий:

1. Запрещаются прерывания OCIE0 и TOIE0 таймера.
2. Соответствующей установкой AS0 выбирается источник тактового сигнала.
3. В TCNT0, OCR0 и TCCR0 записываются новые значения.
4. Если выполняется переключение в асинхронный режим: ожидать TCN0UB, OCR0UB и TCR0UB.
5. Разрешить прерывания, если это необходимо.

- Генератор оптимизирован под использование часового кристалла с частотой 32,768 кГц. Внешний тактовый сигнал, подаваемый на этот вывод, проходит через тот же самый усилитель с полосой пропускания 256 кГц. Таким образом, внешний тактовый сигнал должен быть в диапазоне от 0 Гц до 256 кГц. Частота внешнего тактового сигнала, подаваемого на вывод TOSC1, не должна превышать одной четвертой от тактовой частоты CPU. Отметим, что тактовая частота CPU может быть ниже частоты XTAL, если разрешено деление частоты XTAL.

- При записи в один из регистров TCNT0, OCR0 или TCCR0 записываемая величина пересылается в регистр временного хранения и фиксируется после двух положительных фронтов TOSC1. Пользователь не должен записывать новое значение прежде, чем содержимое регистра временного хранения не будет передано по назначению. Каждый из указанных регистров имеет свой собственный регистр временного хранения, это означает, к примеру, что запись в TCNT0 не исказится при записи в OCR0. Для того, чтобы убедиться в выполнении пересылки в регистр назначения используется регистр статуса асинхронного режима (Asynchronous Status Register - ASSR).

- При вводе режима Sleep после записи в регистры TCNT0, OCR0 or TCCR0 пользователь должен ожидать пока записываемый регистр не будет обновлен, если таймер/счетчик используется для активации прибора. В ином случае MCU перейдет в режим Sleep прежде, чем изменения окажут какой либо эффект. Это особенно важно если для активации прибора используется прерывание по сравнению выхода; сравнение выхода запрещается во время записи в OCR0 или TCNT0. Если цикл записи не завершен (т.е. пользователь введет режим Sleep прежде, чем бит OCR0UB будет сброшен в 0) прибор никогда не получит совпадения при сравнении и MCU не будет активирован. Если таймер/счетчик используется для активации прибора из режима Power Save и если пользователь намеревается возобновить режим Power Save, то необходимо предпринимать меры предосторожности - для сброса логики прерывания необходим один цикл TOSC1. Если время между активацией и восстановлением режима Power Save меньше одного цикла TOSC1 прерывание не произойдет и прибор не будет активирован. Если пользователь сомневается в том, что промежуток времени перед восстановлением режима Power Save достаточен, необходимо использовать следующий алгоритм:

1. Записать значение в TCCR0, TCNT0 или OCR0.
2. Подождать пока соответствующий флаг занятого обновления в регистре

ASSR не будет сброшен в 0.

3. Ввести режим Power Save.

- Генератор частоты 32кГц таймера/счетчика0 работает всегда, за исключением режима Power Down. При восстановлении питания или активации из режима Power Down пользователь должен помнить о том, что генератору для стабилизации необходимо время порядка одной секунды. Пользователю рекомендуется выждать не менее одной секунды прежде чем использовать таймер/счетчик0 после подачи питания или активации из режима Power Down.

- Описание активации прибора из режима Power Save при асинхронном тактировании таймера. Когда условия прерывания выполнены, на следующем цикле тактовой частоты таймера начинается процесс активации, т.е. состояние таймера должно увеличиться как минимум на единицу, прежде, чем процессор сможет прочитать состояние счетчика. Флаги прерываний обновляются за три тактовых цикла процессора после запуска тактовой частоты процессора. В течение этих циклов процессор выполняет команды но условия прерывания еще не читаемы и подпрограмма обработки прерывания не может начать выполнение.

Во время асинхронной работы синхронизация флагов прерываний асинхронного таймера занимает три тактовых цикла процессора плюс один цикл таймера. Таким образом содержимое таймера увеличивается минимум на единицу, прежде чем процессор сможет прочитать содержимое таймера вызвавшее установку флага прерывания. Вывод сравнения выхода меняет состояние в соответствии с тактовым сигналом таймера и не синхронизирован с тактовой частотой процессора.

РЕГИСТР СТАТУСА АСИНХРОННОГО РЕЖИМА - ASSR - (Asynchronous Status Register)

Биты	7	6	5	4	3	2	1	0	
\$30 (\$50)	-	-	-	-	AS0	TCNOUB	OCR0UB	TDR0UB	ASSR
Чтение/Запись	R	R	R	R	R/W	R	R	R	
Начальное состояние	0	0	0	0	0	0	0	0	

• Bit 7..4 - Res: Reserved Bits - Запрезервированные биты

Эти биты в микроконтроллерах ATmega603/103 зарезервированы и при считывании всегда будут показывать 0.

• Bit 3 - AS0: Asynchronous Timer/Counter0 - Асинхронный режим таймера/счетчика0

При установленном в состояние 1 бите таймер/счетчик0 тактируется сигналом вывода TOSC1. При очищенном бите таймер/счетчик0 тактируется внутренним тактовым сигналом СК. При изменении состояния этого бита содержимое TCNT0 может быть повреждено.

• Bit 2 - TCNOUB: Timer/Counter0 Update Busy - Таймер/счетчик0 занят для обновления

Бит устанавливается в состояние 1 при работе таймера/счетчика0 в асинхронном режиме и записанном TCNT0. При обновлении записанного в TCNT0 значения содержимым регистра временного хранения бит аппаратно очищается.

Бит в логическом состоянии 0 означает, что TCNT0 готов к обновлению новым значением.

• Bit 1 - OCR0UB: Output Compare Register0 Update Busy - Сравнение выхода регистра0 занято для обновления

Бит устанавливается в состояние 1 при работе таймера/счетчика0 в асинхронном режиме и записанном OCR0. При обновлении записанного в OCR0 значения содержимым

регистра временного хранения бит аппаратно очищается.

Бит в логическом состоянии 0 означает, что OCR0 готов к обновлению новым значением.

• **Bit 0 - TCROUB: Timer/Counter Control Register0 Update Busy - Регистр управления таймера/счетчика0 занят для обновления**

Бит устанавливается в состояние 1 при работе таймера/счетчика0 в асинхронном режиме и записанном TCCR0. При обновлении записанного в TCCR0 значения содержимым регистра временного хранения бит аппаратно очищается.

Бит в логическом состоянии 0 означает, что TCCR0 готов к обновлению новым значением. Если запись выполняется в любой из трех регистров таймера/счетчика0 когда их флаги занятости для обновления установлены, то записываемое значение может быть повреждено и привести к нежелательному прерыванию.

Считывание состояний TCNT0, OCR0 и TCCR0 отличается. При чтении состояния TCNT0 считывается действительное содержимое, при чтении состояний OCR0 или TCCR0 считывается содержимое регистров временного хранения.

16-разрядный таймер/счетчик1

Блок-схема таймера/счетчика1 приведена на рис. 35

16-разрядный таймер/счетчик1 может получать тактовый сигнал от СК, СК после предварительного делителя и от внешнего вывода. Кроме того его можно остановить, как показано в описании регистра управления таймером/счетчиком1 - TCCR1B (Timer/Counter1 Control Register). В регистрах управления TCCR1A и TCCR1B находятся различные флаги, указывающие на переполнение, совпадение при сравнении и случаи захвата событий. В регистре масок прерываний TIMSK (Timer/Counter Interrupt Mask Register) устанавливаются разрешения/запрещения прерываний таймера/счетчика1. При внешнем тактировании таймера/счетчика1 внешний сигнал синхронизируется частотой тактового генератора CPU. Для правильной работы таймера/счетчика1 по внешнему тактовому сигналу минимальное время между двумя переключениями внешнего тактового сигнала должно быть не менее одного периода тактового сигнала CPU. Синхронизация внешнего тактового сигнала ведется нарастающим фронтом внутреннего тактового сигнала CPU.

Наилучшие точность и разрешение 16-разрядный таймер/счетчик1 обеспечивает при наименьшем коэффициенте предварительного деления. С другой стороны, высокий коэффициент предварительного деления удобен при реализации таймером/счетчиком1 низкоскоростных функций или точной синхронизации редко происходящих действий. Таймер/счетчик1 поддерживает две функции сравнения выхода, используя регистры сравнения выходов А и В - OCR1A и OCR1B в качестве источников данных, сравниваемых с содержимым таймера/счетчика1. Функции сравнения выхода включают очистку счетчика по совпадению сравненияА и воздействие на выводы сравнения выхода при обоих совпадениях сравнения.

Таймер/счетчик1 может быть использован в качестве 8, 9 или 10-разрядного широтно-импульсного модулятора. В этом режиме счетчик и регистры OCR1A/OCR1B работают как сдвоенный самостоятельный ШИМ со сцентрированными импульсами, без формирования ложных импульсов. См. раздел *Таймер/счетчик1 в ШИМ режиме*, где подробно описана эта функция.

Функция захвата входа таймера/счетчика1 обеспечивает захват содержимого таймера/счетчика1 в регистр захвата входа, запускаемый внешним событием на выводе входа захвата PD4/(IC1). Реальные установки захвата события определяются регистром управления таймером/счетчиком1 TCCR1B (Timer/Counter1 Control Register). Кроме

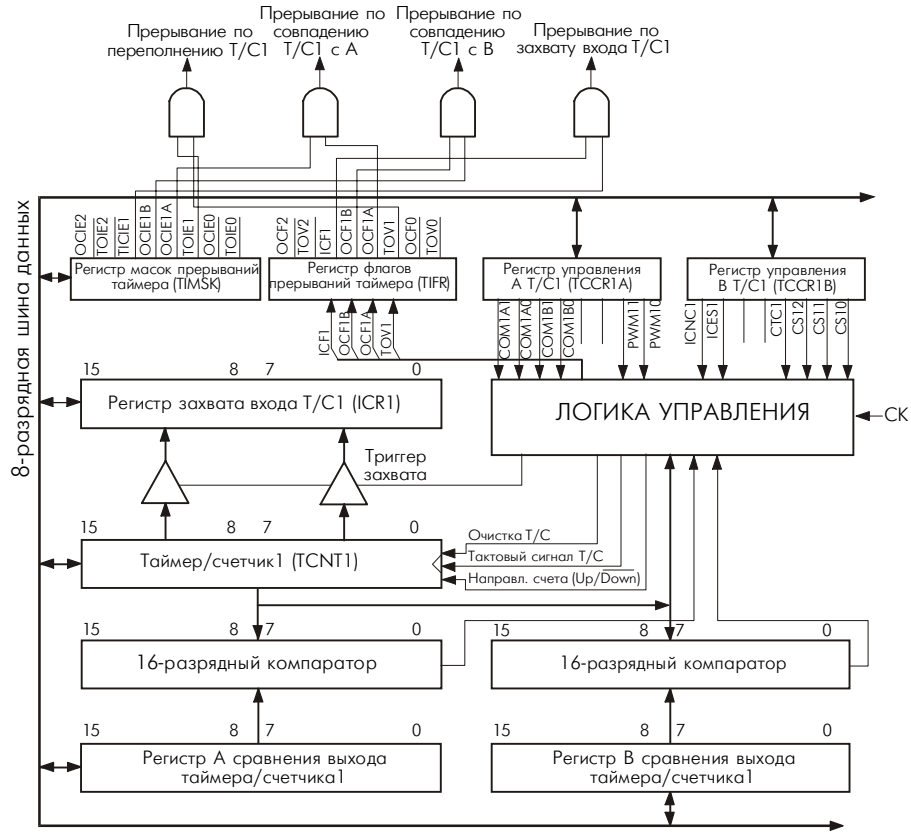


Рис. 35 Блок-схема таймера/счетчика 1

того, для переключения входа захвата может быть использован аналоговый компаратор. Подробнее данная функция описана в разделе *Аналоговый компаратор*. Если разрешена функция подавления шума, действительные условия переключения события захвата тестируются четырьмя выборками, прежде чем захват будет активирован. Тестирование сигнала на входном выводе производится с частотой XTAL.

РЕГИСТР УПРАВЛЕНИЯ А ТАЙМЕРА/СЧЕТЧИКА 1 - TCCR1A - (Timer/Counter1 Control Register A)

Биты	7	6	5	4	3	2	1	0	
\$2F (\$4F)	COM1A1	COM1A0	COM1B1	COM1B0	-	-	PWM11	PWM10	TCCR1A
Чтение/Запись	R/W	R/W	R/W	R/W	R	R	R/W	R/W	
Начальное состояние	0	0	0	0	0	0	0	0	

• **Bits 7,6 - COM1A1, COM1A0: Compare Output Mode1A, bits 1 and 0 - Режим1A сравнения выхода, биты 1 и 0**

Управляющие биты COM1A1 и COM1A0 определяют характер сигнала выхода, следующего за совпадением при сравнении таймера/счетчика 1. Сигнал выхода

поступает на вывод ОС1А (Output CompareA). Поскольку это альтернативная функция порта I/O, то соответствующий бит управления направлением должен быть установлен в 1 (вывод работает на выход). Конфигурирование управления представлено в таблице 15.

Таблица 15. Выбор режима сравнения 1

COM1X1	COM1X0	Описание
0	0	Таймер/счетчик1 отключен от вывода выхода ОС1X
0	1	Переключение выходной линии ОС1X.
1	0	Очистка выходной линии ОС1X (на линии низкий уровень).
1	1	Установка выходной линии ОС1X (на линии высокий уровень).

Примечание: X = A или B

• **Bits 5,4 - COM1B1, COM1B0: Compare Output Mode1B, bits 1 and 0 - Режим1B сравнения выхода, биты 1 и 0**

Управляющие биты COM1B1 и COM1B0 определяют характер сигнала выхода, следующего за совпадением при сравнении таймера/счетчика1. Сигнал выхода поступает на вывод ОС1В (Output CompareB). Поскольку это альтернативная функция порта I/O, то соответствующий бит управления направлением должен быть установлен в 1 (вывод работает на выход). Конфигурирование управления представлено в таблице 15.

В ШИМ режиме функции этих битов отличаются. Подробное описание приведено в таблице 19.

При изменении битов COM1X1/COM1X0 прерывания по сравнению выхода1 должны быть запрещены очисткой битов разрешения прерывания в регистре TIMSK. В противном случае при изменении битов может произойти прерывание

• **Bits 3..2 - Res: Reserved bits - Зарезервированные биты**

Эти биты в микроконтроллерах АТмега603/103 зарезервированы и при считывании всегда будут 0.

• **Bits 1..0 - PWM11, PWM10: Pulse Width Modulator Select Bits - Биты выбора режима ШИМ**

Данные биты определяют установку режима ШИМ, как это показано в таблице 16. Подробности см. в разделе *Таймер/счетчик1 в ШИМ режиме*.

Таблица 16. Выбор ШИМ режима

PWM11	PWM10	Description
0	0	Работа таймера/счетчика1 в ШИМ режиме запрещена
0	1	Работа таймера/счетчика1 в 8-разрядном ШИМ режиме
1	0	Работа таймера/счетчика1 в 9-разрядном ШИМ режиме
1	1	Работа таймера/счетчика1 в 10-разрядном ШИМ режиме

**РЕГИСТР УПРАВЛЕНИЯ В ТАЙМЕРА/СЧЕТЧИКА1 -
TCCR1B - (Timer/Counter1 Control Register B)**

Биты	7	6	5	4	3	2	1	0	
\$2E (\$4E)	ICNC1	ICES1	-	-	CTC1	CS12	CS11	CS10	TCCR1B
Чтение/Запись	R/W	R/W	R	R	R/W	R/W	R/W	R/W	
Начальное состояние	0	0	0	0	0	0	0	0	

• Bit 7 - ICNC1: Input Capture1 Noise Canceler (4 CKs) - Установка режима подавления шума на входе захвата 1

При сброшенном в состояние 0 бите ICNC1 функция подавления шума входного триггера захвата запрещена. Вход захвата переключается по первому нарастающему/падающему фронту, поступившему на вывод входа захвата PD4(IC1). При установленном в состояние 1 бите ICNC1 выполняются четыре последовательных опроса состояния вывода PD4(IC1) и все четыре выборки должны иметь одинаковый (высокий/низкий), определяемый битом ICES1, уровень. Частота опроса соответствует частоте XTAL.

• Bit 6 - ICES1: Input Capture1 Edge Select - Выбор фронта срабатывания на входе захвата 1

При сброшенном в состояние 0 бите ICES1 содержимое таймера/счетчика1, по падающему фронту на выводе входа захвата PD4(IC1), пересылается в регистр захвата входа ICR1. При установленном в 1 бите ICES1 содержимое таймера/счетчика1 пересылается в регистр захвата входа ICR1 по нарастающему фронту на выводе входа захвата PD4(IC1).

• Bits 5, 4 - Res: Reserved bits - Зарезервированные биты

Эти биты в микроконтроллерах ATmega603/103 зарезервированы и при считывании всегда будут 0.

• Bit 3 - CTC1: Clear Timer/Counter1 on Compare Match - Очистка таймера/счетчика1 по совпадению

При установленном в состояние 1 бите CTC1 таймер/счетчик1 сбрасывается в состояние \$0000 в течение тактового цикла, следующего за совпадением при сравненииA. Если бит CTC1 очищен, таймер/счетчик1 продолжает отсчет и не реагирует на совпадение при сравнении. Поскольку совпадение при сравнении детектируется в течение тактового цикла CPU следующего за совпадением, то поведение функции будет отличаться при установке коэффициента предварительного деления таймера/счетчика1 больше 1. При коэффициенте предварительного деления 1 и установленном в регистре сравненияA состоянии C таймер будет считать в соответствии с установкой CTC1:

... | C-1 | C | C+1 | 0 | 1 | ...

При установленном коэффициенте предварительного деления 8 таймер будет считать подобно:

... | C-1, C-1, C-1, C-1, C-1, C-1, C-1, C-1 | C, C, C, C, C, C, C, C | C+1, 0, 0, 0, 0, 0, 0, 0 | ...

В ШИМ режиме состояние бита CTC1 значения не имеет.

• Bits 2,1,0 - CS12, CS11, CS10: Clock Select1, bit 2,1 and 0 - Выбор источника тактовой частоты, биты 2,1 и 0

Установкой состояния данных битов производится выбор источника тактового сигнала (в том числе коэффициента предварительного деления)

Stop условие выполняет функцию разрешения/запрещения таймера/счетчика1 В режимах с предварительным делением на соответствующий коэффициент делится частота СК тактового генератора. При использовании внешнего тактирования

необходимо выполнить соответствующие установки в регистре управления направлением (очистка переводит вывод в режим входа).

Таблица 17. Выбор источника тактового сигнала таймера/счетчика1

CS12	CS11	CS10	Описание
0	0	0	Stop условие - таймер/счетчик1 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 тактирующий сигнал на выводе T1, нарастающий фронт
1	1	1	Внешний тактирующий сигнал на выводе T1, падающий фронт.

ТАЙМЕР/СЧЕТЧИК1 - TCNT1H и TCNT1L

Биты	15	14	13	12	11	10	9	8	
\$2D (\$4D)	MSB								TCNT1H TCNT1L
\$2C (\$4C)								LCB	
Чтение/Запись	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное состояние	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	

16-разрядный регистр содержит текущее значение 16-разрядного таймера/счетчика1. С тем, чтобы CPU могло считывать/записывать и старший и младший байты этого регистра одновременно, обращение реализовано посредством 8-разрядного регистра временного хранения(TEMP). Этот регистр используется также при обращении к OCR1A, OCR1B и ICR1. Если основная программа и подпрограммы обработки прерываний используют обращение к регистрам посредством TEMP, то прерывания должны быть запрещены на время обращения из основной программы

• Запись в таймер/счетчик1 - TCNT1

Когда CPU производит запись в старший байт (TCNT1H) записываемые данные размещаются в регистре TEMP. Затем, когда CPU производит запись в младший байт (TCNT1L) данные младшего байта объединяются с байтом данных регистра TEMP и все 16 битов одновременно переписываются в регистр таймера/счетчика TCNT1. Следовательно, при 16-разрядных операциях обращение к старшему байту (TCNT1H) должно выполняться первым. При использовании таймера/счетчика1 в качестве 8-разрядного таймера достаточно производить запись только младшего байта.

• Чтение таймера/счетчика1 - TCNT1

Когда CPU считывает младший байт (TCNT1L), то содержимое TCNT1L направляется непосредственно в CPU, содержимое старшего байта (TCNT1H) размещается в регистре TEMP и при считывании CPU старшего байта (TCNT1H) его содержимое CPU принимает из регистра TEMP. Следовательно, при 16-разрядных операциях первым должно выполняться обращение к младшему байту (TCNT1L). При

КТЦ-МК

использовании таймера/счетчика1 в качестве 8-разрядного таймера достаточно производить запись только младшего байта.

Таймер/счетчик1 выполнен в виде счетчика с нарастанием или реверсивного счетчика (в ШИМ режиме) и возможностью чтения/записи. Если в таймер/счетчик1 занесено некоторое значение и выбран источник тактового сигнала, то таймер/счетчик1 продолжит отсчет через один тактовый цикл после установки в нем записанного значения.

РЕГИСТРА СРАВНЕНИЯ А ВЫХОДА ТАЙМЕРА/СЧЕТЧИКА1 - OCR1AH и OCR1AL - (Timer/Counter1 Output Compare Register)

Биты	15	14	13	12	11	10	9	8	
\$2B (\$4B)	MSB								
\$2A (\$4A)								LSB	OCR1AH OCR1AL
Чтение/Запись	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное состояние	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	

РЕГИСТРЫ СРАВНЕНИЯ В ВЫХОДА ТАЙМЕРА/СЧЕТЧИКА1 - OCR1BH и OCR1BL - (Timer/Counter1 Output Compare Register)

Биты	15	14	13	12	11	10	9	8	
\$29 (\$49)	MSB								
\$28 (\$48)								LSB	OCR1BH OCR1BL
Чтение/Запись	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное состояние	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	

16-разрядные регистры сравнения выхода обеспечивают и чтение и запись.

Регистры сравнения выхода таймера/счетчика1 хранят данные постоянно сравниваемые с состоянием таймера/счетчика1. Действие, запускаемое совпадением при сравнении определяется содержимым регистра управления и состояния таймера/счетчика1. Совпадение при сравнении может произойти только если таймер/счетчик1 досчитает до значения содержимого OCR. Если в TCNT1 и OCR1A или OCR1B программно будут записаны одинаковые значения, то значения, то совпадение при сравнении сгенерировано не будет.

Совпадение при сравнении устанавливает флаг прерывания по совпадению в тактовом цикле CPU следующем за самим совпадением.

Поскольку регистры сравнения выхода OCR1A и OCR1B являются 16-разрядными, то для обеспечения одновременного занесения старшего и младшего байтов данных в регистры OCR1A/B используется регистр временного хранения TEMP. Когда CPU записывает старший байт, то данные временно сохраняются в регистре TEMP. Когда же CPU записывает младший байт в OCR1AL или OCR1BL, то одновременно содержимое регистра OCR1BH переписывается в OCR1AH или OCR1BH. Следовательно, при 16-разрядных операциях старшие байты регистров OCR1A/B должны записываться первыми.

Кроме того, регистр TEMP используется при обращении к TCNT1 и ICR1. Если основная программа и подпрограммы обработки прерываний, используют обращение к регистрам посредством TEMP, то прерывания должны быть запрещены на время обращения из основной программы.

РЕГИСТР ЗАХВАТА ВХОДА ТАЙМЕРА/СЧЕТЧИКА1 - ICR1H и ICR1L - (Timer/Counter1 Input Capture Register)

Биты	15	14	13	12	11	10	9	8	
\$27 (\$47)	MSB								ICR1H ICR1L
\$26 (\$46)								LSB	
	7	6	5	4	3	2	1	0	
Чтение/Запись	R	R	R	R	R	R	R	R	
	R	R	R	R	R	R	R	R	
Начальное состояние	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	

16-разрядный регистр захвата входа обеспечивает только чтение содержимого.

При обнаружении на выводе захвата входа PD4(IC1) нарастающего или падающего фронта сигнала (определяемого установкой ICES1) текущее состояние таймера/счетчика1 пересылается в регистр захвата входа ICR1. Одновременно устанавливается в состояние 1 флаг захвата входа ICF1.

Поскольку регистр захвата входа является 16-разрядным, то для обеспечения одновременного чтения старшего и младшего байтов данных регистра ICR1 используется регистр временного хранения TEMP. При считывании CPU данных младшего байта содержимое ICR1L пересылается в CPU, а содержимое старшего байта ICR1H размещается в регистре TEMP, чтение старшего байта заключается в переносе в CPU содержимого регистра временного хранения TEMP. Следовательно, при чтении всего 16-разрядного регистра операцию чтения необходимо начинать с младшего байта ICR1L. Регистр TEMP используется также при обращении к TCNT1, OCR1A и OCR1B. Если основная программа и подпрограммы обработки прерываний, используют обращение к регистрам посредством TEMP, то прерывания должны быть запрещены на время обращения из основной программы.

ТАЙМЕР/СЧЕТЧИК1 в ШИМ РЕЖИМЕ

При установленном ШИМ режиме таймер/счетчик1 и регистры сравнения выхода А и В (OCR1A и OCR1B), образуют сдвоенный 8, 9 или 10-разрядный автономный генератор ШИМ с правильным чередованием фаз, отсутствием ложных выбросов и выходами на выводы PD5(OC1A) и OC1B. Таймер/счетчик1 работает как реверсивный счетчик, считающий от \$0000 до TOP (см. таблицу 18), при котором направление счета меняется и отсчет ведется до нуля, после чего цикл повторяется. Когда состояние счетчика совпадет с содержимым 10 младших битов OCR1A или OCR1B, выводы PD5(OC1A)/OC1B устанавливаются или очищаются, в соответствии с установками битов COM1A1/COM1A0 или COM1B1/COM1B0 в регистре управления таймером/счетчиком1 TCCR1A. Подробности см. в таблице 19.

Таблица 18. TOP значения таймера и частота ШИМ

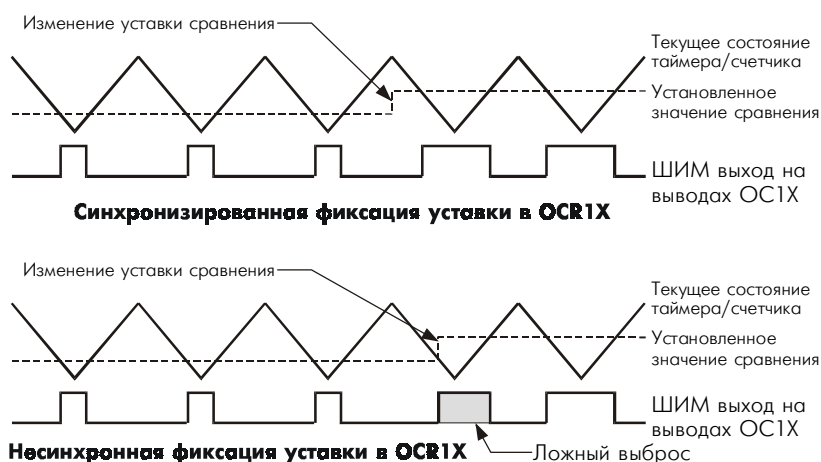
Разрешение ШИМ	TOP значения таймера	Частота ШИМ
8-разрядное	\$00FF (255)	$f_{TC1} / 510$
9- разрядное	\$01FF (511)	$f_{TC1} / 1022$
10- разрядное	\$03FF (1023)	$f_{TC1} / 2046$

Таблица 19. Выбор режима сравнения¹ в ШИМ режиме

COM1X1	COM1X0	Выходной сигнал на OCX1
0	0	Не подключен
0	1	Не подключен
1	0	Очищается по совпадению при счете вверх. Устанавливается по совпадению при счете вниз (не инвертированный ШИМ).
1	1	Очищается по совпадению при счете вниз. Устанавливается по совпадению при счете вверх (инвертированный ШИМ).

Примечание: X = A или B

Отметим, что в ШИМ режиме младшие 10 разрядов OCR1A/OCR1B, при записи, пересылаются в ячейки временного хранения. Они фиксируются по достижении таймером/счетчиком¹ значения TOP. Таким способом обеспечивается защита от появления уширенных ШИМ импульсов (ложных выбросов - glitches) при несинхронной записи OCR1A/OCR1B. См. пример на рис. 36.



Примечание: X = A или B

Рис. 36. Эффект несинхронной фиксации OCR1.

При чтении OCR1A или OCR1B, в промежутке времени между записью и фиксацией, будет считано содержимое ячейки временного хранения. Это означает, что более раннее записанное значение всегда будет считываться из OCR1A/B.

Когда OCR1 содержит \$0000 или TOP, вывод OC1A/OC1B остается на низком уровне, соответственно установкам COM1A1/COM1A0 или COM1B1/COM1B0. Это отображено в таблице 20.

Таблица 20. Состояния выходов в ШИМ режиме при OCR1X = \$0000 или TOP

COM1X1	COM1X0	OCR1X	Состояние выводов OC1X
1	0	\$0000	L
1	0	TOP	H
1	1	\$0000	H
1	1	TOP	L

Примечание: X = A или B

В ШИМ режиме флаг переполнения таймера1 (TOV1) устанавливается при смене направления счета по достижении значения \$0000. Прерывание по переполнению таймера1 работает так же как и в обычном режиме таймера/счетчика, т.е. оно выполняется когда TOV1 установлен и разрешены прерывания по переполнению таймера1 и глобальные прерывания. Это относится и к флагам сравнения выхода таймера1 и прерываниям.

Сторожевой таймер (Watchdog Timer)

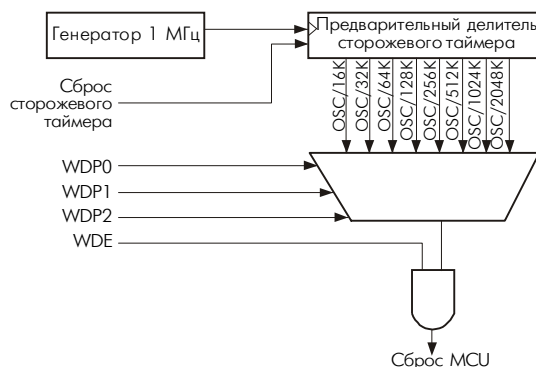


Рис. 37 Сторожевой таймер.

Сторожевой таймер тактируется отдельным встроенным генератором, работающим с частотой 1 МГц при типовом напряжении питания VCC=5 В (см. типовые значения при других значениях VCC). Установкой коэффициента предварительного деления можно изменять длительность интервала до сброса по сторожевому таймеру от 16 тыс. до 2048 тыс. циклов (от 16 до 2048 мс). Команда WDR (Watchdog Reset) сбрасывает сторожевой таймер.

С момента сброса сторожевого таймера можно установить восемь периодов длительности тактового сигнала, устанавливая, таким образом длительность периода сброса. Если период сброса завершается (в течение этого периода не поступил сигнал сброса сторожевого таймера), то микроконтроллер ATmega603/103 сбрасывается и его работа продолжается по вектору сброса. Подробности тактирования сброса по сторожевому таймеру см. выше в разделе *Сброс по сторожевому таймеру*.

При разрешении сторожевого таймера его состояние неизвестно и прежде, чем разрешать сторожевой таймер, необходимо выполнить команду WDR. В ином случае прибор может быть сброшен прежде, чем будет выполнена команда WDR прописанная после разрешения. Для предотвращения случайного запрета, запрет сторожевого таймера должен сопровождаться специальной процедурой выключения. Подробности в описании регистра управления сторожевым таймером.

**РЕГИСТР УПРАВЛЕНИЯ СТОРОЖЕВЫМ ТАЙМЕРОМ -
WDTCR - (Watchdog Timer Control Register)**

Биты	7	6	5	4	3	2	1	0	
\$21 (\$41)	-	-	-	WDTOE	WDE	WDP2	WDP1	WDP0	WDTCR
Чтение/Запись	R	R	R	R/W	R/W	R/W	R/W	R/W	
Начальное состояние	0	0	0	0	0	0	0	0	

• **Bits 7..5 - Res: Reserved bits - Зарезервированные биты**

Эти биты в микроконтроллерах ATmega603/103 зарезервированы и при считывании всегда будут 0.

• **Bit 4 - WDTOE: Watch Dog Turn Off Enable - Разрешение отключения сторожевого таймера**

Данные биты должны быть установлены в состояние 1 при очистке бита WDE. В ином случае сторожевой таймер не будет запрещен. Установленный бит аппаратно очищается после четырех тактовых циклов. См. описание бита WDE в процедуре запрета сторожевого таймера.

• **Bit 3 - WDE: Watch Dog Enable - Разрешение сторожевого таймера**

Если бит WDE установлен в состояние 1 (сторожевой таймер разрешен) и если бит WDE очищен, то функционирование сторожевого таймера запрещено. Бит WDE может быть очищен только если установлен бит WDTOE. Для запрещения разрешенного сторожевого таймера необходимо выполнить следующую процедуру:

1. В одной операции записать логическую 1 в WDTOE и WDE. Логическая 1 должна быть записана в WDE даже если этот бит был установлен перед началом операции запрета сторожевого таймера.
2. За время последующих четырех тактовых циклов записать логический 0 в WDE. Сторожевой таймер будет запрещен.

• **Bits 2..0 - WDP2, WDP1, WDP0: Watch Dog Timer Prescaler 2, 1 and 0 - биты установки коэффициента предварительного деления сторожевого таймера**

Состояния битов WDP2, WDP1 и WDP0 определяют коэффициент предварительного деления тактовой частоты разрешенного сторожевого таймера. Коэффициенты и соответствующие им промежутки времени представлены в таблице 21.

Таблица 21. Выбор коэффициента предварительного деления тактовой частоты сторожевого таймера

WDP2	WDP1	WDP0	Длительность цикла сторожевого таймера
0	0	0	16 тыс. циклов
0	0	1	32 тыс. циклов
0	1	0	64 тыс. циклов
0	1	1	128 тыс. циклов
1	0	0	256 тыс. циклов
1	0	1	512 тыс. циклов
1	1	0	1024 тыс. циклов
1	1	1	2048 тыс. циклов

Обращение к EEPROM при чтении/записи

Обращение к EEPROM ведется посредством регистров расположенных в пространстве I/O. Время обращения при записи составляет от 2,5 до 4 мс, в зависимости от напряжения VCC. Однако существует специальная функция, которая позволяет пользователю обнаруживать момент готовности EEPROM к записи новых данных может быть установлено специальное прерывание по завершению записи EEPROM (EEPROM Write Complete). Случайная запись в EEPROM предотвращается выполнением специальной процедуры, показанной подробнее в описании регистра управления EEPROM.

После процедуры записи в EEPROM CPU, прежде чем начать выполнение следующей команды, останавливается на два тактовых цикла. При чтении EEPROM CPU останавливается на 4 тактовых цикла.

РЕГИСТР АДРЕСА EEPROM- EEARH, EEARL - (EEPROM Address Register)

Бит	15	14	13	12	11	10	9	8	
\$1F (\$3F)	-	-	-	-	EEAR11	EEAR10	EEAR9	EEAR8	EEARH
\$1E (\$3E)	EEAR7	EEAR6	EEAR5	EEAR4	EEAR3	EEAR2	EEAR1	EEAR0	EEARL
Чтение/Запись	R	R	R	R	R/W	R/W	R/W	R/W	
Начальное значение	0	0	0	0	0	0	0	0	

Посредством регистров адреса EEARH и EEARL определяется адрес в пространстве адресов EEPROM емкостью 2 К/4 К. Байты данных EEPROM адресуются линейно в пределах от 0 до 2047/4095. Микроконтроллер ATmega603 оснащен EEPROM с адресным пространством 2 К и бит I/O EEAR11 при чтении всегда будет в состоянии 0.

РЕГИСТР ДАННЫХ EEPROM - EEDR - (EEPROM Data Register)

Бит	7	6	5	4	3	2	1	0	
\$1D (\$3D)	MSB							LSB	EEDR
Чтение/Запись	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное значение	0	0	0	0	0	0	0	0	

• Bits 7..0 - EEDR7..0: EEPROM Data - Данные EEPROM

В процессе выполнения операции записи в регистре EEDR содержатся данные, записываемые в EEPROM по адресу, задаваемому регистром EEAR. При чтении регистр EEDR содержит данные, считываемые из EEPROM по адресу, определяемому EEAR.

РЕГИСТР УПРАВЛЕНИЯ EEPROM - EECR - (EEPROM Control Register)

Бит	7	6	5	4	3	2	1	0	
\$1C (\$3C)	-	-	-	-	EEIE	EEMWE	EWE	EERE	EECR
Чтение/Запись	R	R	R	R	R	R/W	R/W	R/W	
Начальное значение	0	0	0	0	0	0	0	0	

• **Bits 7..4 - Res: Reserved bits - Зарезервированные биты**

Эти биты в микроконтроллерах ATmega603/103 зарезервированы и при считывании всегда покажут состояние 0.

• **Bit 3 - EERIE: EEPROM Ready Interrupt Enable - Разрешение прерывания по готовности EEPROM**

При установленных в состояние 1 бите EERIE и I-бите регистра SREG разрешается прерывание по готовности EEPROM. При очищенном бите EERIE прерывание запрещено. Запрос прерывания по готовности EEPROM при очищенном бите EERIE генерируется непрерывно.

• **Bit 2 - EEMWE: EEPROM Master Write Enable - Управление разрешением записи EEPROM**

Бит EEMWE определяет, будет ли установленный в состояние 1 бит EEMWE разрешать запись в EEPROM. При установленном в состояние 1 бите EEMWE установка бита EEMWE приведет к записи в EEPROM по заданному адресу. Если же бит EEMWE будет находиться в состоянии 0, то установка бита EEMWE никакого эффекта не окажет. Установленный программным путем бит EEMWE аппаратно очищается через четыре тактовых цикла. См. описание процедуры записи EEPROM в описании бита EEMWE.

• **Bit 1 - EEWWE: EEPROM Write Enable - Разрешение записи EEPROM**

Сигнал разрешения записи EEPROM EEWWE (EEPROM Write Enable Signal) является стробом записи в EEPROM. Запись установленных данных по установленному адресу EEPROM выполняется по установке бита EEWWE. При этом бит EEMWE обязательно должен быть в состоянии 1, иначе запись не произойдет. Процесс записи EEPROM выполняется следующей процедурой (очередность выполнения пунктов 2 и 3 значения не имеет):

- 1.
2. Подождать сброса бита EEWWE в состояние 0.
3. Записать новый адрес в EEAR (при необходимости).
4. Записать новые данные (при необходимости).
5. Установить в состояние 1 бит EEMWE регистра EECR.
6. В течение четырех тактовых циклов после установки EEMWE установить в состояние 1 бит EEWWE.

По истечении времени записи (типичное значение 2,5 мс при $V_{CC} = 5$ В или 4 мс при $V_{CC} = 2,7$ В) бит EEWWE аппаратно очищается. Пользовательское программное обеспечение может тестировать состояние этого бита для определения момента сброса его в 0, чтобы начать запись следующего байта. После установки бита EEWWE CPU, прежде чем начать выполнение следующей команды, останавливается на два тактовых цикла.

• **Bit 0 - EERE: EEPROM Read Enable - Разрешение чтения EEPROM**

Сигнал разрешения чтения EERE (EEPROM Read Enable Signal) является стробом чтения EEPROM. Бит EERE должен быть установлен по установлению в регистре EEAR требуемого адреса. После аппаратной очистки бита EERE считываемые данные будут располагаться в регистре EEDR. Считывание байта данных выполняется одной командой и не требует опроса бита EERE. При установленном бите EERE CPU останавливается на четыре тактовых цикла, прежде чем начнет выполнение следующей команды. Пользователю необходимо тестировать состояние бита EEWWE перед началом операции чтения. Если новые данные или адрес будут записываться в регистры I/O EEPROM в то время, когда будет выполняться операция записи, то операция записи будет прервана и результат записи будет неопределенным.

Защита EEPROM от разрушения

Содержимое EEPROM может быть разрушено при снижении напряжения V_{cc} до уровня, при котором CPU и EEPROM работают неправильно. Для решения этой проблемы используются те же приемы, что используются для обеспечения сохранности данных в EEPROM системных плат.

Разрушение данных EEPROM, при слишком низком напряжении питания, происходит в двух случаях. Во-первых, для правильного выполнения последовательности операций записи необходимо, чтобы напряжение питания было не ниже уровня, гарантирующего правильное их выполнение. Во-вторых, само CPU, при слишком низком напряжении питания, может неправильно выполнять команды.

Разрушения данных легко избежать если следовать следующим рекомендациям (достаточно выполнения одной из трех):

1. Удержание сигнала сброса в активном (низком) состоянии во время снижения напряжения питания. Лучше всего это реализовывать внешней схемой защиты от снижения напряжения, называемой часто Brown-Out Detector (BOD). Вопросы организации сброса по падению напряжения и детектирования снижения напряжения рассмотрены в руководстве по применению AVR 190 и AVR 180.
2. Удержание ядра AVR микроконтроллера в Power Down Sleep режиме в период снижения напряжения V_{CC} . Это предотвратит неправильное декодирование и выполнение команд CPU, что защитит регистры EEPROM от случайных записей.
3. Сохранение констант в Flash памяти, если нет необходимости изменять их программно. Flash память не изменяется CPU и, следовательно, не может быть повреждена.

Последовательный периферийный интерфейс - SPI - (Serial Peripheral Interface)

Последовательный периферийный интерфейс (SPI) обеспечивает высокоскоростной синхронный обмен данными между микроконтроллерами ATmega603/103 и периферийными устройствами или между несколькими микроконтроллерами ATmega603/103.

Основные характеристики SPI интерфейса:

- Полнодуплексный 3-проводный синхронный обмен данными.
- Режим работы ведущий или ведомый.
- Обмен данными с передаваемыми первыми старшим или младшим битами.
- Четыре программируемые скорости обмена данными.
- Флаг прерывания по окончании передачи.
- Активация из Idle режима (только в режиме ведомого)

Соединения между ведущим и ведомым CPU, использующими SPI интерфейс, показаны на рис. 39. Вывод PB1(SCK) является выходом тактового сигнала ведущего микроконтроллера и входом тактового сигнала ведомого. По записи ведущим CPU данных в SPI регистр начинает работать тактовый генератор SPI и записанные данные сдвигаются через вывод выхода PB2(MOSI) ведущего микроконтроллера на вывод входа PB2 (MOSI) ведомого микроконтроллера. После сдвига одного байта тактовый генератор SPI останавливается, устанавливая флаг окончания передачи (SPIF). Если в регистре SPCR будет установлен бит разрешения прерывания SPI (SPIE), то произойдет запрос прерывания. Вход выбора ведомого $PB0(\overline{SS})$, для выбора индивидуального SPI устройства в качестве ведомого, устанавливается на низкий уровень. При установке высокого

уровня на выводе PB0(SS) порт SPI деактивируется и вывод PB2(MOSI) может быть использован в качестве вывода входа. Режим ведущий/ведомый может быть установлен и программным способом установкой или очисткой бита MSTR в регистре управления SPI.

Два сдвиговых регистра ведущего и ведомого микроконтроллеров можно рассматривать как один разнесенный 16-разрядный циклический сдвиговый регистр. См. Рис 39. При сдвиге данных из ведущего микроконтроллера в ведомый одновременно происходит сдвиг данных из ведомого микроконтроллера в ведущий, т.е. в течение одного цикла сдвига происходит обмен данными между ведущим и ведомым микроконтроллерами.

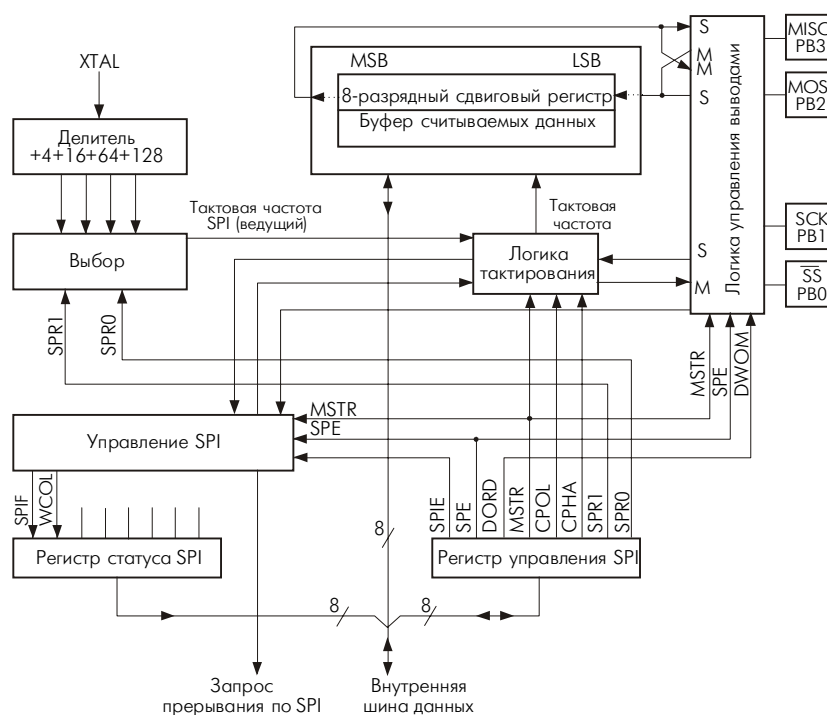


Рис. 38 Блок-схема SPI

В системе организовано одиночное буферирование передающей стороны и двойное буферирование на приемной стороне. Это означает то, что передаваемые символы не могут быть записаны в регистр данных SPI прежде, чем будет полностью завершен цикл сдвига.

С другой стороны, при приеме данных принимаемый символ должен быть считан из регистра данных SPI прежде, чем будет завершен прием следующего символа, в противном случае предшествующий символ будет потерян.

При разрешенном SPI направления данных выводов MOSI, MISO, SCK и SS настраиваются в соответствии со следующей таблицей:

Таблица 22. Настройка выводов SPI

Вывод	Направление, ведущий SPI	Направление, ведомый SPI
MOSI	Определяется пользователем	Вход
MISO	Вход	Определяется пользователем
SCK	Определяется пользователем	Вход
\overline{SS}	Определяется пользователем	Вход

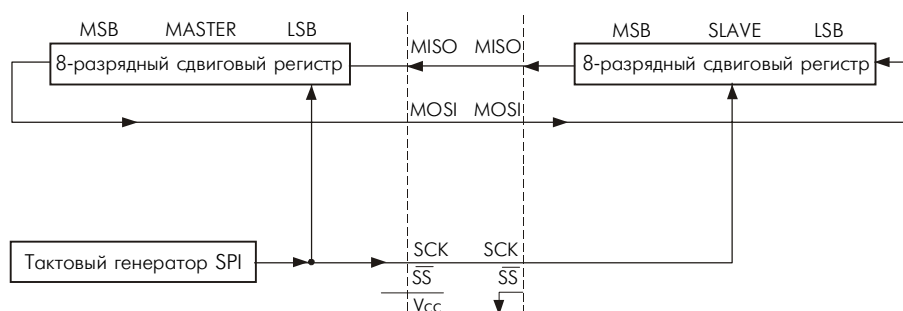


Рис. 39 Межсоединения ведущего и ведомого SPI.

ФУНКЦИОНИРОВАНИЕ ВХОДА \overline{SS}

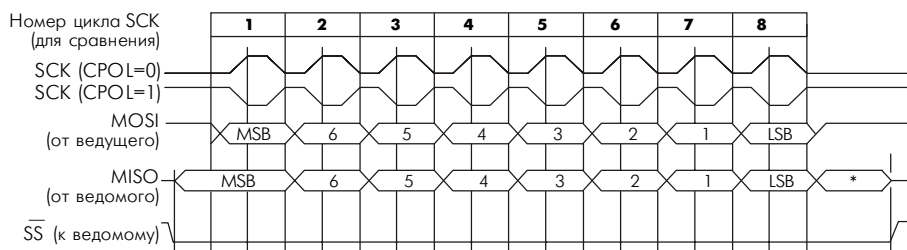
При работе SPI ведущим (бит MSTR регистра SPCR установлен), пользователь имеет возможность установить направление работы вывода \overline{SS} . Если вывод \overline{SS} сконфигурирован как выход, то вывод является выводом общего назначения и он не активируется системой SPI. Если же вывод \overline{SS} сконфигурирован как вход, то для обеспечения работа ведущего SPI он должен удерживаться на высоком уровне. Если, в режиме ведущего, вывод \overline{SS} является входом и внешней периферийной схемой на него подан низкий уровень, то SPI воспримет его как обращение другого ведущего SPI к себе как к ведомому. Чтобы избежать конфликтной ситуации на шине, система SPI выполняет следующие действия:

1. Бит MSTR в регистре SPCR очищается и SPI система становится ведомой. Результатом этого является то, что MOSI и SCK выводы становятся входами.
2. Устанавливается флаг SPIF регистра SPSR и, если разрешено прерывание SPI, начнется выполнение подпрограммы обработки прерывания.

Таким образом, когда управляемый прерыванием передающий SPI используется в ведущем режиме, и существует вероятность подачи на вывод \overline{SS} управляющего сигнала низкого уровня, прерывание должно всегда проверять установлен ли еще бит MSTR. Если же бит MSTR был очищен выбором режима ведомого, то он должен быть установлен пользователем.

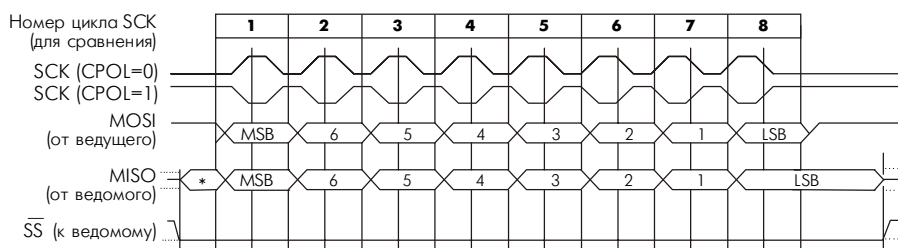
Если же SPI работает в режиме ведомого, то вывод \overline{SS} постоянно работает на вход. Если на вывод \overline{SS} подан низкий уровень, то SPI активируется и MISO, если это определено пользователем, становится выходом. Все остальные выходы являются входами. Если вывод \overline{SS} удерживается на высоком уровне, то все выходы являются входами, SPI пассивен, что означает, что он не будет получать входящих данных.

Существует четыре варианта комбинации фазы и полярности SCK относительно последовательных данных, определяемые управляющими битами CPHA и CPOL. Форматы передачи данных SPI показаны на Рис. 40 и 41.



*Неопределенно, но обычно это MSB уже принятого символа

Рис. 40. Формат пересылок SPI при CPHA = 0



* Неопределенно, но обычно это LSB уже переданного символа.

Рис. 41. Формат пересылок SPI при CPHA = 1

РЕГИСТР УПРАВЛЕНИЯ SPI - SPCR - (Control Register)

Бит	7	6	5	4	3	2	1	0	
\$OD (\$2D)	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPRO	SPCR
Чтение/Запись	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное значение	0	0	0	0	0	0	0	0	

• **Bit 7 - SPIE: SPI Interrupt Enable - Разрешение прерывания SPI.**

Установка бита SPIE в состояние 1 приводит к установке бита SPIF регистра SPSR и, при разрешении глобального прерывания, к выполнению прерывания SPI.

• **Bit 6 - SPE: SPI Enable - Разрешение SPI**

Установка бита SPE в состояние 1 разрешает подключение \overline{SS} , MOSI, MISO и SCK к выводам PB4, PB5, PB6 и PB7.

• **Bit 5 - DORD: Data Order - Порядок данных**

При установленном в состояние 1 бите DORD передача слова данных происходит LSB вперед. При очищенном бите DORD первым передается MSB слова данных.

• **Bit 4 - MSTR: Master/Slave Select - Выбор режима ведущий/ведомый**

При установленном в состояние 1 бите MSTR SPI работает в ведущем режиме и при очищенном бите в ведомом режиме. Если \overline{SS} сконфигурирован как вход и на него подан низкий уровень при установленном MSTR, то MSTR будет сброшен и будет установлен бит SPIF в регистре SPSR. Чтобы вновь разрешить ведущий режим SPI, пользователь должен установить MSTR.

• **Bit 3 - CPOL: Clock Polarity - Полярность тактового сигнала**

SCK в режиме ожидания находится на высоком уровне при установленном в состоянии 1 бите CPOL и на низком уровне при сброшенном бите CPOL. См. рис. 40 и 41.

• **Bit 2 - CPHA : Clock Phase - Фаза тактового сигнала**

Работа этого бита отражена на рис. 40 и 41.

• **Bits 1,0 - SPR1, SP0: SPI Clock Rate Select 1 and 0 - Выбор частоты тактового сигнала, биты 1 и 0**

Эти два бита управляют частотой тактового сигнала прибора, работающего в ведущем режиме. В ведомом режиме состояния битов влияния не оказывают. Состояния битов и устанавливаемый коэффициент деления частоты f_{CL} показаны в таблице:

Таблица 23. Соотношения между SCK и частотой генератора.

SPR1	SPR0	Тактовая частота SCK
0	0	$f_{cl} / 4$
0	1	$f_{cl} / 16$
1	0	$f_{cl} / 64$
1	1	$f_{cl} / 128$

РЕГИСТР СТАТУСА SPI - SPSR - (Status Register)

Бит	7	6	5	4	3	2	1	0	
\$0E (\$2E)	SPIF	WCOL	-	-	-	-	-	-	SPSR
Чтение/Запись	R	R	R	R	R	R	R	R	
Начальное значение	0	0	0	0	0	0	0	0	

• **Bit 7 - SPIF: SPI Interrupt Flag - Флаг прерывания по SPI**

По завершении обмена последовательными данными бит SPIF устанавливается в состояние 1 и, если бит SPIE в регистре SPCR установлен и разрешено глобальное прерывание, генерируется сигнал прерывания. Бит SPIF очищается аппаратно при выполнении подпрограммы обработки соответствующего вектора прерывания. Бит SPIF может быть очищен также при первом считывании состояния регистра статуса SPI, с установленным битом SPIF, с последующим обращением к регистру данных SPI (SPDR).

• **Bit 6 - WCOL: Write Collision flag - Флаг ошибки при записи**

Бит WCOL устанавливается в состояние 1 если в процессе передачи данных выполнялась запись в регистр данных (SPDR). Чтение содержимого регистра данных, как и запись в него, выполненные во время пересылки данных, могут привести к неверному результату. Бит WCOL (и бит SPIF) аппаратно очищаются (сбрасываются в состояние 0) при первом считывании регистра статуса SPI, с установленным WCOL, с последующим обращением к регистру данных SPI (SPDR).

• **Bit 5..0 - Res: Reserved bits - Резервированные биты**

Эти биты в микроконтроллерах ATmega603/103 зарезервированы и при считывании всегда покажут состояние 0.

РЕГИСТР ДАННЫХ SPI - SPDR - (SPI Data Register)

Бит	7	6	5	4	3	2	1	0	
\$0F (\$2F)	MSB							LSB	SPDR
Чтение/Запись	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное значение	0	0	0	0	0	0	0	0	

Регистр данных SPI представляет собой регистр с возможностью чтения/записи и предназначен для пересылки данных между регистровым файлом и сдвиговым регистром SPI. Запись в регистр SPDR инициирует передачу данных, считывание регистра приводит к чтению сдвигового регистра приема.

UART - универсальный асинхронный приемопередатчик

Микроконтроллеры ATmega603/103 оснащены полнодуплексными универсальными приемопередатчиками (UART). Их основные возможности следующие:

- Генератор обеспечивает любую скорость передачи информации в бодах
- Высокая скорость передачи при низкой частоте XTAL
- 8-разрядный или 9-разрядный форматы данных
- Фильтрация шума
- Обнаружение переполнения
- Обнаружение ошибок формирования кадров
- Детектирование бита ложного старта
- Три отдельных прерывания: по завершению передачи (TX Complete), по пустому регистру передаваемых данных (TX Data Register Empty) и по завершению приема (RX Complete).

Передача данных

Блок-схема передатчика UART показана на рис. 42.

Передача данных инициируется записью передаваемых данных в регистр данных I/O UART (UDR). Данные пересылаются из UDR в сдвиговый регистр передачи в следующих случаях:

- Новый символ записан в UDR после того как был выведен из регистра стоповый бит предшествовавшего символа. Сдвиговый регистр загружается немедленно.
- Новый символ записан в UDR прежде, чем был выведен стоповый бит предшествовавшего символа. Сдвиговый регистр загружается после выхода стопового бита передаваемого символа, находившегося в сдвиговом регистре.

Если из 10(11)-разрядного сдвигового регистра передачи выведена вся информация (сдвиговый регистр передачи пуст) данные из UDR пересылаются в сдвиговый регистр. В это время устанавливается бит UDRE (UART Data Register Empty) регистра статуса UART (USR). При установленном в состояние 1 бите UDRE UART готов принять следующий символ. Запись в UDR очищает бит UDRE. В то самое время, когда данные пересылаются из UDR в 10(11)-разрядный сдвиговый регистр, бит 0 сдвигового регистра сбрасывается в состояние 0 (состояние 0 - стартовый бит) а бит 9 или 10 устанавливается в состояние 1 (состояние 1 - стоповый бит). Если в регистре управления UART (UCR) установлен бит CHR9 (т.е. выбран режим 9-разрядного слова данных), то бит TXB8 регистра UCR пересылается в бит 9 сдвигового регистра передачи.

Сразу после пересылки данных в сдвиговый регистр тактом бод-генератора стартовый бит сдвигается на вывод TXD. За ним следует LSB данных. Когда будет

выдан стоповый бит сдвиговый регистр загружается новой порцией данных, если она была записана в UDR во время передачи. В процессе загрузки бит UDRE находится в установленном состоянии. Если же новые данные не будут загружены в UDR до выдачи стопового бита, флаг UDRE остается установленным. В этом случае, после того как стоповый бит будет присутствовать на выводе TXD в течение одного такта, в регистре статуса UART (USR) устанавливается флаг завершения передачи TXC (TX Complete Flag).

Установленный в состояние 1 бит TXEN регистра UCR разрешает передачу UART. При очищенном бите TXEN (сброшенном в состояние 0) вывод PE1 может быть использован в качестве вывода I/O общего назначения. При установленном бите TXEN передатчик UART подключается к PE1 и использует его в качестве вывода выхода, независимо от установки бита DDE1 в DDRE.

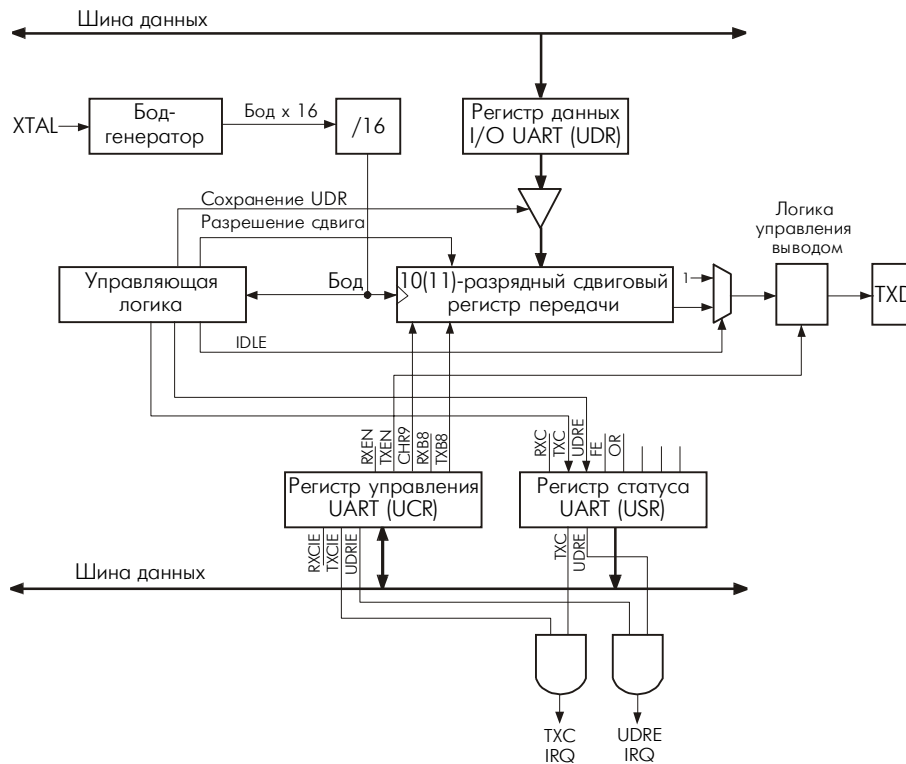


Рис. 42. Передатчик UART

Прием данных

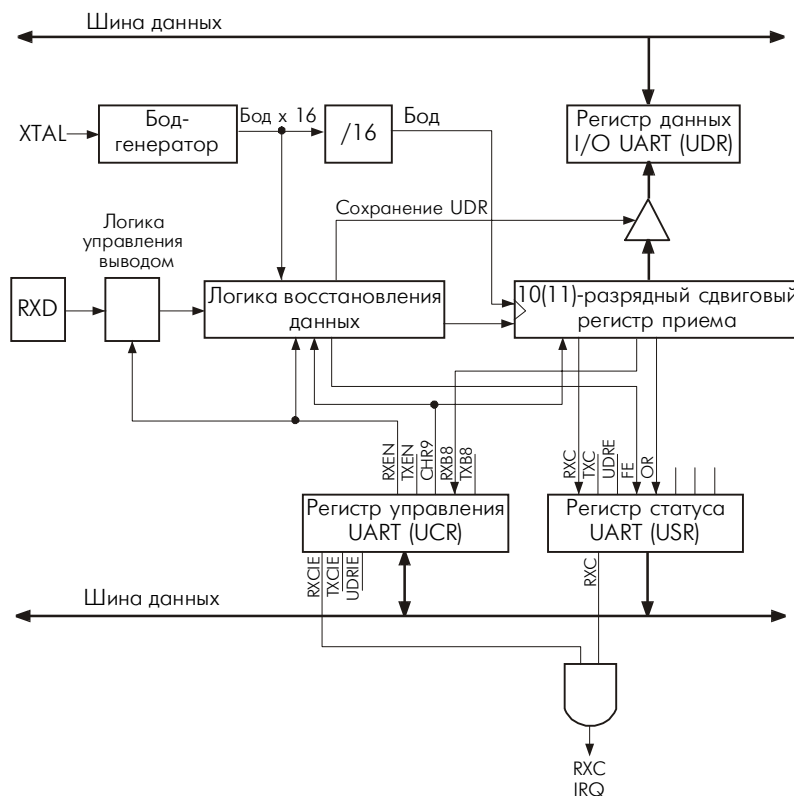


Рис. 43 Приемник UART.

Логика восстановления данных производит выборку состояний вывода RXD с частотой в 16 раз большей, чем частота бодов. При нахождении линии в пассивном состоянии одиночная выборка нулевого логического уровня будет интерпретироваться как падающий фронт стартового бита и будет запущена последовательность детектирования стартового бита. Считается, что первая выборка обнаружила первый нулевой логический уровень вероятного стартового бита. На выборках 8, 9 и 10 приемник вновь тестирует вывод RXD на изменение логических состояний. Если две или более из этих трех выборок обнаружат логические 1, то данный вероятный стартовый бит отвергается как шумовой всплеск и приемник начнет выявлять и анализировать следующие переходы из 1 в 0.

Если же был обнаружен действительный стартовый бит, то начинается выборка следующих за стартовым битом информационных битов. Эти биты также тестируются на выборках 8, 9 и 10. Логическое состояние бита принимается по двум и более (из трех) одинаковым состояниям выборок. Все биты вводятся в сдвиговый регистр приемника с тем значением, которое было определено тестированием выборок. Тестирование выборок битов принимаемых символов показано на Рис. 44.



Рис. 44. Тестирование выборок принимаемых данных

При поступлении стопового бита необходимо чтобы не менее двух выборок из трех подтвердили прием стопового бита (показали высокий уровень). Если же две или более выборок покажут состояния 0, то, при пересылке принятого байта в UDR, в регистре статуса UART (USR) устанавливается бит ошибки кадра FE (Framing Error). Для обнаружения ошибки кадра пользователь перед чтением регистра UDR должен проверять состояние бита FE. Флаг FE очищается при считывании содержимого регистра данных UART (UDR). Вне зависимости от того принят правильный стоповый бит или нет, данные пересылаются в регистр UDR и устанавливается флаг RXC в регистре статуса UART (USR). Регистр UDR фактически является двумя физически отдельными регистрами, один из которых служит для передачи данных и другой для приема. При считывании UDR обращение ведется к регистру приема данных, при записи обращение ведется к регистру передачи. Если выбран режим обмена 9-разрядными словами данных (установлен бит CHR9 регистра UCR), при пересылке данных в UDR бит RXB8 регистра UCR загружается в бит 9 сдвигового регистра передачи. Если после получения символа к регистру UDR не было обращения, начиная с последнего приема, в регистре UCR устанавливается флаг переполнения (OR). Это означает, что новые данные, пересылаемые в сдвиговый регистр не могут быть переданы в UDR и потеряны. Бит OR буферизован и доступен тогда, когда в UDR читается байт достоверных данных. Пользователю, для обнаружения переполнения, необходимо всегда проверять флаг OR после считывания содержимого регистра UDR.

При очищенном (сброшенном в логическое состояние 0) бите RXEN регистра UCR приемник запрещен. Это означает, что вывод PE0 может использоваться в качестве вывода I/O общего назначения. При установленном бите RXEN, приемник UART подключается к выводу PE0, который работает как вывод входа, вне зависимости от установки бита DDE0 в DDRE.

При установке UART вывода PE0 на работу в качестве входа, бит PORTE0 может использоваться для управления нагрузочным резистором вывода.

Управление UART

РЕГИСТР ДАННЫХ UART - UDR - (UART I/O Data Register)

Бит	7	6	5	4	3	2	1	0	
\$0C (\$2C)	MSB							LSB	UDR
Чтение/Запись	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное значение	0	0	0	0	0	0	0	0	

В действительности регистр UDR является двумя физически разделенными регистрами - регистром передачи данных и регистром приема данных, использующими одни и те же адреса I/O. При записи в регистр запись производится в регистр передачи данных UART, при чтении происходит чтение содержимого регистра приема данных UART.

РЕГИСТР СТАТУСА UART - USR - (UART Status Register)

Бит	7	6	5	4	3	2	1	0	
\$0B (\$2B)	RXC	TXC	UDRE	FE	DOR	-	-	-	USR
Чтение/Запись	R	R/W	R	R	R	R	R	R	
Начальное значение	0	0	0	0	0	0	0	0	

Регистр USR обеспечивает только чтение информации о состоянии UART.

• **Bit 7 - RXC: UART Receive Complete - Прием завершен**

Данный бит устанавливается в состояние 1 при пересылке принятого символа из сдвигового регистра приема в UDR. Бит устанавливается вне зависимости от отсутствия или наличия ошибок приема кадра. При установленном в UCR бите RXCIE и установленном бите RXC выполняется прерывание по завершению приема UART. Бит RXC очищается при считывании UDR. При приеме данных инициированном прерыванием, подпрограмма обработки прерывания по завершению приема UART должна считать UDR, с тем, чтобы очистить RXC, иначе по окончании подпрограммы обработки прерывания произойдет новое прерывание

• **Bit 6 - TXC: UART Transmit Complete - Передача завершена**

Данный бит устанавливается в состояние 1 когда весь символ (включая стоповый бит) выведен из сдвигового регистра передачи и в UDR не записаны новые данные. Этот флаг используется при полудуплексном связанном интерфейсе, когда оборудование передачи должно установить режим приема и освободить коммуникационную шину сразу после завершения передачи.

При установленном в регистре UCR бите TXCIE установка TXC приведет к выполнению прерывания по завершению передачи UART. Флаг TXC очищается аппаратно при выполнении обработки соответствующего вектора прерывания. Очистить бит TXC можно записью в бит логической 1.

• **Bit 5 - UDRE: UART Data Register Empty - Регистр данных пуст**

Данный бит устанавливается в состояние 1 когда символ, записанный в UDR, пересылается в сдвиговый регистр передачи. Установка этого бита означает, что передатчик готов к получению нового символа для передачи.

Когда бит UDRIE в UCR установлен, до тех пор пока установлен UDRE, выполняется прерывание по завершению передачи UART. Бит UDRE очищается при записи в UDR. При приеме данных инициированном прерыванием, подпрограмма обработки прерывания по пустому регистру данных UART должна считать UDR, с тем, чтобы очистить UDRE, иначе по окончании подпрограммы прерывания произойдет новое прерывание. Во время сброса бит UDRE устанавливается в состояние 1 с тем, чтобы индицировать готовность передатчика.

• **Bit 4 - FE: Framing Error - Ошибка кадра**

Данный бит устанавливается в состояние 1 при обнаружении условий ошибочного приема кадра, т.е. когда стоповый бит входящего символа в состоянии 0. Бит FE очищается при приеме стопового бита с логическим уровнем 1.

• **Bit 3 - DOR: Data OverRun - Переполнение данных**

Бит DOR устанавливается в состояние 1 при обнаружении условий переполнения, т.е. когда символ, уже находящийся в регистре UDR, не считан перед пересылкой нового символа из сдвигового регистра приема. Бит DOR буферизован, что означает, что он будет оставаться установленным пока не будут считаны правильные данные из UDR. Бит DOR очищается (сбрасывается в 0) когда данные приняты и пересланы в UDR.

• **Bits 2..0 - Res: Reserved bits - Зарезервированные биты**

Эти биты в микроконтроллерах ATmega603/103 зарезервированы и при считывании всегда покажут состояние 0.

РЕГИСТР УПРАВЛЕНИЯ UART - UCR - (UART Control Register)

Бит	7	6	5	4	3	2	1	0	
\$0A (\$2A)	RXCIE	TXCIE	UDRIE	RXEN	TXEN	CHR9	RXB8	TXB8	UCR
Чтение/Запись	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное значение	0	0	0	0	0	0	0	0	

• **Bit 7 - RXCIE: RX Complete Interrupt Enable - Разрешение прерывания по завершению приема**

При установленном в состояние 1 бите RXCIE и установленном разрешении глобального прерывания установка бита RXC в регистре USR приведет к выполнению прерывания по завершению приема.

• **Bit 6 - TXCIE: TX Complete Interrupt Enable - Разрешение прерывания по завершению передачи**

При установленном в состояние 1 бите TXCIE и установленном разрешении глобального прерывания установка бита TXC в регистре USR приведет к выполнению прерывания по завершению передачи.

• **Bit 5 - UDRIE: UART Data Register Empty Interrupt Enable - Разрешение прерывания по пустому регистру данных**

При установленном в состояние 1 бите UDRIE и установленном разрешении глобального прерывания установка бита UDRE в регистре USR приведет к выполнению прерывания по пустому регистру данных UART.

• **Bit 4 - RXEN: Receiver Enable - Разрешение приемника**

Установленный в состояние 1 бит RXEN разрешает приемник UART. Если приемник запрещен, то флаги статуса TXC, DOR и FE установить невозможно. Если эти флаги установлены, то очистка бита RXEN не приведет к очистке этих флагов.

• **Bit 3 - TXEN: Transmitter Enable - Разрешение передатчика**

Установленный в состояние 1 бит TXEN разрешает передатчик UART. При запрещении передатчика во время передачи символа, передатчик не будет заблокирован прежде, чем будут полностью переданы символ в сдвиговом регистре плюс любой находящийся в UDR следующий символ.

• **Bit 2 - CHR9: 9 Bit Characters - Режим 9-разрядных символов**

При установленном в состояние 1 бите CHR9 передаются и принимаются 9-разрядные символы плюс стартовый и стоповый биты. Девятые биты читаются и записываются с использованием битов RXB8 и TXB8 (соответственно) регистра UCR. Девятый бит данных может использоваться как дополнительный стоповый бит или бит контроля четности.

• **Bit 1 - RXB8: Receive Data Bit 8 - Прием 8-разрядных данных**

При установленном в состояние 1 бите CHR9 бит RXB8 является девятым битом данных принятого символа.

• **Bit 0 - TXB8: Transmit Data Bit 8 - Передача 8-разрядных данных**

При установленном в состояние 1 бите CHR9 бит TXB8 является девятым битом данных передаваемого символа.

БОД-ГЕНЕРАТОР (Baud Rate Generator)

Бод-генератор представляет собой делитель, генерирующий импульсы передачи с частотой, определяемой выражением:

$$\text{BAUD} = \frac{f_{\text{СК}}}{16(\text{UBRR}+1)}, \quad \text{где:}$$

- BAUD = частота в бодах.
- $f_{\text{СК}}$ = частота кварцевого генератора.
- UBRR = содержимое регистра UBRR (Baud Rate register = 0 - 255).

При использовании стандартных кварцевых кристаллов, наиболее часто используемые скорости передачи в бодах могут быть получены установками UBRR, представленными в Табл. 24. При установках UBRR, указанных в таблице, реальные скорости в бодах будут иметь отличия менее 2% от стандартных скоростей (показано в таблице полужирным шрифтом).

Таблица 24. Установки UBRR при различных стандартных частотах кварцевых кристаллов

Скорость (бод)	1 MHz	Ошиб-ка,%	1,8432 MHz	Ошиб-ка,%	2 MHz	Ошиб-ка,%	2,4576 MHz	Ошиб-ка,%
2400	UBRR= 25	0,2	UBRR= 47	0,0	UBRR= 51	0,2	UBRR= 63	0,0
4800	UBRR= 12	0,2	UBRR= 23	0,0	UBRR= 25	0,2	UBRR= 31	0,0
9600	UBRR= 6	7,5	UBRR= 11	0,0	UBRR= 12	0,2	UBRR= 15	0,0
14400	UBRR= 3	7,8	UBRR= 7	0,0	UBRR= 8	3,7	UBRR= 10	3,1
19200	UBRR= 2	7,8	UBRR= 5	0,0	UBRR= 6	7,5	UBRR= 7	0,0
28800	UBRR= 1	7,8	UBRR= 3	0,0	UBRR= 3	7,8	UBRR= 4	6,3
38400	UBRR= 1	22,9	UBRR= 2	0,0	UBRR= 2	7,8	UBRR= 3	0,0
57600	UBRR= 0	7,8	UBRR= 1	0,0	UBRR= 1	7,8	UBRR= 2	12,5
76800	UBRR= 0	22,9	UBRR= 1	33,3	UBRR= 1	22,9	UBRR= 1	0,0
115200	UBRR= 0	84,3	UBRR= 0	0,0	UBRR= 0	7,8	UBRR= 0	25,0

Скорость (бод)	3,2768 MHz	Ошиб-ка,%	3,6864 MHz	Ошиб-ка,%	4 MHz	Ошиб-ка,%	4,608 MHz	Ошиб-ка,%
2400	UBRR= 84	0,4	UBRR= 95	0,0	UBRR= 103	0,2	UBRR= 119	0,0
4800	UBRR= 42	0,8	UBRR= 47	0,0	UBRR= 51	0,2	UBRR= 59	0,0
9600	UBRR= 20	1,6	UBRR= 23	0,0	UBRR= 25	0,2	UBRR= 29	0,0
14400	UBRR= 13	1,6	UBRR= 15	0,0	UBRR= 16	2,1	UBRR= 19	0,0
19200	UBRR= 10	3,1	UBRR= 11	0,0	UBRR= 12	0,2	UBRR= 14	0,0
28800	UBRR= 6	1,6	UBRR= 7	0,0	UBRR= 8	3,7	UBRR= 9	0,0
38400	UBRR= 4	6,3	UBRR= 5	0,0	UBRR= 6	7,5	UBRR= 7	6,7
57600	UBRR= 3	12,5	UBRR= 3	0,0	UBRR= 3	7,8	UBRR= 4	0,0
76800	UBRR= 2	12,5	UBRR= 2	0,0	UBRR= 2	7,8	UBRR= 3	6,7
115200	UBRR= 1	12,5	UBRR= 1	0,0	UBRR= 1	7,8	UBRR= 2	20,0

Скорость (бод)	7,3728 MHz	Ошиб-ка, %	8 MHz	Ошиб-ка %	9,216 MHz	Ошиб-ка %	11,059 MHz	Ошиб-ка %
2400	UBRR= 191	0,0	UBRR= 207	0,2	UBRR= 239	0,0	UBRR= 287	-
4800	UBRR= 95	0,0	UBRR= 103	0,2	UBRR= 119	0,0	UBRR= 143	0,0
9600	UBRR= 47	0,0	UBRR= 51	0,2	UBRR= 59	0,0	UBRR= 71	0,0
14400	UBRR= 31	0,0	UBRR= 34	0,8	UBRR= 39	0,0	UBRR= 47	0,0
19200	UBRR= 23	0,0	UBRR= 25	0,2	UBRR= 29	0,0	UBRR= 35	0,0
28800	UBRR= 15	0,0	UBRR= 16	2,1	UBRR= 19	0,0	UBRR= 23	0,0
38400	UBRR= 11	0,0	UBRR= 12	0,2	UBRR= 14	0,0	UBRR= 17	0,0
57600	UBRR= 7	0,0	UBRR= 8	3,7	UBRR= 9	0,0	UBRR= 11	0,0
76800	UBRR= 5	0,0	UBRR= 6	7,5	UBRR= 7	6,7	UBRR= 8	0,0
115200	UBRR= 3	0,0	UBRR= 3	7,8	UBRR= 4	0,0	UBRR= 5	0,0

РЕГИСТР БОД-ГЕНЕРАТОРА UART- UBRR - (UART Baud Rate Register)

Бит	7	6	5	4	3	2	1	0	
\$09 (\$29)	MSB							LSB	UBRR
Чтение/Запись	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное значение	0	0	0	0	0	0	0	0	

Регистр UBRR является 8-разрядным регистром, с возможностью чтения/записи, определяющим скорость UART в соответствии с установками Табл. 24.

Аналоговый компаратор

Аналоговый компаратор сравнивает уровни на положительном выводе PE2 (AC+) и отрицательном выводе PE3 (AC-). При напряжении на положительном выводе PE2 (AC+) большем, чем напряжение на отрицательном выводе PE3 (AC-), выход аналогового компаратора АСО устанавливается в состояние 1. Выход компаратора может быть использован для управления входом захвата таймера/счетчика 1. Кроме того, компаратор может формировать свой запрос прерывания. Пользователь может задать формирование запроса на прерывание по нарастающему или падающему фронту или по переключению. Блок-схема аналогового компаратора, со схемами обрaмления, показана на Рис. 45.

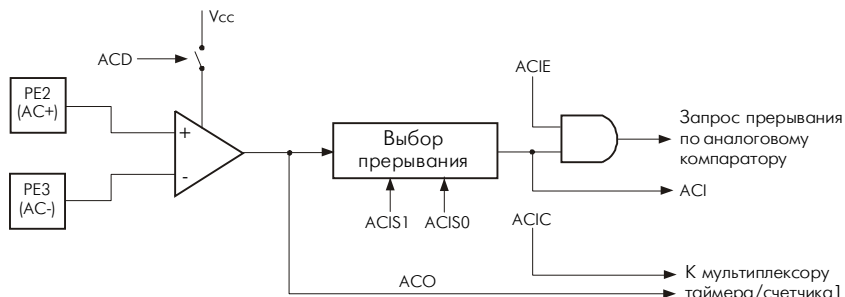


Рис. 45 Блок-схема аналогового компаратора.

**РЕГИСТР СТАТУСА И УПРАВЛЕНИЯ АНАЛОГОВОГО КОМПАРТОРА -
ACSR - (The Analog Comparator Control and Status Register)**

Бит	7	6	5	4	3	2	1	0	
\$08 (\$28)	ACD	-	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0	ACSR
Чтение/Запись	R/W	R	R	R/W	R/W	R/W	R/W	R/W	
Начальное значение	0	0	0	0	0	0	0	0	

• **Bit 7 - ACD: Analog Comparator Disable - Запрет аналогового компаратора**

При установленном в состояние 1 бите ACD аналоговый компаратор запрещен. Для выключения аналогового компаратора установку данного бита можно производить в любое время. Отключение аналогового компаратора позволяет снизить потребление в активном и Idle режимах. При изменении состояния бита ACD необходимо запрещать прерывание по аналоговому компаратору очисткой бита ACIE в регистре ACSR. В противном случае при изменении состояния бита ACD может произойти прерывание.

• **Bit 6 - Res: Reserved bit - Резервированный бит**

Этот бит в микроконтроллерах ATmega603/103 резервирован и при считывании всегда покажет состояние 0.

• **Bit 5 - ACO: Analog Comparator Output - Выход аналогового компаратора**

Бит ACO связан непосредственно с выходом компаратора.

• **Bit 4 - ACI: Analog Comparator Interrupt Flag - Флаг прерывания по аналоговому компаратору**

Данный бит устанавливается в состояние 1 в случае формирования компаратором прерывания, определяемого ACIS1 и ACIS0. Подпрограмма обработки прерывания по аналоговому компаратору будет выполняться при установленном бите ACIE и установленном бите глобального прерывания в регистре SREG. Бит ACI очищается аппаратно при выполнении соответствующего вектора обработки прерывания, Бит ACI можно очистить, также, записью во флаг логической 1.

Отметим однако, что при модификации других битов регистра ACSR командами SBI или CBI бит ACI будет очищен, если он был установлен перед этими операциями.

• **Bit 3 - ACIE: Analog Comparator Interrupt Enable - Разрешение прерывания по аналоговому компаратору**

При установленном бите ACIE и установленном бите глобального прерывания регистра SREG активируется прерывание по аналоговому компаратору. При сброшенном бите ACIE прерывание запрещено.

• **Bit 2 - ACIC: Analog Comparator Input Capture enable - Разрешение входа захвата аналогового компаратора**

Установленный в состояние 1 бит ACIC разрешает срабатывание функции захвата входа таймера/счетчика1 по переключению аналогового компаратора. В этом случае выход аналогового компаратора подсоединяется непосредственно ко входной цепи логики захвата входа, что обеспечивает использование функций подавления шума и выбора вида срабатывания прерывания по захвату входа таймера/счетчика1. При очищенном бите ACIC соединения нет. Для запуска прерывания по захвату входа таймера/счетчика1 бит TICIE1 в регистре масок прерываний TIMSK должен быть установлен в состояние 1.

• **Bits 1,0 - ACIS1, ACIS0: Analog Comparator Interrupt Mode Select - Выбор режима прерывания по аналоговому компаратору**

Эти биты определяют характер события компаратора, при котором запускается

прерывание по аналоговому компаратору. Варианты установок показаны в Таблице 25.

Таблица 25. Установки битов ACIS1/ACIS0

ACIS1	ACIS0	Режим прерывания
0	0	Прерывание по переключению выхода компаратора
0	1	Зарезервировано
1	0	Прерывание по падающему фронту на выходе компаратора
1	1	Прерывание по нарастающему фронту на выходе компаратора

Примечание: При изменении состояния битов ACIS1/ACIS0 прерывание по аналоговому компаратору должно быть запрещено очисткой бита разрешения прерывания в регистре ACSR. В противном случае при изменении состояния битов может произойти прерывание.

Аналого-цифровой преобразователь - (Analog to Digital Converter)

Основные характеристики:

- Разрешение 10 разрядов
- Точность $\pm 1/2$ LSB
- Время преобразования 70...280 мс
- 8 мультиплексируемых каналов входа
- Режимы циклического и однократного преобразования
- Прерывание по завершению ADC преобразования
- Устройство подавления шумов Sleep режима

Микроконтроллеры ATmega603/103 оснащены 10-разрядным ADC последовательного приближения. ADC подсоединен к 8-канальному аналоговому мультиплексору, позволяющему использовать любой вывод порта F в качестве входа ADC. ADC содержит усилитель выборки/хранения, удерживающий напряжение входа ADC во время преобразования на неизменном уровне. Блок-схема ADC представлена на Рис. 46. Для питания ADC используются два отдельных вывода: AVCC и AGND. Вывод AGND должен быть подсоединен к GND и напряжение AVCC не должно отличаться от напряжения VCC более чем на 0,4 В. Способы подключения этих выводов см. в разделе Технология подавления шума ADC.

Внешнее напряжение сравнения подается на вывод AREF и должно быть в диапазоне от 2,7 В до AVCC.

Работа аналого-цифрового преобразователя

Аналого-цифровой преобразователь может работать в двух режимах: режиме однократного преобразования и режиме циклического преобразования. В режиме однократного преобразования каждое преобразование инициируется пользователем. В режиме циклического преобразования ADC осуществляет выборку и обновление содержимого регистра данных ADC непрерывно. Выбор режима производится битом ADFR регистра ADCSR.

Работа ADC разрешается установкой в состояние 1 бита ADEN в регистре ADCSR. Первому преобразованию, начинающемуся после разрешения ADC, предшествует пустое инициализирующее преобразование. На пользователе это отражается лишь тем, что первое преобразование будет занимать 27 тактовых циклов, вместо обычных 14.

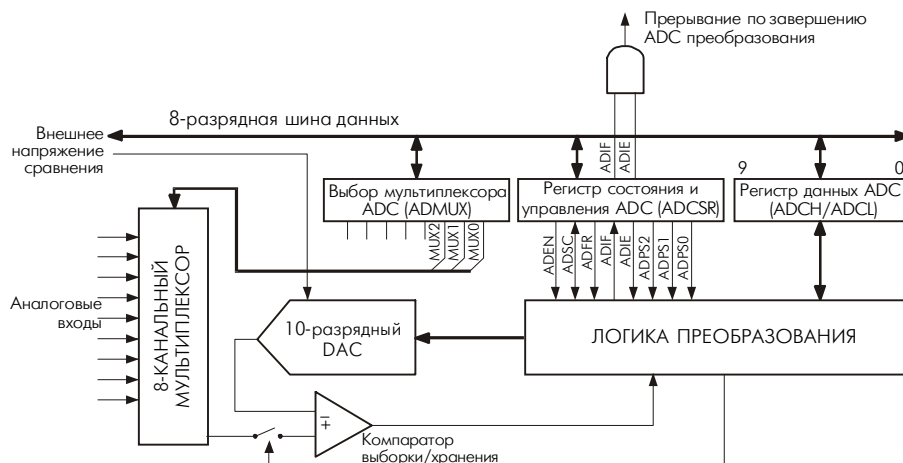


Рис. 46. Блок-схема аналого-цифрового преобразователя.

Преобразование начинается с установки в состояние 1 бита начала преобразования ADSC. Этот бит находится в состоянии 1 в течение всего цикла преобразования и сбрасывается, по завершении преобразования, аппаратно. Если в процессе выполнения преобразования выполняется смена канала данных, то ADC вначале закончит текущее преобразование и лишь потом выполнит переход к другому каналу.

Поскольку ADC формирует 10-разрядный результат, то по завершении преобразования результирующие данные размещаются в двух регистрах данных ADCH и ADCL. Для обеспечения соответствия результирующих данных считываемому уровню используется специальная логика защиты. Этот механизм работает следующим образом: при считывании данных первым должен быть считан регистр ADCL. Как только ADCL считан обращение ADC к регистрам данных блокируется. Таким образом, если после считывания состояния ADCL, но до считывания ADCH, будет завершено следующее преобразование, ни один из регистров не будет обновлен и записанный ранее результат не будет искажен. Обращение ADC к регистрам ADCH и ADCL разрешается по завершении считывания содержимого регистра ADCH.

ADC имеет свое собственное прерывание, которое может быть активировано по завершению преобразования. Когда обращение ADC к регистрам запрещено, в процессе считывания регистров ADCL и ADCH, прерывание будет активироваться, даже если результат будет потерян.

Предварительное деление

ADC работает с тактовой частотой в диапазоне от 50 до 200 кГц. В режиме циклического преобразования для преобразования необходимо 14 тактовых циклов, т.е. преобразование выполняется за время от 70 до 280 мс. В режиме однократного преобразования преобразование выполняется за 15 тактовых циклов. Если тактовая частота выйдет за указанные пределы, то правильность результата не гарантируется. Биты ADPS0 - ADPS2 используются для обеспечения необходимого диапазона тактовой частоты ADC при частоте XTAL свыше 100 кГц.

Функция подавления шума ADC

Функция подавления шума обеспечивает включение Idle режима на время выполнение преобразования, что позволяет снизить шумы, создаваемые ядром CPU. Для реализации этой функции необходимо выполнить следующую процедуру:

1. Отключить ADC очисткой бита ADEN.
2. Включить ADC и одновременно запустить преобразование установкой битов ADEN и ADSC. Таким образом запускается пустое преобразование за которым последует рабочее преобразование.
3. В течение 14 тактовых циклов пустого преобразования ADC ввести Idle режим.
4. Если, прежде чем будет завершено рабочее преобразование, не произойдет другого прерывания, то прерывание ADC активирует MCU и будет выполнена подпрограмма обработки прерывания по завершению преобразования ADC.

РЕГИСТР ВЫБОРА МУЛЬТИПЛЕКСОРА ADC - ADMUX (ADC Multiplexer Select Register)

Бит	7	6	5	4	3	2	1	0	
\$07 (\$27)	-	-	-	-	-	MUX2	MUX1	MUX0	ADMUX
Чтение/Запись	R	R	R	R	R	R/W	R/W	R/W	
Начальное значение	0	0	0	0	0	0	0	0	

• Bits 7..3 - Res: Reserved Bits - Зарезервированные биты

Эти биты в микроконтроллерах ATmega603/103 зарезервированы и при считывании всегда покажут состояние 0.

• Bits 2..0 - MUX2..MUX0: Analog Channel Select Bits 2-0 - Биты выбора аналогового канала

Состояние данных битов определяет какой из восьми аналоговых каналов (0 - 7) будет подключен к ADC.

РЕГИСТР УПРАВЛЕНИЯ И СОСТОЯНИЯ ADC - ADCSR - (ADC Control and Status Register)

Бит	7	6	5	4	3	2	1	0	
\$06 (\$26)	ADEN	ADSC	ADFR	ADIF	ADIE	ADPS2	ADPS1	ADPS0	ADCSR
Чтение/Запись	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное значение	0	0	0	0	0	0	0	0	

• Bit 7 - ADEN: ADC Enable - Разрешение ADC

Установка данного бита в состояние 1 разрешает ADC. Очистка бита запрещает ADC. Запрещение ADC в процессе преобразования прекращает преобразование.

• Bit 6 - ADSC: ADC Start Conversion - Запуск преобразования ADC

В режиме однократного преобразования для запуска каждого цикла преобразования необходимо устанавливать бит ADSC в состояние 1. В циклическом режиме бит ADSC устанавливается в состояние 1 только при запуске первого цикла преобразования. Каждый раз после первой установки бита ADSC, выполненной после разрешения ADC или одновременно с разрешением ADC, будет выполняться пустое преобразование, предшествующее активируемому преобразованию.. Это пустое преобразование активирует ADC.

КТЦ-МК

ADSC будет сохранять состояние 1 в течение всего цикла преобразования и сбрасывается по завершении преобразования. При выполнении пустого преобразования, предшествующего активируемому, бит ADSC остается установленным до завершения активируемого преобразования. Запись 0 в этот бит эффекта не оказывает.

• **Bit 5 - ADFR: ADC Free Run Select - Установка циклического режима работы ADC**

При установленном в состояние 1 бите ADFR ADC будет работать в циклическом режиме. В этом режиме ADC производит выборки и обращения к регистрам непрерывно (одно за другим). Очистка бита приводит к прекращению циклического режима.

• **Bit 4 - ADIF: ADC Interrupt Flag - Флаг прерывания ADC**

Данный бит устанавливается в состояние 1 по завершению преобразования и обновления регистров данных. Прерывание по завершению преобразования ADC выполняется если в состояние 1 установлены бит ADIE и I-бит регистра SREG. Бит ADIF сбрасывается аппаратно при выполнении подпрограммы обработки соответствующего вектора прерывания. Кроме того, бит ADIF может быть очищен записью во флаг логической 1. Этого необходимо остерегаться при чтении-модификации-записи ADCSR, поскольку может быть запрещено отложенное прерывание. Это применимо и в случаях использования команд SBI и CBI.

• **Bit 3 - ADIE: ADC Interrupt Enable - Разрешение прерывания ADC**

При установленных в состояние 1 бите ADIE и I-бите регистра SREG активируется прерывание по завершению преобразования ADC.

• **Bits 2..0 - ADPS2..ADPS0: ADC Prescaler Select Bits - Выбор коэффициента предварительного деления**

Данные биты определяют коэффициент деления частоты XTAL для получения необходимой тактовой частоты ADC.

Таблица 26. Выбор коэффициента предварительного деления

ADPS2	ADPS1	ADPS0	Коэффициент деления
0	0	0	Без деления
0	0	1	2
0	1	0	4
0	1	1	8
1	0	0	16
1	0	1	32
1	1	0	64
1	1	1	128

РЕГИСТРЫ ДАННЫХ ADC - ADCL и ADCH - (ADC Data Register)

Бит	15	14	13	12	11	10	9	8	
\$05 (\$25)	-	-	-	-	-	-	ADC9	ADC8	ADCH
\$04 (\$24)	ADC7	ADC6	ADC5	ADC4	ADC3	ADC2	ADC1	ADC0	ADCL
Чтение/Запись	R	R	R	R	R	R	R	R	
Начальное значение	0	0	0	0	0	0	0	0	

По завершении цикла преобразования результат преобразования размещается в этих двух регистрах. Важно, чтобы в циклическом режиме считывались оба регистра и чтобы регистр ADCL считывался перед считыванием ADCH.

Сканирование аналоговых каналов

Поскольку смена аналоговых каналов происходит после завершения цикла преобразования в циклическом режиме смена каналов (сканирование каналов) может происходить без прерывания преобразователя. Обычно для выполнения смены канала выполняется прерывание по завершению преобразования. Однако пользователю необходимо принять к сведению следующее соображение: прерывание активируется сразу по готовности результата к считыванию. В циклическом режиме следующее преобразование начинается через один тактовый цикл ADC после активации прерывания. Если содержимое ADMUX будет изменено в течение этого одного тактового цикла, то новые установки будут задействованы при начале нового преобразования. Если же изменение состояния ADMUX произойдет позднее этого тактового цикла, то при активированном преобразовании будут использоваться предшествовавшие установки.

Технология подавления шума ADC

Цифровые схемы самого микроконтроллера ATmega603/103 и внешние цифровые схемы генерируют электромагнитное излучение, которое может влиять на точность измерения аналоговых сигналов. Если точность преобразования является определяющей, то можно использовать следующие приемы:

1. Аналоговая часть ATmega603/103 и все аналоговые компоненты устройства должны иметь на печатной плате отдельную аналоговую землю. Эта аналоговая земля должна иметь соединение с цифровой землей в одной точке печатной платы.
2. Проводники аналоговых сигналов необходимо делать как можно короче, стремиться проводить их поверх аналоговой земли и, по возможности, как можно дальше от дорожек высокоскоростных цифровых сигналов.
3. Вывод AVCC микроконтроллеров ATmega603/103 должен подключаться к напряжению питания VCC через RC цепочку, как показано на Рис. 47.
4. Для подавления шума CPU можно использовать функцию подавления шума ADC.
5. Если какие-то выводы PORTF используются в качестве цифровых входов, то важно, чтобы в процессе преобразования на этих выводах не происходили переключения.

Характеристики ADC по постоянному току

Температура окружающей среды TA= от -40°C до 85°C

Обознач.	Параметр	Условия	Min	Тип	Max	Ед. измерения
	Разрешение			10		Биты
	Интегральная нелинейность	VREF > 2V		0,2	0,5	LSB
	Дифференциальная нелинейность	VREF > 2V		0,2	0,5	LSB
	Ошибка нуля (смещение)			1		LSB
	Время преобразования		70		280	мкс
	Тактовая частота		50		200	кГц
AV _{CC}	Аналоговое напряжение питания		V _{CC} -0,3 ⁽¹⁾		V _{CC} +0,3 ⁽²⁾	В
V _{REF}	Напряжение сравнения		AGND		AV _{CC}	В
R _{REF}	Входное сопротивление входа сравнения		6	10	13	кОм
R _{AIN}	Входное сопротивление аналогового входа			100		МОм

Примечания: 1. Минимальное при AVCC=2,7В

2. Максимальное при AVCC=6,0В

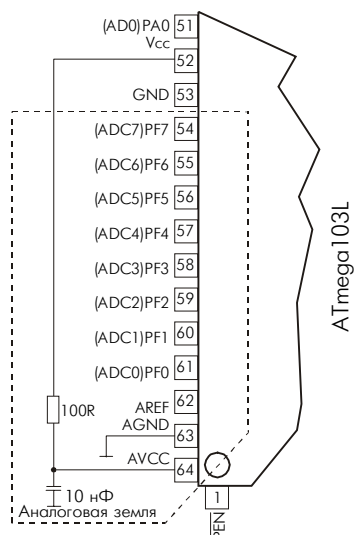


Рис. 47. Подключение напряжения питания ADC

Порты I/O

Порт А

Порт А является 8-разрядным двунаправленным I/O портом и оснащен встроенными нагрузочными резисторами.

Взаимодействие с портом А осуществляется тремя расположенными в пространстве I/O памяти данными регистрами: регистром данных - PORTA, \$1B(\$3B), регистром направления данных - DDRA, \$1A(\$3A) и регистром адресов выводов входа - PINA, \$19(\$39). Регистр адресов выводов входа порта А обеспечивает возможность только чтения, регистры данных и направления данных порта А обеспечивают возможность и чтения и записи. Все выводы порта А оснащены индивидуально подключаемыми встроенными нагрузочными резисторами.

Выходные буферы выводов порта А обеспечивают втекающий ток до 40 мА, что достаточно для прямого управления LED дисплеями. Если выводы с PA0 по PA7 используются в качестве входов и внешним сигналом удерживаются на низком уровне, то вытекающий ток обеспечивается подключением внутренних нагрузочных резисторов. Выводы порта А могут выполнять, дополнительную к основной функции, функцию обеспечения взаимодействия с внешней дополнительной SRAM данных - они могут быть сконфигурированы как младшие разряды шины адреса/данных внешней SRAM данных.

Дополнительная функция включается установкой бита SRE (разрешение внешней SRAM) в регистре управления MCU (MCUCR), при этом установки регистра направления данных игнорируются.

РЕГИСТР ДАННЫХ ПОРТА А - PORTA - (PORT A Data Register)

Бит	7	6	5	4	3	2	1	0	
\$1B (\$3B)	PORTA7	PORTA6	PORTA5	PORTA4	PORTA3	PORTA2	PORTA1	PORTA0	PORTA
Чтение/Запись	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное значение	0	0	0	0	0	0	0	0	

РЕГИСТР НАПРАВЛЕНИЯ ДАННЫХ ПОРТА А - DDRA - (PORT A Data Direction Register)

Бит	7	6	5	4	3	2	1	0	
\$1A (\$3A)	DDA7	DDA6	DDA5	DDA4	DDA3	DDA2	DDA1	DDA0	DDRA
Чтение/Запись	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное значение	0	0	0	0	0	0	0	0	

РЕГИСТР ВЫВОДОВ ВХОДА ПОРТА А - PINA - (PORT A Input Pins Address)

Бит	7	6	5	4	3	2	1	0	
\$19 (\$39)	PINA7	PINA6	PINA5	PINA4	PINA3	PINA2	PINA1	PINA0	PINA
Чтение/Запись	R	R	R	R	R	R	R	R	
Начальное значение	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	

PINA - адрес выводов входа порта А не является регистром в полном смысле этого слова, и эти адреса обеспечивают считывание физического состояния каждого вывода порта. При считывании PORTA считывается состояние фиксаторов данных порта А, а при считывании PINA считываются непосредственно логические состояния выводов.

РАБОТА ПОРТА А В КАЧЕСТВЕ ЦИФРОВОГО I/O ОБЩЕГО НАЗНАЧЕНИЯ

Все 8 битов порта А при их использовании в качестве цифрового I/O работают одинаково.

Таблица 27. Воздействие битов DDAn на характер работы выводов порта А

DDAn	PORTAn	I/O	Нагрузочный резистор	Описание
0	0	Вход	Не подключен	Третье состояние (Hi-Z)
0	1	Вход	Подключен	При низком уровне PAn обеспечивают вытекающий ток.
1	0	Выход	Не подключен	Низкий уровень, двухтактный выход
1	1	Выход	Не подключен	Высокий уровень, двухтактный выход

Примечание: PAn - выходы I/O общего назначения, n=7, 6, ... 1, 0 - номера выводов порта А.

Биты DDAn регистра DDRA определяют направление работы соответствующего вывода. При установленном в состояние 1 бите DDAn вывод PAn конфигурируется как вывод выхода. При очищенном бите DDAn (сброшенном в 0) вывод PAn конфигурируется как вывод входа.

Если бит PORTAn установлен в состояние 1, когда соответствующий вывод сконфигурирован как вход, то нагрузочный MOS резистор активируется (подключается). Для отключения нагрузочного резистора бит PORTAn необходимо очистить или же необходимо сконфигурировать вывод как выход.

СХЕМОТЕХНИКА ПОРТА А

Каждый из выводов порта синхронизирован, однако на рисунке защелка синхронизации не показана.

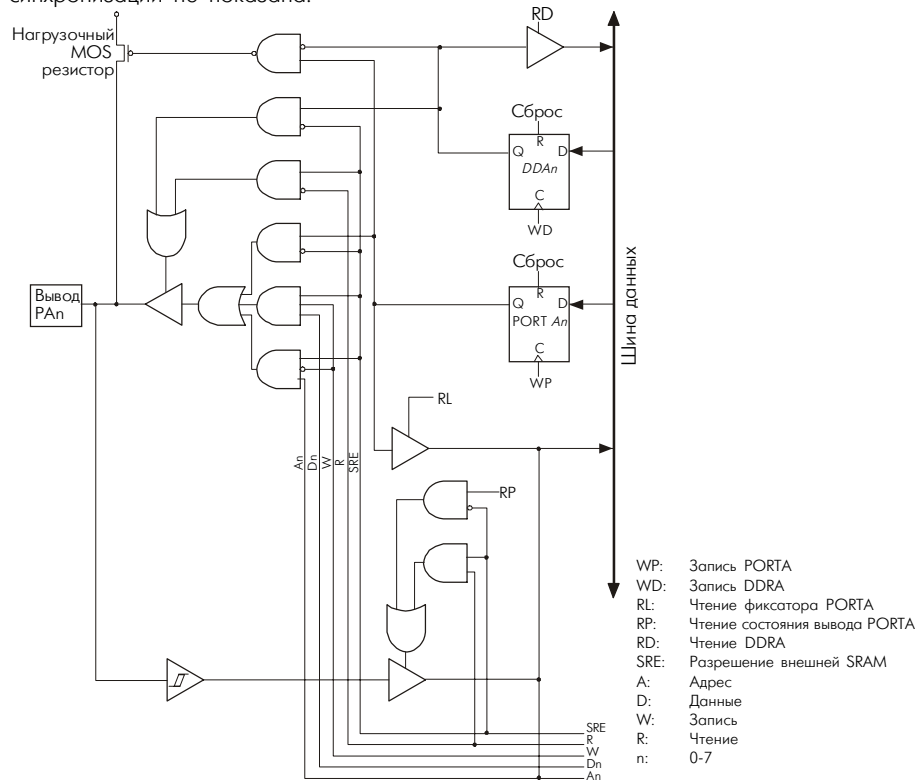


Рис. 48 Схема организации вывода порта А (выводы PA0 - PA7)

Порт В

Порт В является 8-разрядным двунаправленным I/O портом и оснащен встроенными нагрузочными резисторами.

Взаимодействие с портом В осуществляется тремя расположенными в пространстве I/O памяти данными регистрами: регистром данных - PORTB, \$18(\$38), регистром направления данных - DDRB, \$17(\$37) и регистром адресов выводов входа - PINB, \$16(\$36). Регистр адресов выводов входа порта В обеспечивает возможность только чтения, регистры данных и направления данных порта В обеспечивают возможность и чтения и записи. Все выводы порта В оснащены индивидуально подключаемыми встроенными нагрузочными резисторами.

Выходные буферы выводов порта В обеспечивают вытекающий ток до 40 мА, что достаточно для прямого управления LED дисплеями. Если выводы с PB0 по PB7 используются в качестве входов и внешним сигналом удерживаются на низком уровне, то вытекающий ток обеспечивается подключением внутренних нагрузочных резисторов. Выводы порта В могут выполнять, дополнительно к основной функции, функции, представленные в Таблице 29.

Таблица 28. Дополнительные функции выводов порта В

Вывод порта	Дополнительная функция
PB0	Вход выбора ведомого - \overline{SS} (SPI Slave Select input)
PB1	Тактовый сигнал последовательной SPI шины - SCK (SPI Bus Serial Clock)
PB2	Установка Ведущий выход/Ведомый вход SPI шины - MOSI (SPI Bus Master Output/Slave Input)
PB3	Установка Ведущий вход/Ведомый выход SPI шины - MISO (SPI Bus Master Input/Slave Output)
PB4	Сравнение выхода и PWM выход таймера/счетчика0 - OC0A/PWM0A (Output Compare and PWM Output for Timer/Counter0)
PB5	Сравнение выхода и PWM выход А таймера/счетчика1 - OC1A/PWM1A (Output Compare and PWM Output A for Timer/Counter1)
PB6	Сравнение выхода и PWM выход В таймера/счетчика1 - OC1B/PWM1B (Output Compare and PWM Output B for Timer/Counter1)
PB7	Сравнение выхода и PWM выход таймера/счетчика2 - OC2/PWM2 (Output Compare and PWM Output for Timer/Counter2)

Включение выводов для выполнения дополнительных функций производится посредством регистров DDRB и PORTB.

РЕГИСТР ДАННЫХ ПОРТА В - PORTB - (PORT B Data Register)

Бит	7	6	5	4	3	2	1	0	
\$18 (\$38)	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	PORTB
Чтение/Запись	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное значение	0	0	0	0	0	0	0	0	

РЕГИСТР НАПРАВЛЕНИЯ ДАННЫХ ПОРТА В - DDRB - (PORT B Data Direction Register)

Бит	7	6	5	4	3	2	1	0	
\$17 (\$37)	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	DDRB
Чтение/Запись	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное значение	0	0	0	0	0	0	0	0	

РЕГИСТР ВЫВОДОВ ВХОДА ПОРТА В - PINB - (PORT B Input Pins Address)

Бит	7	6	5	4	3	2	1	0	
\$16 (\$36)	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	PINB
Чтение/Запись	R	R	R	R	R	R	R	R	
Начальное значение	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	

PINB - адрес выводов входа порта В не является регистром в полном смысле этого слова, эти адреса обеспечивают считывание физического состояния каждого вывода порта. При считывании PORTB считывается состояние фиксаторов данных порта В, а при считывании PINB считываются непосредственно логические состояния выводов.

РАБОТА ПОРТА В В КАЧЕСТВЕ ЦИФРОВОГО I/O ОБЩЕГО НАЗНАЧЕНИЯ

Все 8 битов порта В при их использовании в качестве цифрового I/O работают одинаково.

Таблица 29 Воздействие битов DDBn на характер работы выводов порта В

DDBn	PORTBn	I/O	Нагрузочный резистор	Описание
0	0	Вход	Не подключен	Третье состояние (Hi-Z)
0	1	Вход	Подключен	При низком уровне Pbn обеспечивают вытекающий ток.
1	0	Выход	Не подключен	Низкий уровень, двухтактный выход
1	1	Выход	Не подключен	Высокий уровень, двухтактный выход

Примечание: Pbn - выводы I/O общего назначения, n=7, 6, ... 1, 0 - номера выводов порта В.

Биты DDBn регистра DDRB определяют направление работы соответствующего вывода. При установленном в состояние 1 бите DDBn вывод Pbn конфигурируется как вывод выхода. При очищенном бите DDBn (сброшенном в 0) вывод Pbn конфигурируется как вывод входа.

Если бит PORTBn установлен в состояние 1, когда соответствующий вывод сконфигурирован как вход, то нагрузочный MOS резистор активируется (подключается). Для отключения нагрузочного резистора бит PORTBn необходимо очистить или же необходимо сконфигурировать вывод как выход.

ДОПОЛНИТЕЛЬНЫЕ ФУНКЦИИ ВЫВОДОВ ПОРТА В

Дополнительные функции выводов порта В следующие:

- **OC2/PWM2, Бит 7**

Вывод сравнения выхода таймера/счетчика2 (OC2) или PWM выход таймера/счетчика2, находящегося в PWM режиме. Для выполнения этих функций вывод должен быть сконфигурирован как выход.

- **OC1B/PWM1B, Бит 6**

Вывод сравнения выхода В таймера/счетчика1 (OC1B) или PWM выход В таймера/счетчика1, находящегося в PWM режиме. Для выполнения этих функций вывод должен быть сконфигурирован как выход.

- **OC1A/PWM1A, Бит 5**

Вывод сравнения выхода А таймера/счетчика1 (OC1A) или PWM выход А таймера/счетчика1, находящегося в PWM режиме. Для выполнения этих функций вывод должен быть сконфигурирован как выход.

- **OC0/PWM0, Бит 4**

Вывод сравнения выхода таймера/счетчика0 (OC0) или PWM выход таймера/счетчика0, находящегося в PWM режиме. Для выполнения этих функций вывод должен быть сконфигурирован как выход.

- **MISO - PORTB, Бит 3**

Определяет вывод SPI канала как вход данных в режиме ведущего или как выход данных в режиме ведомого. При разрешении SPI как ведущего, этот вывод конфигурируется как вход, независимо от установки бита DDB3.

При разрешении SPI как ведомого направление данных на этом выводе управляется битом DDB3 и, если вывод определен как вход, подключение нагрузочного резистора управляется битом PORTB3. Подробности см. в описании SPI порта.

- **MOSI - PORTB, Бит 2**

Вывод SPI канала, определяемый в ведущем режиме SPI как вход данных и как выход данных в ведомом режиме. При SPI разрешенном как ведомый этот бит конфигурируется как вход, независимо от состояния бита DDB2. При разрешении SPI как ведущего направление данных на этом выводе управляется битом DDB2 и, если вывод определен

как вход, подключение нагрузочного резистора управляется битом PORTB2. Подробности см. в описании SPI порта.

• **СК - PORTB, Бит 1**

Выход тактового сигнала в ведущем режиме SPI канала, вход тактового сигнала в ведомом режиме SPI канала. При SPI разрешенном как ведомый этот бит конфигурируется как вход, независимо от состояния бита DDB1. При разрешении SPI как ведущего направление данных на этом выводе управляется битом DDB1 и, если вывод определен как вход, подключение нагрузочного резистора управляется битом PORTB1. Подробности см. в описании SPI порта.

• **SS - PORTB, Бит 0**

Вход выбора порта как ведомого. При SPI разрешенном как ведомый этот бит конфигурируется как вход, независимо от состояния бита DDB0. Как ведомый SPI активируется когда на этот вход подан низкий уровень. При SPI разрешенном в качестве ведущего, направление данных на этом выводе управляется состоянием бита DDB0. Если вывод определен как вход, подключение нагрузочного резистора управляется битом PORTB0. Подробности см. в описании SPI порта.

СХЕМОТЕХНИКА ПОРТА В

Каждый из выводов порта синхронизирован, однако на рисунке защелка синхронизации не показана.

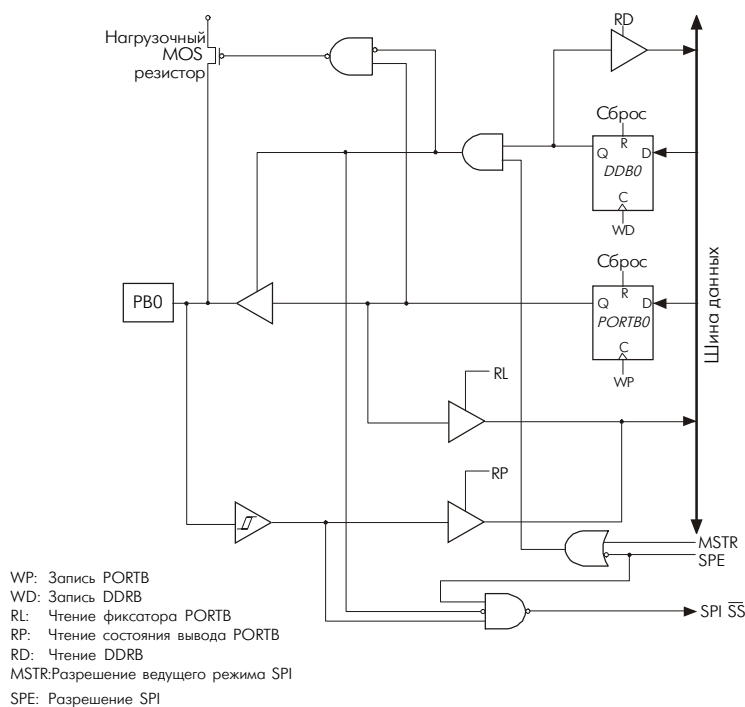


Рис. 49 Схема организации вывода порта В (вывод PB0)

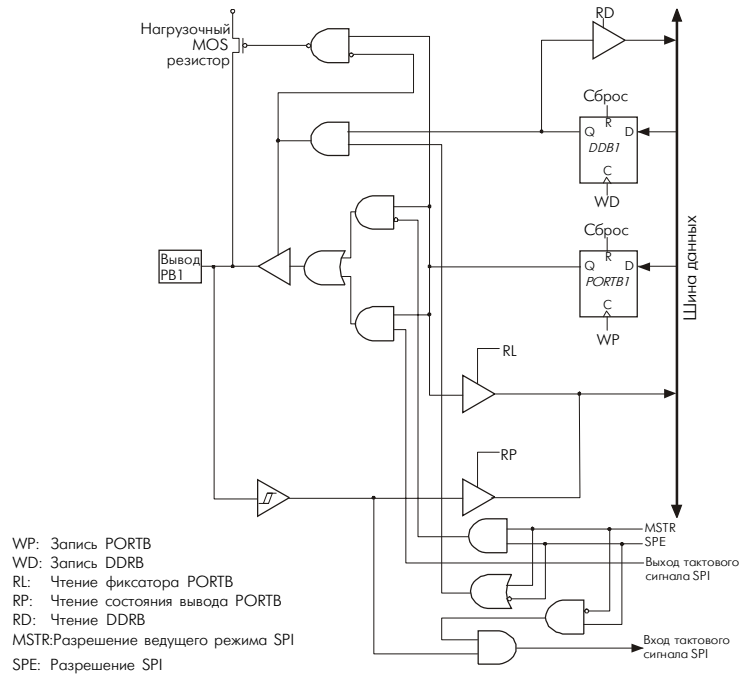


Рис. 50 Схема организации вывода порта В (вывод PB1)

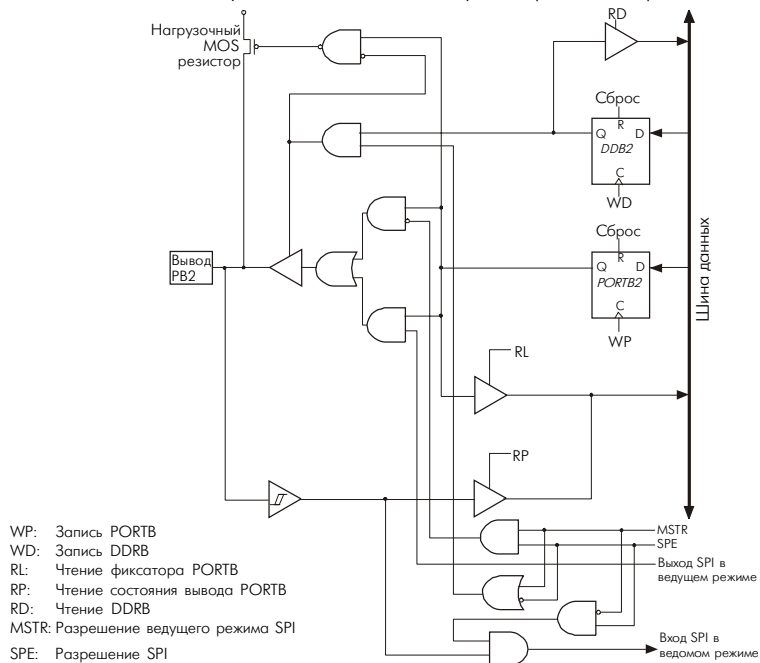


Рис. 51 Схема организации вывода порта В (вывод PB2)

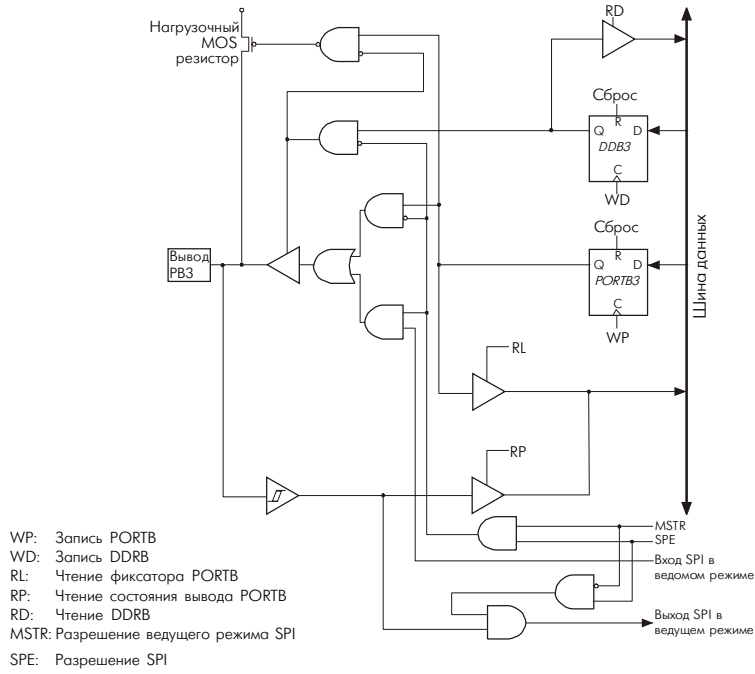


Рис. 52 Схема организации вывода порта В (вывод PB3)

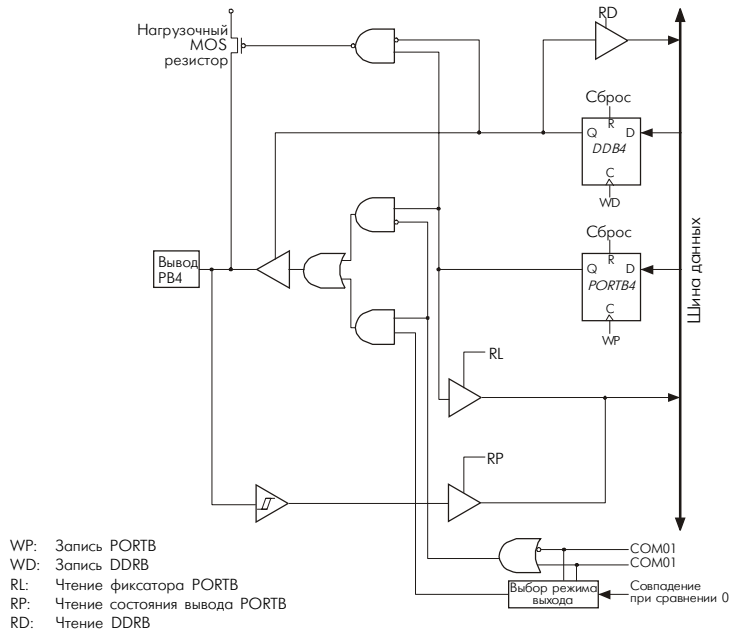


Рис. 53 Схема организации вывода порта В (вывод PB4)

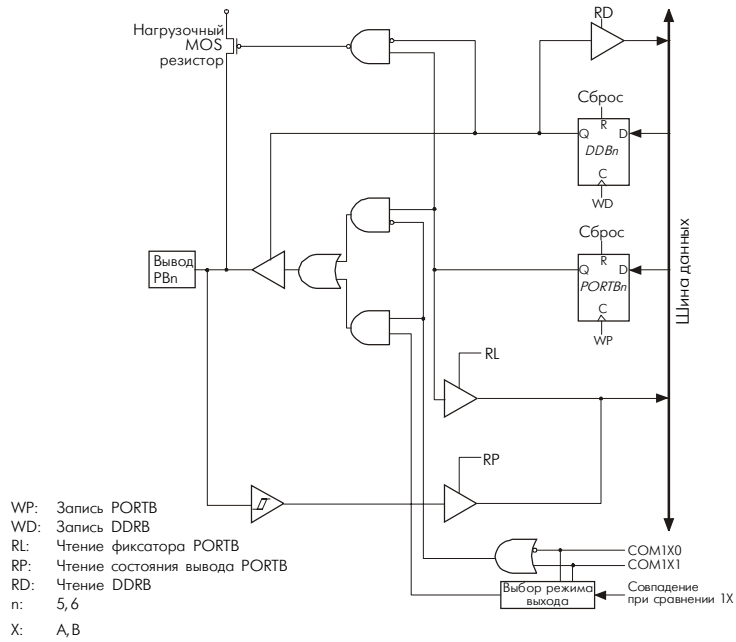


Рис. 54 Схема организации выводов порта В (выводы PB5 и PB6)

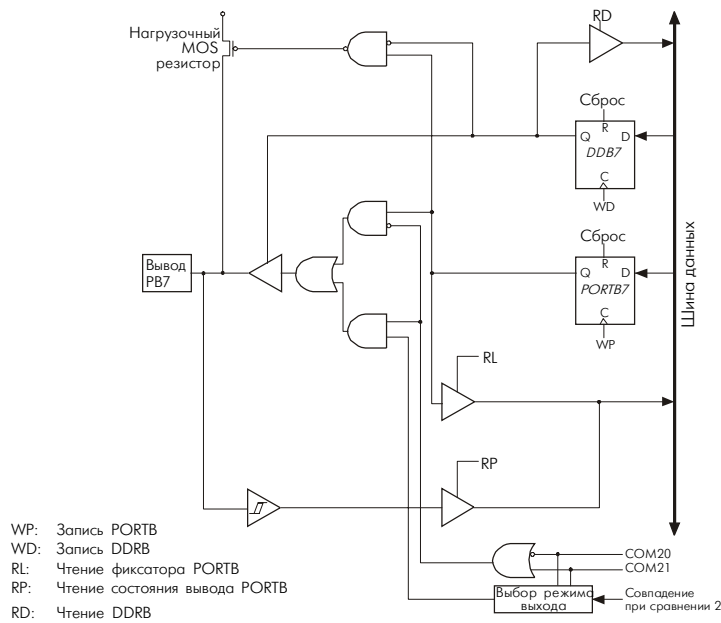


Рис. 55 Схема организации вывода порта В (вывод PB7)

Порт С

Порт С представляет собой 8-разрядный выходной порт. Кроме основной функции выходы порта С выполняют дополнительную функцию обеспечения взаимодействия с внешней дополнительной SRAM. При использовании внешней SRAM через выходы порта С выводится старший байт адреса внешней SRAM.

РЕГИСТР ДАННЫХ ПОРТА С - PORTC

Бит	7	6	5	4	3	2	1	0	
\$15 (\$35)	PORTC7	PORTC6	PORTC5	PORTC4	PORTC3	PORTC2	PORTC1	PORTC0	PORTC
Чтение/Запись	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное значение	0	0	0	0	0	0	0	0	

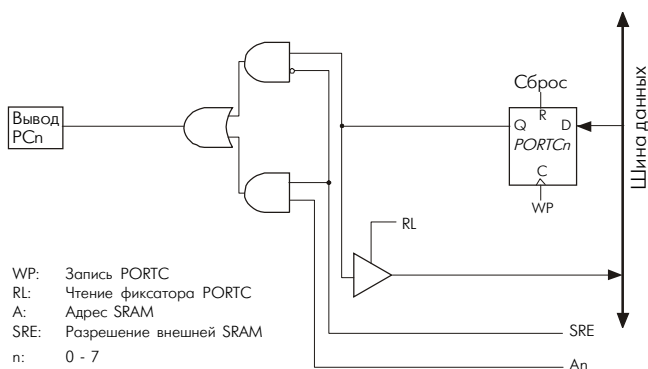


Рис. 56 Схема организации выводов порта С (выводы PC0 - PC7)

Порт D

Порт D является 8-разрядным двунаправленным I/O портом и оснащен встроенными нагрузочными резисторами.

Взаимодействие с портом D осуществляется тремя расположенными в пространстве I/O памяти данными регистрами: регистром данных - PORTD, \$12(\$32), регистром направления данных - DDRD, \$11(\$31) и регистром адресов выводов входа - PIND, \$10(\$30). Регистр адресов выводов входа порта D обеспечивает возможность только чтения, регистры данных и направления данных порта D обеспечивают возможность и чтения и записи.

Выходные буферы выводов порта D обеспечивают вытекающий ток до 40 мА. Если выходы с PD0 по PD7 используются в качестве входов и внешним сигналом удерживаются на низком уровне, то вытекающий ток обеспечивается подключением нагрузочных резисторов. Выводы порта D могут выполнять, дополнительные к основной, представленные в таблице.

При использовании выводов порта для дополнительных функций, их функционирование определяется установками регистров DDRD и PORTD.

Таблица 30. Дополнительные функции выводов порта D

Вывод порта	Дополнительная функция
PD0	Вход внешнего прерывания0 - $\overline{INT0}$ - (External Interrupt0 Input)
PD1	Вход внешнего прерывания0 - $\overline{INT1}$ - (External Interrupt1 Input)
PD2	Вход внешнего прерывания0 - $\overline{INT2}$ - (External Interrupt2 Input)
PD3	Вход внешнего прерывания0 - $\overline{INT3}$ - (External Interrupt3 Input)
PD4	Вход триггера захвата таймера/счетчика1 - IC1 - (Timer/Counter1 Input Capture Trigger)
PD6	Вход тактового сигнала таймера/счетчика1 - T1 - (Timer/Counter1 Clock Input)
PD7	Вход тактового сигнала таймера/счетчика2 - T2 - (Timer/Counter2 Clock Input)

РЕГИСТР ДАННЫХ ПОРТА D - PORTD - (PORT D Data Register)

Бит	7	6	5	4	3	2	1	0	
\$12 (\$32)	PORTD7 PORTD6 PORTD5 PORTD4 PORTD3 PORTD2 PORTD1 PORTD0								PORTD
Чтение/Запись	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное значение	0	0	0	0	0	0	0	0	

РЕГИСТР НАПРАВЛЕНИЯ ДАННЫХ ПОРТА D - DDRD - (PORT D Data Direction Register)

Бит	7	6	5	4	3	2	1	0	
\$11 (\$31)	DDD7	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	DDRD
Чтение/Запись	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное значение	0	0	0	0	0	0	0	0	

РЕГИСТР АДРЕСА ВЫВОДОВ ВХОДА ПОРТА D - PIND - (PORT D Input Pins Address)

Бит	7	6	5	4	3	2	1	0	
\$10 (\$30)	PIND7	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	PIND
Чтение/Запись	R	R	R	R	R	R	R	R	
Начальное значение	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	

PIND - адрес выводов входа порта D не является регистром в полном смысле этого слова, и эти адреса обеспечивают считывание физического состояния каждого вывода порта. При считывании PORTD считывается состояние фиксаторов данных порта D, а при считывании PIND считываются непосредственно логические состояния выводов.

РАБОТА ПОРТА D В КАЧЕСТВЕ ЦИФРОВОГО I/O ОБЩЕГО НАЗНАЧЕНИЯ

Выводы порта PDn являются выводами I/O общего назначения. Состояния битов DDDn регистра DDRD определяют направление работы этих выводов. При установленном в состояние 1 бите DDDn вывод PDn конфигурируется как вывод выхода, сброс бита DDDn в состояние 0 конфигурирует вывод PDn как вывод входа. При установке вывода PDn в состояние 1, если он сконфигурирован как вход, активируется MOS нагрузочный резистор. Для отключения нагрузочного резистора вывод PDn должен быть очищен (сброшен в состояние 0) или же должен быть сконфигурирован как вывод выхода.

Таблица 31. Влияние состояния битов DDDn на выводы порта D

DDDn	PORTDn	I/O	Нагрузочный резистор	Описание
0	0	Вход	Отключен	Третье состояние (Hi-Z)
0	1	Вход	Подключен	При низком уровне PDn обеспечивают вытекающий ток
1	0	Выход	Отключен	Низкий уровень, двухтактный выход
1	1	Выход	Отключен	Высокий уровень, двухтактный выход

ДОПОЛНИТЕЛЬНЫЕ ФУНКЦИИ ВЫВОДОВ ПОРТА D

INT0 .. INT3 - PORTD, Биты 0..3

Выводы внешних прерываний с 0 по 3. Выводы PD0 - PD3 могут быть использованы как внешние активные низким уровнем источники прерываний MCU. Активация встроенных нагрузочных MOS резисторов описана выше. Разрешение источников прерываний и другие подробности описаны в разделе посвященном прерываниям.

IC1 - PORTD, Бит 4

Вывод захвата входа таймера/счетчика1. При поступлении на вывод нарастающего или падающего фронта (зависит от установки) содержимое таймера/счетчика1 пересылается в регистр захвата входа таймера/счетчика1. Для обеспечения реализации данной функции вывод должен быть сконфигурирован как вход (DDD4 должен быть сброшен в состояние 0). Более подробно данная функция описана в описании таймера/счетчика1. Активация встроенного нагрузочного MOS резистора описана выше.

T1 - PORTD, Бит 6

T1 - вывод таймера/счетчика1. Подробности в описании таймера/счетчика1.

T2 - PORTD, Бит 7

T2 - вывод таймера/счетчика2. Подробности в описании таймера/счетчика2.

СХЕМОТЕХНИКА ПОРТА D

Каждый из выводов порта синхронизирован, однако на рисунке защелка синхронизации не показана.

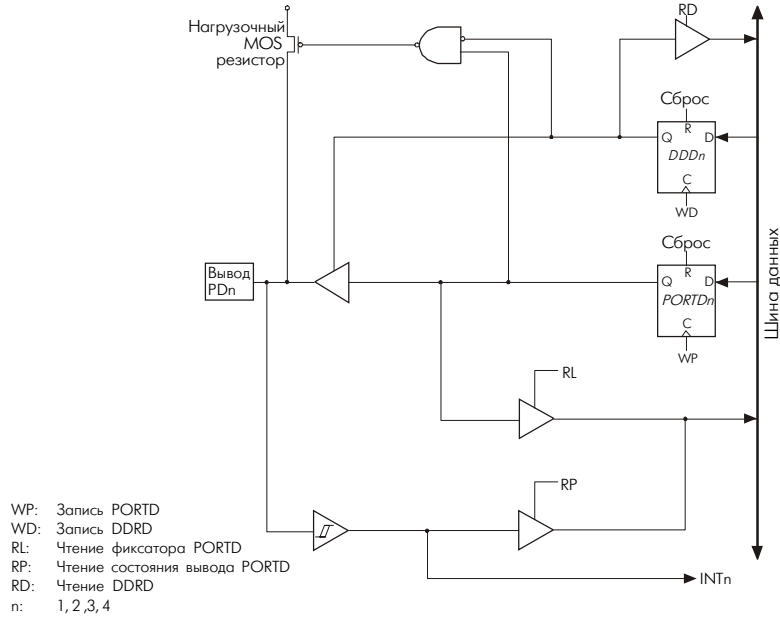


Рис. 57 Схема организации выводов порта D (выводы PD0, PD1, PD2 и PD3)

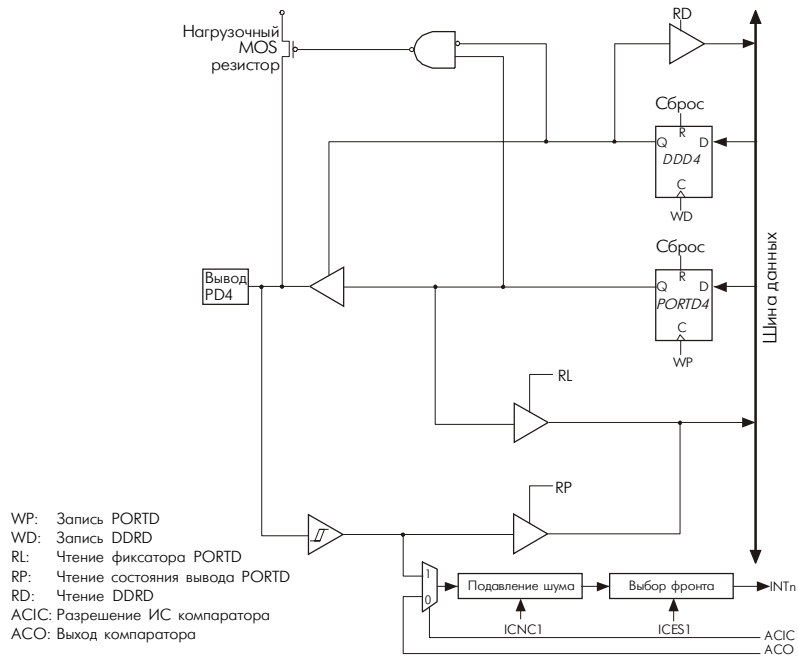


Рис. 58 Схема организации вывода порта D (вывод PD4)

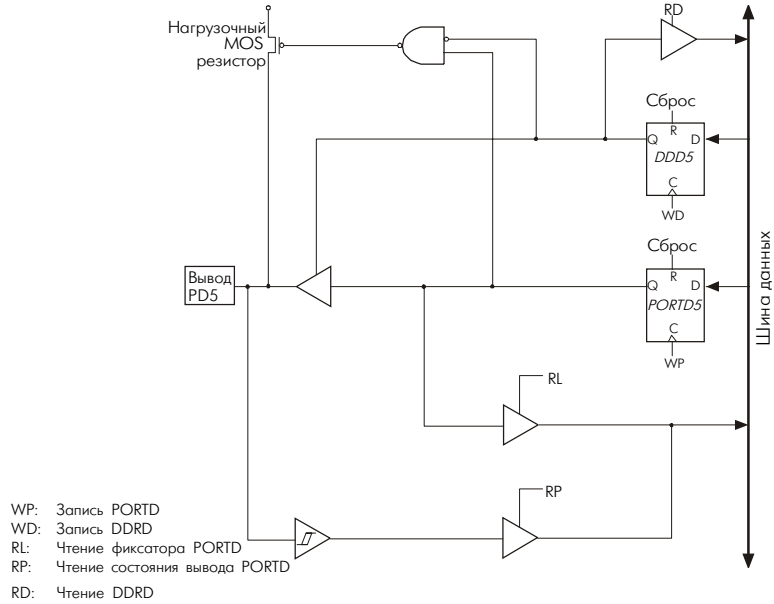


Рис. 59 Схема организации вывода порта D (вывод PD5)

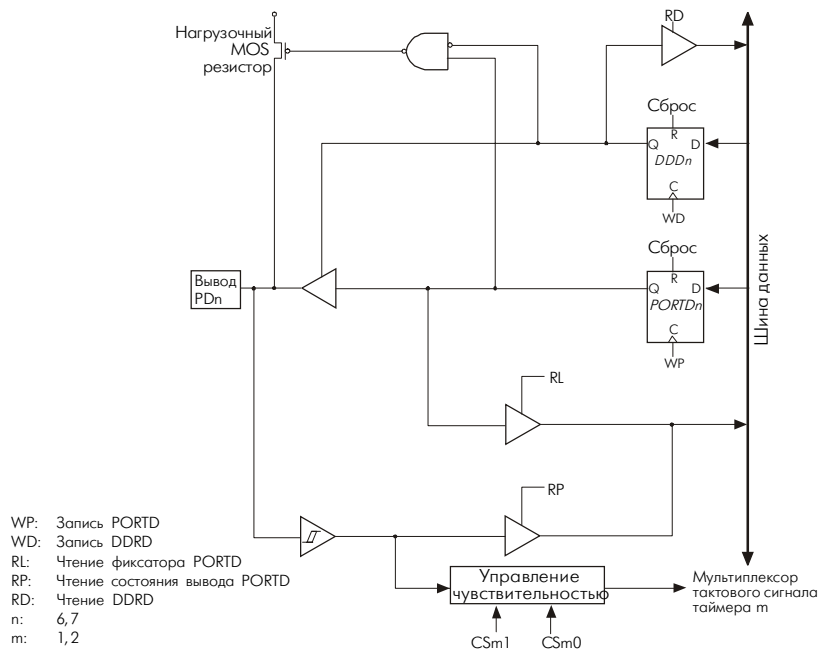


Рис. 60 Схема организации выводов порта D (выводы PD6 и PD7)

Порт E

Порт E является 8-разрядным двунаправленным I/O портом и оснащен встроенными нагрузочными резисторами.

Взаимодействие с портом E осуществляется тремя расположенными в пространстве I/O памяти данными регистрами: регистра данных - PORTE, \$03(\$23), регистра направления данных - DDRE, \$02(\$22) и регистра адресов выводов входа - PINE, \$01(\$21). Регистр адресов выводов входа порта E обеспечивает возможность только чтения, регистры данных и направления данных порта E обеспечивают возможность и чтения и записи.

Выходные буферы выводов порта D обеспечивают протекающий ток до 40 мА. Если выводы с PE0 по PE7 используются в качестве входов и внешним сигналом удерживаются на низком уровне, то протекающий ток обеспечивается подключением нагрузочных резисторов. Выводы порта E могут выполнять, дополнительные к основным функции, представленные в таблице.

Таблица 32. Дополнительные функции выводов порта E

Вывод порта	Дополнительная функция
PE0	Вход программируемых данных или прием UART (PDI/RXD)
PE1	Выход программируемых данных или передача UART (PDO/TXD)
PE2	Положительный вход аналогового компаратора (AC+)
PE3	Отрицательный вход аналогового компаратора (AC-)
PE4	Вход внешнего прерывания (INT4)
PE5	Вход внешнего прерывания (INT5)
PE6	Вход внешнего прерывания (INT6)
PE7	Вход внешнего прерывания (INT7)

При использовании выводов порта для дополнительных функций, их функционирование определяется установками регистров DDRE и PORTE.

РЕГИСТР ДАННЫХ ПОРТА E - PORTE - (PORT E Data Register)

Бит	7	6	5	4	3	2	1	0	
\$03 (\$23)	PORTE7	PORTE6	PORTE5	PORTE4	PORTE3	PORTE2	PORTE1	PORTE0	PORTE
Чтение/Запись	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное значение	0	0	0	0	0	0	0	0	

РЕГИСТР НАПРАВЛЕНИЯ ДАННЫХ ПОРТА E - DDRE - (PORT E Data Direction Register)

Бит	7	6	5	4	3	2	1	0	
\$02 (\$22)	DDE7	DDE6	DDE5	DDE4	DDE3	DDE2	DDE1	DDE0	DDRE
Чтение/Запись	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Начальное значение	0	0	0	0	0	0	0	0	

РЕГИСТР АДРЕСА ВЫВОДОВ ВХОДА ПОРТА E - PINE - (PORT E Input Pins Address)

Бит	7	6	5	4	3	2	1	0	
\$01 (\$21)	PINE7	PINE6	PINE5	PINE4	PINE3	PINE2	PINE1	PINE0	PINE
Чтение/Запись	R	R	R	R	R	R	R	R	R
Начальное значение	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	

PINЕ - адрес выводов входа порта E не является регистром в полном смысле этого слова, эти адреса обеспечивают считывание физического состояния каждого вывода порта. При считывании PORTE считывается состояние фиксаторов данных порта E, а при считывании PINE считываются непосредственно логические состояния выводов.

РАБОТА ПОРТА E В КАЧЕСТВЕ ЦИФРОВОГО I/O ОБЩЕГО НАЗНАЧЕНИЯ

Выводы порта PE_n являются выводами I/O общего назначения. Состояния битов DDE_n регистра DDRE определяют направление работы этих выводов. При установленном в состояние 1 бите DDE_n вывод PE_n конфигурируется как вывод выхода, сброс бита DDE_n в состояние 0 конфигурирует вывод PE_n как вывод входа. При установке вывода в PE_n состояние 1, если он сконфигурирован как вход, активируется MOS нагрузочный резистор. Для отключения нагрузочного резистора вывод PE_n должен быть очищен (сброшен в состояние 0) или же должен быть сконфигурирован как вывод выхода.

Таблица 33. Влияние состояния битов DDE_n на выводы порта E

DDE _n	PORTE _n	I/O	Нагрузочный резистор	Описание
0	0	Вход	Отключен	Третье состояние (Hi-Z)
0	1	Вход	Подключен	При низком уровне PD _n обеспечивают вытекающий ток
1	0	Выход	Отключен	Низкий уровень, двухтактный выход
1	1	Выход	Отключен	Высокий уровень, двухтактный выход

n: 7, 6 ... 0 - номера выводов порта E.

ДОПОЛНИТЕЛЬНЫЕ ФУНКЦИИ ВЫВОДОВ ПОРТА E

PDI/RXD - PORTE, Бит 0

PDI - вход последовательных программируемых данных. В процессе последовательной загрузки данных программы этот вывод используется для ввода данных в ATmega603/103.

RXD - вход приема UART.

PDO/TXD - PORTE, Бит 1

PDO выход последовательных программируемых данных. В процессе последовательной загрузки данных программы этот вывод используется для вывода данных ATmega603/103.

TXD - выход передатчика UART.

AC+ - PORTE, Бит 3

AC+ - положительный вход аналогового компаратора. Данный вывод соединен непосредственно с положительным входом аналогового компаратора.

AC- - PORTE, Бит 4

AC- - отрицательный вход аналогового компаратора. Данный вывод соединен непосредственно с отрицательным входом аналогового компаратора.

INT4 .. INT7 - PORTD, Биты с 4 по 7

INT4 .. INT7 - источники внешних прерываний с 4 по 7. Выводы с 4 по 7 могут использоваться как источники внешних прерываний MCU. Прерывания на этих выводах могут запускаться по низкому уровню на входе или по нарастающему или падающему фронту сигнала. Активация встроенных нагрузочных MOS резисторов описана выше. Способы разрешения источников прерываний и детализация прерываний приведены в описании прерываний.

СХЕМОТЕХНИКА ПОРТА E

Каждый из выводов порта синхронизирован, однако на рисунке защелка синхронизации не показана.

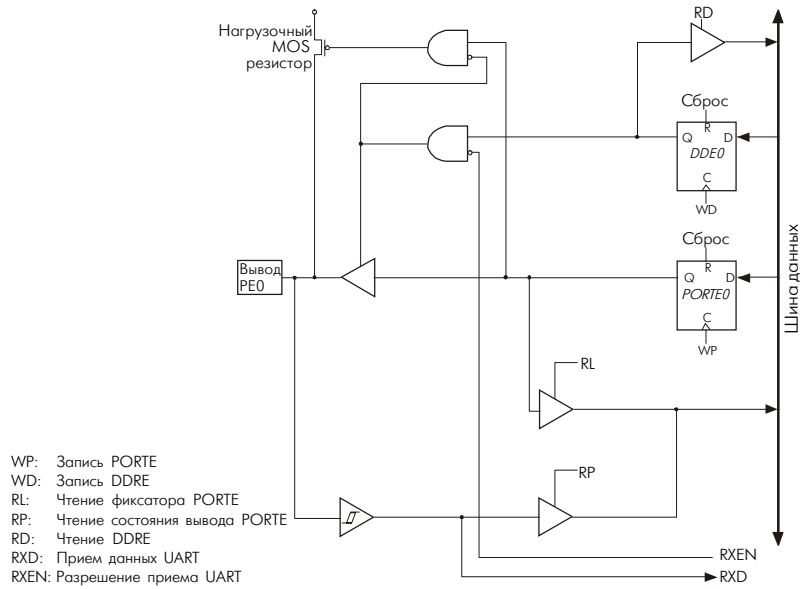


Рис. 61 Схема организации вывода порта E (вывод PE0)

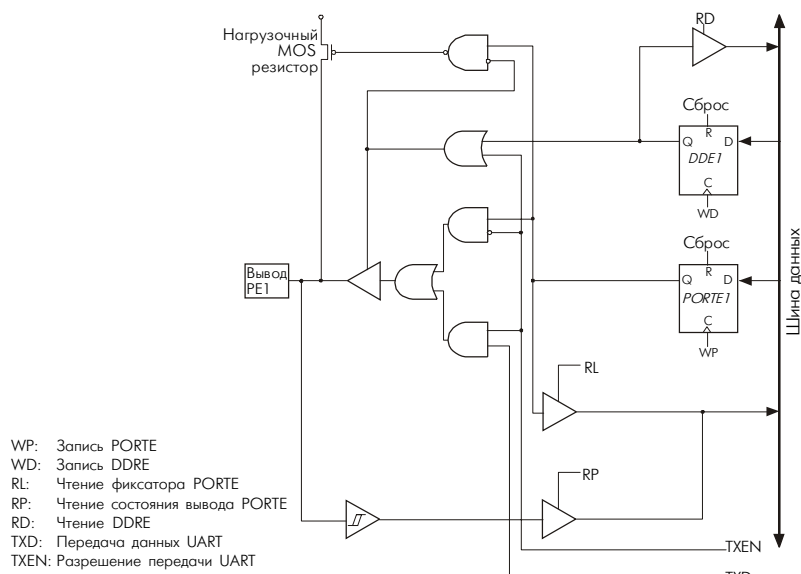


Рис. 62 Схема организации вывода порта E (вывод PE1)

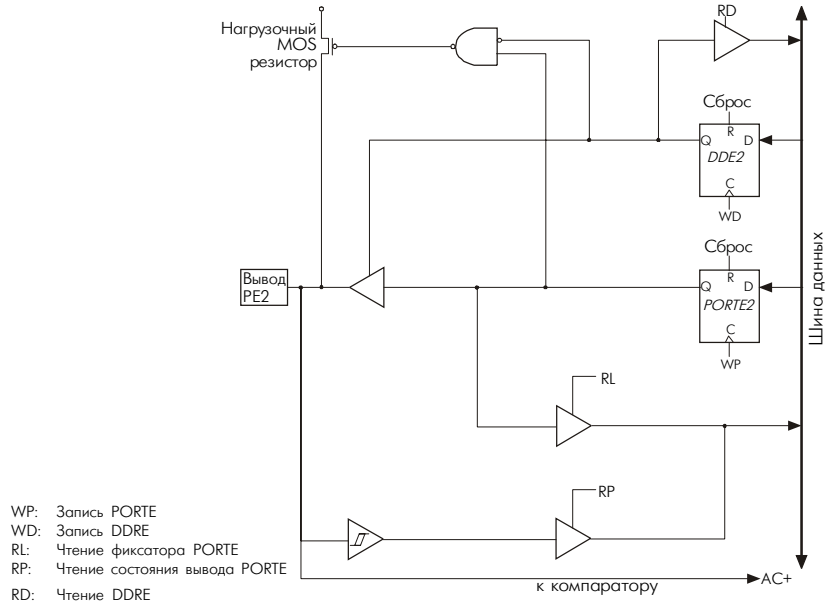


Рис. 63 Схема организации вывода порта E (вывод PE2)

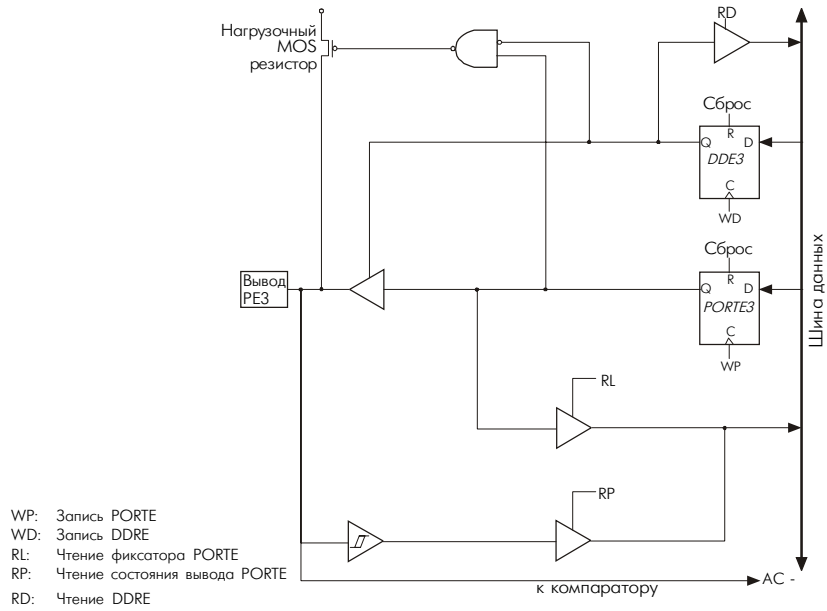


Рис. 64 Схема организации вывода порта E (вывод PE3)

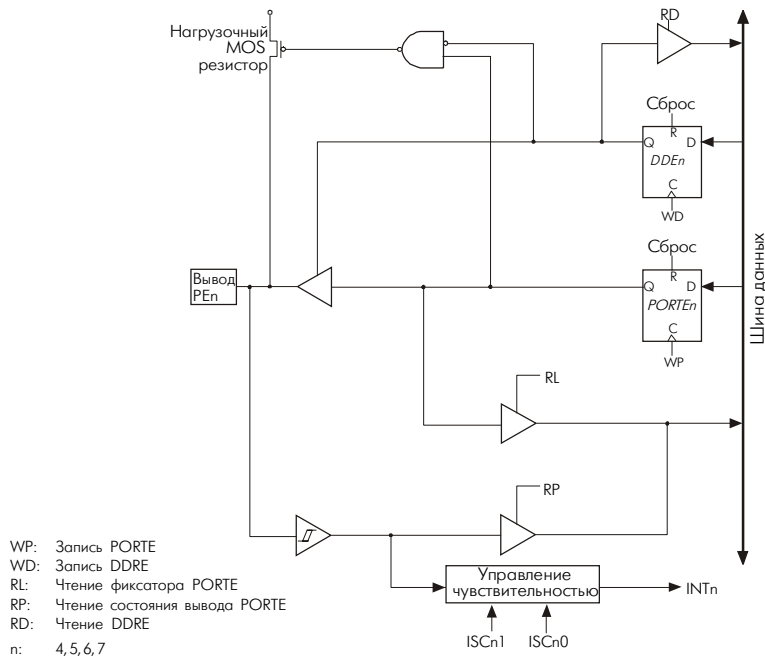


Рис. 65 Схема организации выводов порта E (выводы с PE4 по PE7)

Порт F

Порт F является 8-разрядным портом. В пространстве памяти I/O этому порту соответствует только PINF, \$00(\$20) - выводы входа порта F. Все входы порта F соединены с аналоговым мультиплексором присоединенным, в свою очередь, к аналого-цифровому преобразователю. Выводы порта F, кроме выполнения функций входов мультиплексора, могут быть использованы и в качестве цифровых входов, что позволяет пользователю в одно и то же время использовать часть выводов порта F в качестве цифровых входов и оставшуюся часть в качестве аналоговых входов.

АДРЕСА ВЫВОДОВ ВХОДОВ ПОРТА F - PINF- (PORT F Input Pins Address)

Бит	7	6	5	4	3	2	1	0	
\$00 (\$20)	PINF7	PINF6	PINF5	PINF4	PINF3	PINF2	PINF1	PINF0	PINF
Чтение/Запись	R	R	R	R	R	R	R	R	
Начальное значение	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	Hi-Z	

PINF - адрес выводов входа порта E не является регистром в полном смысле этого слова, эти адреса обеспечивают считывание физического состояния каждого вывода порта.

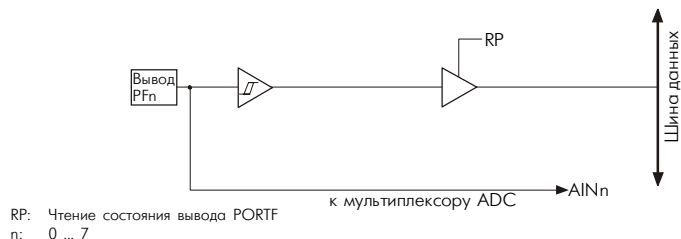


Рис. 66 Схема организации выводов порта F (выводы с PFO по PF7)

Программирование памяти

Биты блокирования программирования памяти

MCU микроконтроллеров ATmega603/103 оснащено двумя битами которые могут быть оставлены незапрограммированными (в состоянии 1) или запрограммированы (состояние 0). Влияние состояний битов на работу микроконтроллеров показано в Таблице 34.

Таблица 34. Режимы защиты битов блокировки

Биты блокирования программирования			Тип защиты
Режим	LB1	LB2	
1	1	1	Нет блокирования программирования
2	0	1	Программирование Flash и EEPROM запрещено
3	0	0	То же, что и режим 2, но запрещена и проверка.

Примечание: Биты блокировки, при выполнении операции очистки кристалла, могут быть только стерты (состояние 1)

Биты-предохранители

Микроконтроллеры ATmega603/103 оснащены четырьмя битами-предохранителями SPIEN, SUT1, SUT0 и EESAVE. При запрограммированном в состояние 0 бите SPIEN разрешается последовательная загрузка программы. По умолчанию бит SPIEN находится в состоянии 0, в режиме последовательного программирования он недоступен и при выполнении операции очистки кристалла его состояние не меняется. Биты-предохранители SUT1 и SUT0 определяют длительность цикла запуска MCU. Подробности см. в Таблице 6. По умолчанию эти биты не запрограммированы (состояние битов 11) и задают длительность цикла запуска в 16 мс.

При запрограммированном бите-предохранителе EESAVE память EEPROM предварительно обрабатывается циклом очистки кристалла. По умолчанию бит-предохранитель EESAVE не запрограммирован (состояние 1), он не может быть запрограммирован если хотя бы один бит блокировки запрограммирован.

Байты сигнатуры (коды идентификации)

Все микроконтроллеры фирмы Atmel оснащены тремя байтами кода сигнатуры, позволяющими идентифицировать прибор. Этот код может быть считан и в последовательном и в параллельном режимах.

Эти три байта расположены в отдельном адресном пространстве.

Для микроконтроллера ATmega603 это:

1. \$00: \$1E (показывает что прибор изготовлен фирмой Atmel)
2. \$01: \$06 (показывает что прибор оснащен 64 Кбайтами Flash памяти)
3. \$02: \$01 (если по адресу \$01 находится содержимое \$06, то это микроконтроллер ATmega603)

Для микроконтроллера Atmega103 это:

1. \$00: \$1E (показывает что прибор изготовлен фирмой Atmel)
2. \$01: \$01 (показывает что прибор оснащен 128Кбайтами Flash памяти)
3. \$02: \$01 (если по адресу \$01 находится содержимое \$01, то это микроконтроллер Atmega103)

Программирование Flash и EEPROM памяти

Микроконтроллеры ATmega603/103 оснащены внутрисистемно программируемой Flash памятью, емкостью 64/128 Кбайт, и 2/4 Кбайтами EEPROM памяти данных. При поставке микроконтроллеров и встроенная Flash память программ и EEPROM память данных находятся в очищенном состоянии (т.е. содержимое в состоянии \$FF) и они готовы к программированию. Приборы поддерживают режим высоковольтного (12 В) параллельного программирования и режим низковольтного последовательного программирования. Напряжение программирования 12 В используется только если программирование разрешено, в ином случае ток по этому выводу не потребляется. Режим последовательного программирования является обычным способом загрузки программ и данных в микроконтроллеры, находящиеся непосредственно в системе пользователя.

Матрица памяти программ микроконтроллеров ATmega603/103 организована из 256/512 страниц по 256 байт каждая. При программировании Flash памяти данные программы фиксируются в буфере страницы, что позволяет программировать сразу целую страницу данных программы в любом из режимов программирования.

Матрица EEPROM памяти данных микроконтроллеров программируется по-байтово (байт - за - байтом) во всех режимах программирования. В последовательном режиме программирования встроенная функция самотактирования EEPROM выполняет автоматическую предварительную очистку каждого программируемого байта.

Режим параллельного программирования

В данном разделе описывается программирование и проверка Flash памяти программ, EEPROM памяти данных, битов блокирования программирования памяти и битов-предохранителей в режиме параллельного программирования. Используемые при этом импульсы должны быть длительностью не менее 500 нс, если это специально не оговорено.

Обозначения сигналов

В данном разделе ряд выводов микроконтроллеров ATmega603/103 указывается обозначениями сигналов, отражающими их функциональное назначение в процессе параллельного программирования, а не по обозначениям выводов. Выводы не указанные в таблице имеют обычные обозначения.

Таблица 35. Соответствие обозначений выводов обозначениям сигналов

Обозначение сигнала в режиме программирования	Обозначение вывода	I/O	Описание
RDY / BSY	PD1	O	0: Прибор занят программированием, 1: Прибор готов к новой команде
\overline{OE}	PD2	I	Разрешение выхода (Активен низким уровнем)
\overline{WR}	PD3	I	Импульс записи (Активен низким уровнем)
BS1	PD4	I	Выбор байта (бит 0)
XA0	PD5	I	Режим XTAL (бит 0)
XA1	PD6	I	Режим XTAL (бит 1)
BS2	PD7	I	Выбор байта (бит 2 -всегда на низком уровне)
PAGEL	PA0	I	Загрузка страницы программирования памяти

Биты XA1/XA0 определяют действие, запускаемое по положительному импульсу на XTAL1. Установки битов представлены в следующей таблице:

Таблица 36. Кодирование битов XA1 и XA0

XA1	XA0	Характер действия при поступлении импульса на XTAL1
0	0	Загрузка Flash или EEPROM адреса (старший или младший байт адреса Flash памяти определяет бит BS1)
0	1	Загрузка данных (старший или младший байт адреса Flash памяти определяет бит BS1)
1	0	Загрузка команды
1	1	Нет действия, ожидание

При поступлении импульсов \overline{WR} или \overline{OE} загруженная команда определяет действие на входе или выходе. Команда является байтом, в котором каждый бит определяет функцию, как это отражено в таблице:

Таблица 37. Кодирование битов байта команды

N бита	Выполняемая операция при установленном бите
7	Очистка кристалла
6	Запись бита-предохранителя. Размещаются в байте данных по следующим битам: D5: SPIEN бит-предохранитель, D3: EESAVE бит-предохранитель, D1: SUT1 бит-предохранитель, D0: SUT0 бит-предохранитель (Примечание: запись 0 для программирования, запись 1 для стирания)
5	Запись бита блокирования. Размещаются в байте данных по следующим битам: D2: LB2, D1: LB1 (Примечание: запись 0 для программирования)
4	Запись Flash или EEPROM памяти (определяется битом 0)
3	Чтение сигнатуры
2	Чтение битов блокирования и битов-предохранителей. Размещаются в байте данных по следующим битам: D5: SPIEN бит-предохранитель, D3: EESAVE бит-предохранитель, D2: LB2, D1: SUT1/LB1, D0: SUT0 (Примечание: состояние 0 программируется)
1	Чтение из Flash или EEPROM памяти (определяется битом 0)
0	0 : Обращение к Flash, 1 : Обращение к EEPROM

Установка режима программирования

Для установки микроконтроллера в режим параллельного программирования используется следующий алгоритм:

1. Подать напряжение 4,5 - 5,5 В между выводами VCC и GND.
2. Установить на выводах $\overline{\text{RESET}}$ и BS низкий уровень и ожидать не менее 100 нс.
3. Подать напряжение 11,5 - 12,5 В на вывод $\overline{\text{RESET}}$. Состояние вывода BS1 в течение 100 нс после подачи напряжения +12 В меняться не должно, иначе режим программирования установлен не будет.

Очистка кристалла

Операция очистки кристалла очищает Flash и EEPROM память и биты блокирования. Биты блокирования не сбрасываются до тех пор, пока память программ не будет полностью очищена. Состояние битов-предохранителей при очистке кристалла остаются неизменным. Очистка кристалла должна выполняться перед программированием Flash памяти.

Загрузка команды очистки кристалла (Chip Erase)

1. Установить биты XA1 и XA0 в состояние 1 и 0. Эта установка разрешает загрузку команды.
2. Установить бит BS1 в состояние 0.
3. Установить биты PB(7 - 0) в состояние 1000 0000. Это команда очистки кристалла.
4. Подать на вывод XTAL1 положительный импульс. Этим импульсом загружается команда и запускается процедура очистки матриц Flash и EEPROM памяти. После подачи импульса на XTAL1 на $\overline{\text{WR}}$ подается отрицательный импульс, чтобы обеспечить в конце цикла стирания памяти стирание бита блокировки, и ожидать по крайней мере 10 мс. Процедура очистки кристалла не приводит к какой либо активности вывода RDY/ $\overline{\text{BSY}}$.

Программирование Flash памяти

Flash память микроконтроллеров ATmega603/103 организован из 256/512 страниц по 256 байт каждая. При программировании Flash памяти данные программы фиксируются в буфере страницы, что позволяет одновременно программировать целую страницу. Ниже приведена процедура, описывающая программирование всего объема Flash памяти.

A: Загрузка команды программирования Flash памяти (Program Flash)

1. Установить биты XA1 и XA0 в состояние 1 и 0, соответственно. Эта установка разрешает загрузку команды.
2. Установить бит BS1 в состояние 0.
3. Установить биты PB(7 - 0) в состояние 0001 0000. Это команда программирования Flash памяти.
4. Подать на вывод XTAL1 положительный импульс. Этим импульсом загружается команда.

B: Загрузка младшего байта адреса

1. Установить биты XA1 и XA0 в состояние 0 и 0. Эта установка разрешает загрузку адреса
2. Установить бит BS1 в состояние 0.
3. Установить в битах PB(7 - 0) младший байт адреса (\$00 - \$FF).
4. Подать на вывод XTAL1 положительный импульс. Этим импульсом загружается младший байт адреса.

C: Загрузка байта данных

1. Установить биты XA1 и XA0 в состояние 0 и 1, соответственно. Эта установка разрешает загрузку данных.
2. Установить в битах PB(7 - 0) младший байт данных (\$00 - \$FF).
3. Подать на вывод XTAL1 положительный импульс. Этим импульсом загружается байт данных.

D: Фиксация младшего байта данных

1. Установить бит BS1 в состояние 0. Эта установка выбирает младший байт данных.
2. Подать на вывод PAGED положительный импульс. Этим импульсом загружается байт данных.

E: Загрузка байта данных

1. Установить биты XA1 и XA0 в состояние 0 и 1, соответственно. Эта установка разрешает загрузку данных.
2. Установить в битах PB(7 - 0) старший байт данных (\$00 - \$FF).
3. Подать на вывод XTAL1 положительный импульс. Этим импульсом загружается байт данных.

F: Фиксация старшего байта данных

1. Установить бит BS1 в состояние 1. Эта установка выбирает старший байт данных.
2. Подать на вывод PAGED положительный импульс. Этим импульсом загружается байт данных.

G: Пункты с B по F повторяются 128 раз до полного заполнения буфера страницы.H: Загрузка старшего байта адреса

1. Установить биты XA1 и XA0 в состояние 0 и 0, соответственно. Эта установка разрешает загрузку адреса.
2. Установить бит BS1 в состояние 1. Эта установка выбирает старший байт адреса.
3. Установить в битах PB(7 - 0) старший байт адреса (для Atmega603: \$7F, для Atmega103: \$FF).
4. Подать на вывод XTAL1 положительный импульс. Этим импульсом загружается старший байт адреса.

I: Программирование страницы

1. Подать на вывод WR отрицательный импульс. Этим импульсом запускается программирование страницы данных. Вывод переходит RDY/BSY на низкий уровень.
2. Ожидать перехода вывода RDY/BSY на высокий уровень.

J: Завершение программирования страницы

1. Установить биты XA1 и XA0 в состояние 1 и 0. Эта установка разрешает загрузку команды.
2. Установить биты PB(7 - 0) в состояние 0000 0000. Это команда «Нет операции».
3. Подать на вывод XTAL1 положительный импульс. Этим импульсом загружается команда и сбрасываются все внутренние сигналы записи.

K: Пункты с A по J повторяются 256/512 раз, или до тех пор, пока все данные не будут запрограммированы.

Программирование EEPROM

Алгоритм программирования EEPROM памяти данных приведен ниже (подробности загрузки команд, адресов и данных см. в разделе Программирование Flash памяти).

1. Загрузить команду 0001 0001.
2. Загрузить младший адрес EEPROM (\$00 - \$FF).
3. Загрузить старший адрес EEPROM (для ATmega603: \$07, для ATmega103: \$0F).
4. Загрузить младший байт данных EEPROM (\$00 - \$FF).
5. Подать на вывод \overline{WR} отрицательный импульс и ожидать перехода вывода RDY/ \overline{BSY} на высокий уровень.

Команда загружается только перед программированием первого байта.

Чтение Flash памяти

Алгоритм чтения Flash памяти данных приведен ниже (подробности загрузки команд, адресов и данных см. в разделе Программирование Flash памяти).

1. Загрузить команду 0000 0010.
2. Загрузить младший адрес (\$00 - \$FF).
3. Загрузить старший адрес (для ATmega603: \$7F, для ATmega103: \$FF).
4. Установить \overline{OE} в состояние 0 и BS1 в состояние 0. По выводам PB(7 - 0) можно считать младший байт данных.
5. Установить BS в состояние 1. По выводам PB(7 - 0) можно считать старший байт данных.
6. Установить \overline{OE} в состояние 1.

Команда загружается только перед чтением первого байта.

Чтение памяти EEPROM

Алгоритм чтения EEPROM памяти данных приведен ниже (подробности загрузки команд, адресов и данных см. в разделе Программирование Flash памяти).

1. Загрузить команду 0000 0011.
2. Загрузить младший адрес EEPROM (\$00 - \$FF).
3. Загрузить старший адрес EEPROM (для ATmega603: \$07, для ATmega103: \$0F).
4. Установить \overline{OE} в состояние 0 и BS1 в состояние 0. По выводам PB(7 - 0) можно считать байт данных EEPROM.
5. Установить \overline{OE} в состояние 1.

Команда загружается только перед чтением первого байта.

Программирование битов-предохранителей

Алгоритм программирования битов-предохранителей приведен ниже (подробности загрузки команд, адресов и данных см. в разделе Программирование Flash памяти).

1. Загрузить команду 0100 0000.
2. Загрузить данные
Бит 5 = 0 программирует SPIEN бит-предохранитель. Бит 5 = 1 стирает SPIEN бит-предохранитель.
Бит 3 = 0 программирует EESAVE бит-предохранитель. Бит 3 = 1 стирает EESAVE бит-предохранитель.
Бит 2 = всегда в состоянии 1.
Бит 1 = 0 программирует SUT1 бит-предохранитель. Бит 1 = 1 стирает SUT1 бит-

предохранитель.

Бит 0 = 0 программирует SUTO бит-предохранитель. Бит 0 = 1 стирает SUTO бит-предохранитель.

3. Подать на вывод \overline{WR} отрицательный импульс и ожидать перехода вывода RDY/ \overline{BSY} на высокий уровень

Программирование битов блокирования

Алгоритм программирования битов блокирования приведен ниже (подробности загрузки команд, адресов и данных см. в разделе Программирование Flash памяти.

1. Загрузить команду 0010 0000.

2. Загрузить данные.

Бит 2 = 0 программирует бит блокирования 2

Бит 1 = 0 программирует бит блокирования 1

3. Подать на вывод \overline{WR} отрицательный импульс и ожидать перехода вывода RDY/ \overline{BSY} на высокий уровень

Биты блокирования могут быть очищены только при выполнении операции очистки кристалла

Чтение битов-предохранителей и битов блокирования

Алгоритм чтения битов-предохранителей и битов блокирования приведен ниже (подробности загрузки команд, адресов и данных см. в разделе Программирование Flash памяти.

1. Загрузить команду 0000 0100.

2. Установить \overline{OE} в состояние 0 и BS1 в состояние 0 или 1. По выводам PB(7 - 0) можно считать состояния битов-предохранителей и битов блокирования.

При BS1 = 0

Бит 5: бит-предохранитель SPIEN (0 - запрограммирован, 1 - очищен)

Бит 3: бит-предохранитель SPIEN (0 - запрограммирован, 1 - очищен)

Бит 1: бит-предохранитель SUT1 (0 - запрограммирован, 1 - очищен)

Бит 0: бит-предохранитель SUTO (0 - запрограммирован, 1 - очищен).

При BS1 = 0

Бит 2: бит блокирования LB1 (0 - запрограммирован, 1 - очищен)

Бит 1: бит блокирования LB0 (0 - запрограммирован, 1 - очищен)

3. Установить \overline{OE} в состояние 1.

Чтение байтов сигнатуры

Алгоритм чтения байтов сигнатуры приведен ниже (подробности загрузки команд, адресов и данных см. в разделе Программирование Flash памяти.

1. Загрузить команду 0000 1000.

2. Загрузить байт адреса (\$00 - \$02).

3. Установить \overline{OE} в состояние 0 и BS1 в состояние 0. По выводам PB(7 - 0) можно считать байт сигнатуры.

4. Установить \overline{OE} в состояние 1.

Команда загружается только перед чтением первого байта.

Программирование Flash и EEPROM памяти по последовательному каналу (Serial Downloading)

И Flash память программ и EEPROM память данных могут быть запрограммированы по последовательной шине SPI. При этом вывод $\overline{\text{RESET}}$ должен быть подключен к уровню GND. Последовательный интерфейс организуется посредством выводов SCK, RXD/PDI (вход) и TXD/PDO (выход). После установки на выводе $\overline{\text{RESET}}$ низкого уровня, прежде чем начать операции программирования/стирания, необходимо выполнить команду «Разрешить программирование». При программировании EEPROM в процессе самотактируемой операции программирования выполняется автоматическое стирание (только в режиме последовательного программирования) и, следовательно, на требуется выполнения команды очистки кристалла. Операция очистки кристалла устанавливает содержимое всех ячеек памяти и в памяти программ и в EEPROM памяти в состояние \$FF.

Матрицы памяти программ и EEPROM памяти расположены в отдельных адресных пространствах:

ATmega603: с \$0000 по \$7FFF для памяти программ и с \$0000 по \$07FF для EEPROM памяти.

ATmega103: с \$0000 по \$FFFF для памяти программ и с \$0000 по \$0FFF для EEPROM памяти.

Внешний системный тактовый сигнал подается на вывод XTAL1. Кварцевый кристалл подсоединяется между выводами XTAL1 и XTAL2. Минимальная длительность периодов высокого и низкого уровня на входе последовательного тактового сигнала (SCK) должна быть:

Низкий уровень - не менее 2 тактовых циклов XTAL1.

Высокий уровень - не менее 2 тактовых циклов XTAL1.

Алгоритм программирования по последовательному каналу

Для программирования и проверки микроконтроллера ATmega103/L в последовательном режиме рекомендуется следующий алгоритм (формат 4-байтовых команд см. в таблице 39):

1. Выполнить последовательность включения питания:

Подать на выводы VCC и GND напряжение питания, причем выводы $\overline{\text{RESET}}$ и SCK во время подачи питания должны быть на уровне 0. Если к выводам XTAL1 и XTAL2 не подсоединен кварцевый кристалл, то тактовый сигнал подается на вывод XTAL1. В ряде систем программирующий не может гарантировать нахождение вывода SCK на низком уровне в течение всего периода подачи питания. В этом случае на вывод $\overline{\text{RESET}}$ должен быть подан положительный импульс длительностью не менее двух циклов XTAL1 после того как на SCK будет установлен уровень 0.

2. Сделать выдержку не менее 20 мс и разрешить программирование в последовательном режиме посылкой на вывод PE0 (PDI/RXD) последовательной команды «Разрешить программирование».

3. При выдаче третьего байта команды «Разрешить программирование», значение, установленное как вторым байтом (\$53), будет возвращено в течение передачи третьего байта. Если значение \$53 не будет возвращено необходимо подать на вывод SCK положительный импульс и вновь послать команду «Разрешить программирование». Если возврат значения \$53 не будет получен в течение 32 попыток, то связь устанавливалась с неработающим устройством.

4. Если была выполнена очистка кристалла (выполняется если необходимо

очистить Flash память) делается выдержка 10 мс, подается положительный импульс на вывод RESET и начинается выполнение алгоритма с пункта 2.

5. Flash память программируется сразу по одной странице. Страница загружается по одному байту подачей 7 младших битов адреса и байта данных вместе с командой «Загрузить страницу памяти программ». Страница памяти программ сохраняется загрузкой команды «Записать страницу памяти программ» вместе с 9 старшими битами адреса.

6. Матрица EEPROM программируется по одному байту загрузкой адреса и данных вместе с соответствующей командой «Записать». Соответствующие ячейки памяти EEPROM, перед записью новых данных, автоматически очищаются. Следующий байт может быть записан через 4 мс (при напряжении питания 2,7 В) или через 1 мс (при напряжении питания 5,0 В).

7. Любой байт в области памяти может быть проверен командой «Читать», которая возвращает содержимое по заданному адресу через последовательный выход вывода PE1(PDO/TXD).

8. По полном завершении сеанса программирования, для перехода к нормальной работе, вывод RESET должен быть установлен на высокий уровень.

9. Выполнить последовательность отключения питания (при необходимости). Установить низкий уровень на выводе XTAL1 (если не использовался кварцевый кристалл). Установить высокий уровень на выводе RESET. Отключить напряжение питания VCC.

Таблица 38. Система команд программирования в последовательном режиме

Команда	Формат команды				Описание
	Байт 1	Байт 2	Байт 3	Байт 4	
Разрешить программирование (Programming Enable)	1010 1100	0101 0011	xxx xxx	xxx xxx	Разрешает последовательное программирование после установки RESET на низкий уровень
Очистить кристалл (Chip Erase)	1010 1100	100x xxxx	xxx xxx	xxx xxx	Очищает Flash и EEPROM память
Читать память программ (Read Program Memory)	0010 H000	aaaa aaaa	bbbb bbbb	oooo oooo	Читает H (1 или 0) и данные о памяти программ по слову адреса a:b
Загрузить страницу памяти программ (Load Program Memory Page)	0100 H000	xxx xxx	xbbb bbbb	iiii iii	Записывает H (1 или 0) и данные i в память программ по слову адреса b
Записать страницу памяти программ (Write Program Memory Page)	0100 1100	aaaa aaaa	bxxx xxx	xxx xxx	Записывает страницу памяти программ по адресу a:b
Читать EEPROM память (Read EEPROM Memory)	1010 0000	xxx aaaa	bbbb bbbb	oooo oooo	Читает данные o из EEPROM памяти по адресу a:b
Записать в EEPROM память (Write EEPROM Memory)	1100 0000	xxx aaaa	bbbb bbbb	iiii iii	Записывает данные i в EEPROM память по адресу a:b

КТЦ-МК

Читать биты блокировки (Read Lock Bits)	0101 1000	xxxx xxxx	xxxx xxxx	xxxx x 21x	Читает биты блокировки. 0 - запрограммирован, 1 - незапрограммирован
Записать биты блокировки (Write Lock Bits)	1010 1100	111x x 21x	xxxx xxxx	xxxx xxxx	Записывает биты блокировки. При программировании биты 1,2 = 0
Читать биты-предохранители (Read Fuse Bits)	0101 0000	xxxx xxxx	xxxx xxxx	xx 5x 61 43	Читает биты-предохранители. 0 - запрограммирован, 1 - незапрограммирован
Записать биты-предохранители (Write Fuse Bits)	1010 1100	101x 61 43	xxxx xxxx	xxxx xxxx	Записывает биты-предохранители. Устанавливает биты 6, 4,3=0 для программирования битов, 1 для очистки битов
Читать байт сигнатуры (Read Signature Byte)	0011 0000	xxxx xxxx	xxxx xxbb	oooo oooo	Читает байт сигнатуры о по адресу b

Примечания:

- a = старшие биты адреса
- b = младшие биты адреса
- H = 0 - младший байт, 1 - старший байт
- o = выход данных
- i = вход данных
- x = состояние значения не имеет
- 1 = бит блокировки 1
- 2 = бит блокировки 2
- 3 = бит-предохранитель SUT0
- 4 = бит-предохранитель SUT1
- 5 = бит-предохранитель SPIEN
- 6 = бит-предохранитель EESAVE

При последовательной записи в микроконтроллеры ATmega603/103 данные микроконтроллером ATmega 103 выбираются по нарастающему фронту на SCK.

При последовательном чтении данных из микроконтроллеров ATmega603/103 данные тактируются по падающему фронту на SCK. Объяснение представлено на Рис. 67.

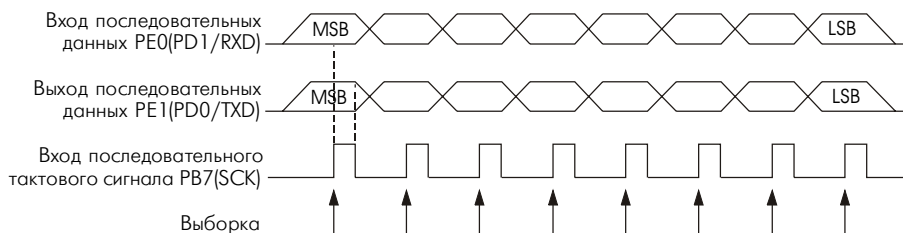


Рис. 67. Временные диаграммы режима последовательного программирования.

Предельно допустимые режимы эксплуатации

Диапазон рабочих температур	от -40°C до +105°C
Диапазон температур хранения	от -65°C до +150°C
Напряжение на любом выводе относительно земли	от -1,0 В до +7,0 В
Максимальное рабочее напряжение	6,6 В
Потребление по постоянному току	300 мА

Примечание: Воздействия за пределами указанных в таблице режимов могут привести к выходу прибора из строя. Указанные режимы являются стрессовыми и постоянная работа приборов в этих и других режимах за пределами, указанными в спецификациях, недопустима. Длительное воздействие предельных режимов может привести к снижению надежности приборов.

Характеристики по постоянному току

TA = от -40°C до 85°C, VCC = от 2,7 В до 6,0 В (если не оговорено другое)

Обознач.	Параметр	Условия	Мин	Тип	Макс	Ед. измерен.
V _{IL}	Входное напряжение низкого уровня		-0,5		0,3 V _{CC}	В
V _{IL1}	Входное напряжение низкого уровня	XTAL	-0,5		0,2 V _{CC}	В
V _{IH}	Входное напряжение высокого уровня	Исключая (XTAL, RESET)	0,6 V _{CC}		V _{CC} +0,5	В
V _{IH1}	Входное напряжение высокого уровня	XTAL	0,8 V _{CC}		V _{CC} +0,5	В
V _{IH2}	Входное напряжение высокого уровня	RESET	V _{CC}		V _{CC} +0,5	В
V _{OL}	Выходное напряжение низкого уровня ⁽¹⁾ Порты А, В, С, D	I _{OL} = 20 мА, V _{CC} = 5В I _{OL} = 10 мА, V _{CC} = 3В			0,6 0,5	В
V _{OH}	Выходн. напряжение высокого уровня Порты А, В, С, D	I _{OH} = -3 мА, V _{CC} = 5В I _{OH} = -1,5 мА, V _{CC} = 3В	4,2 2,3			В
I _{IL}	Входной ток утечки вывода I/O	V _{CC} = 6В, низкий уровень	-8,0		8,0	мкА
I _{IH}	Входной ток утечки вывода I/O	V _{CC} = 6 В, высокий уровень	-8,0		8,0	мкА
RRST	Нагрузочный резистор сброса		100		500	кОм
R _{I/O}	Нагрузочный резистор I/O		35		120	кОм
I _{CC}	Потребляемый ток в режимах:	Активный, 4 МГц, V _{CC} =3 В(2)			3,0	мА
		Idle, 4 МГц, V _{CC} =3 В		1,0	1,2	мА
		Power Down, 4 МГц, V _{CC} =3 В, WDT разрешен		8,5	15	мкА
		Power Down, 4 МГц, V _{CC} =3 В, WDT запрещен		< 1	2,0	мкА
V _{ACIO}	Напряжение смещения входа аналогового компаратора	V _{CC} = 5 В			40	мВ
I _{ACLK}	Утечка по входу аналогового компаратора	V _{CC} = 5 В, V _{IN} = V _{CC} / 2	-50		50	нА
t _{ACPD}	Задержка аналогового компаратора	V _{CC} = 2,7 В, V _{CC} = 4,0 В		750 500		нс

КТЦ-МК

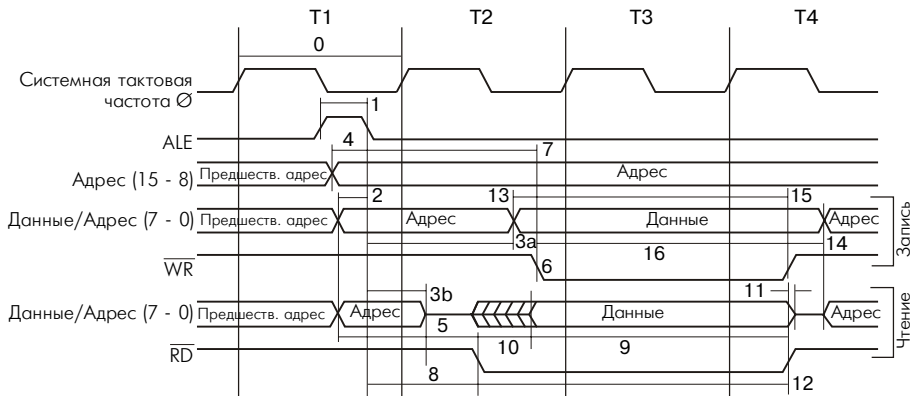
Примечания: 1. В установившемся режиме (не в переходном) значения IOL должны внешними средствами ограничиваться на уровне:
 Максимальный IOL на каждом выводе порта - 10 мА
 Максимальный IOL по всем выводам выхода - 300 мА
 Порт А - 26 мА
 Порты А, В, D - 15 мА
 В случае превышения тестовых величин IOL величина VOL может превысить соответствующие значения.
 Не гарантируется вытекающий ток вывода больше указанного.
 2. При тактовой частоте XTAL = 4 МГц тактовая частота шины тоже 4 МГц.
 3. Минимальное значение VCC в режиме Power Down составляет 2 В

Таблица 39. Характеристики внешней памяти данных, напряжение питания от 4,0 до 6,0 В, без состояния ожидания

Обозн.	Параметр	Генератор 8 МГц		Настраиваемый генератор		Ед. измер.
		Мин	Макс	Мин	Макс	
0	$1/t_{\text{CLCL}}$	Частота генератора		0,0	8,0	МГц
1	t_{LHLL}	32,5		$0,5t_{\text{CLCL}} - 30,0$		нс
2	t_{AVLL}	22,5		$0,5t_{\text{CLCL}} - 40,0$		нс
3a	$t_{\text{LLAX_ST}}$	67,5		$0,5t_{\text{CLCL}} - 5,0$		нс
3b	$t_{\text{LLAX_LD}}$	15,0		15,0		нс
4	t_{AVLCL}	22,5		$0,5t_{\text{CLCL}} - 40,0$		нс
5	t_{AVRL}	95,0		$1,0t_{\text{CLCL}} - 30,0$		нс
6	t_{AVWL}	157,5		$1,5t_{\text{CLCL}} - 30,0$		нс
7	t_{LLWL}	105,0	145	$1,0t_{\text{CLCL}} - 20,0$	$1,0t_{\text{CLCL}} + 20,0$	нс
8	t_{LLRL}	42,5	82,5	$0,5t_{\text{CLCL}} - 20,0$	$0,5t_{\text{CLCL}} + 20,0$	нс
9	t_{DVRLH}	60,0		60,0		нс
10	t_{RLDV}		70,0		$1,0t_{\text{CLCL}} - 55,0$	нс
11	t_{RHDX}	0,0		0,0		нс
12	t_{RLRH}	105,0		$1,0t_{\text{CLCL}} - 20,0$		нс
13	t_{DVWL}	27,5		$0,5t_{\text{CLCL}} - 35,0$		нс
14	t_{WHDX}	0,0		0,0		нс
15	t_{DVWH}	95,0		$1,0t_{\text{CLCL}} - 30,0$		нс
16	t_{WLWH}	42,5		$0,5t_{\text{CLCL}} - 20,0$		нс

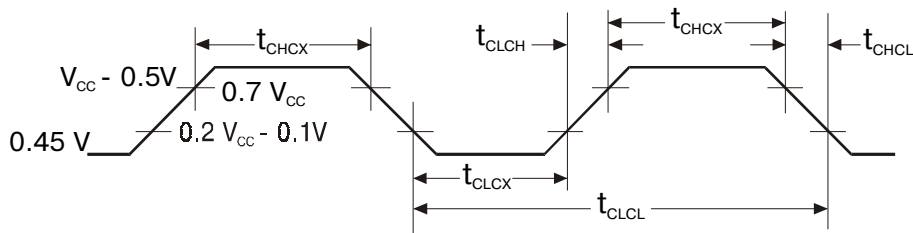
Таблица 40. Характеристики внешней памяти данных, напряжение питания от 4,0 до 6,0 В, состояния ожидания 1 цикл

Обозн.	Параметр	Генератор 8 МГц		Настраиваемый генератор		Ед. измер.
		Мин	Макс	Мин	Макс	
0	$1/t_{\text{CCL}}$ Частота генератора			0,0	8,0	МГц
10	t_{RLDV} Низкое чтение до действительных данных		195,0		$2,0t_{\text{CCL}} - 55,0$	нс
12	t_{RLRH} Ширина импульса RD	230,0		$2,0t_{\text{CCL}} - 20,0$		нс
15	t_{DWH} Действительность данных до высокого WR	220,0		$2,0t_{\text{CCL}} - 30,0$		нс
16	t_{VLWH} Ширина импульса WR	167,5		$1,5t_{\text{CCL}} - 20,0$		нс



Примечание: Тактовый цикл T3 присутствует только в том случае, когда разрешено состояние ожидания внешней SRAM.

Рис. 68. Временные диаграммы тактирования внешней памяти



Временная диаграмма внешнего тактового сигнала

Таблица 41. Характеристики внешней памяти данных, напряжение питания от 2,7 до 6,0 В, без состояния ожидания

	Обозн.	Параметр	Генератор 8 МГц		Настраиваемый генератор		Ед. измер.		
			Мин	Макс	Мин	Макс			
0	$1/t_{\text{CLCL}}$	Частота генератора			0,0	4,0	МГц		
1	t_{LHLL}	Ширина импульса ALE	70,0		$0,5t_{\text{CLCL}}$	-55,0	нс		
2	t_{AVLL}	Действительность адреса А до низкого ALE	60,0		$0,5t_{\text{CLCL}}$	-65,0	нс		
3a	$t_{\text{LLAX_ST}}$	Удержание адреса после низкого ALE, команды ST/STD/STS	130,0		$0,5t_{\text{CLCL}}$	-65,0	нс		
3b	$t_{\text{LLAX_LD}}$	Удержание адреса после низкого ALE, команды LD/LDD/LDS	15,0		15,0		нс		
4	t_{AVLLC}	Действительность адреса С до низкого ALE	60,0		$0,5t_{\text{CLCL}}$	-65,0	нс		
5	t_{AVRL}	Действительность адреса до низкого RD	200,0		$1,0t_{\text{CLCL}}$	-50,0	нс		
6	t_{AVWL}	Действительность адреса до низкого WR	325,0		$1,5t_{\text{CLCL}}$	-50,0	нс		
7	t_{LLWL}	Низкий ALE до низкого WR	230,0	270,0	$1,0t_{\text{CLCL}}$	-20,0	$1,0t_{\text{CLCL}}$	+20,0	нс
8	t_{LLRL}	Низкий ALE до низкого RD	105,0	145,0	$0,5t_{\text{CLCL}}$	-20,0	$0,5t_{\text{CLCL}}$	+20,0	нс
9	t_{DVRLH}	Установка данных до высокого RD	95,0		95,0			нс	
10	t_{RLDV}	Низкое чтение до действительных данных		170,0			$1,0t_{\text{CLCL}}$	-80,0	нс
11	t_{RHDX}	Удержание данных после высокого RD	0,0		0,0			нс	
12	t_{RLRH}	Ширина импульса RD	230,0		$1,0t_{\text{CLCL}}$	-20,0		нс	
13	t_{DVWL}	Установка данных до низкого WR	70,0		$0,5t_{\text{CLCL}}$	-55,0		нс	
14	t_{VHDX}	Удержание данных после высокого WR	0,0		0,0			нс	
15	t_{DVWH}	Действительность данных до высокого WR	210,0		$1,0t_{\text{CLCL}}$	-40,0		нс	
16	t_{VLWH}	Ширина импульса WR	105,0		$0,5t_{\text{CLCL}}$	-20,0		нс	

Таблица 42. Характеристики внешней памяти данных, напряжение питания от 2,7 до 6,0 В, состояния ожидания 1 цикл

	Обозн.	Параметр	Генератор 8 МГц		Настраиваемый генератор		Ед. измер.		
			Мин	Макс	Мин	Макс			
0	$1/t_{\text{CLCL}}$	Частота генератора			0,0	8,0	МГц		
10	t_{RLDV}	Низкое чтение до действительных данных		420,0			$2,0t_{\text{CLCL}}$	-20,0	нс
12	t_{RLRH}	Ширина импульса RD	480,0		$2,0t_{\text{CLCL}}$	-20,0		нс	
15	t_{DVWH}	Действительность данных до высокого WR	460,0		$2,0t_{\text{CLCL}}$	-40,0		нс	
16	t_{VLWH}	Ширина импульса WR	355,5		$1,5t_{\text{CLCL}}$	-20,0		нс	

Временные характеристики внешнего тактового сигнала

Обозн.	Параметр	$V_{cc}=2,7$ до $6,0$ В		$V_{cc}=4,0$ до $6,0$ В		Ед. изм.
		0	4	0	6	
$1/t_{CLCL}$	Частота тактового генератора	0	4	0	6	МГц
t_{CLCL}	Период тактовой частоты	250		167		нс
t_{CHCX}	Длительность по высокому уровню	0		0		нс
t_{CLCX}	Длительность по низкому уровню	0		0		нс
t_{CLCH}	Длительность нарастающего фронта		1,6		0,5	мкс
t_{CHCL}	Длительность падающего фронта		1,6		0,5	мкс

Информация для заказа

Рабочая частота, МГц	Напряжение питания, В	Код для заказа	Тип корпуса	Диапазон рабочих температур
4	2,7 - 6,0	Atmega603L-4AC	64A	Коммерческий (от 0°C до 70°C)
		Atmega603L-4AI	64A	Промышленный (от -40°C до 85°C)
6	4,0 - 6,0	Atmega603L-6AC	64A	Коммерческий (от 0°C до 70°C)
		Atmega603L-6AI	64A	Промышленный (от -40°C до 85°C)
4	2,7 - 6,0	Atmega103L-4AC	64A	Коммерческий (от 0°C до 70°C)
		Atmega103L-4AI	64A	Промышленный (от -40°C до 85°C)
6	4,0 - 6,0	Atmega103L-6AC	64A	Коммерческий (от 0°C до 70°C)
		Atmega103L-6AI	64A	Промышленный (от -40°C до 85°C)

Тип корпуса

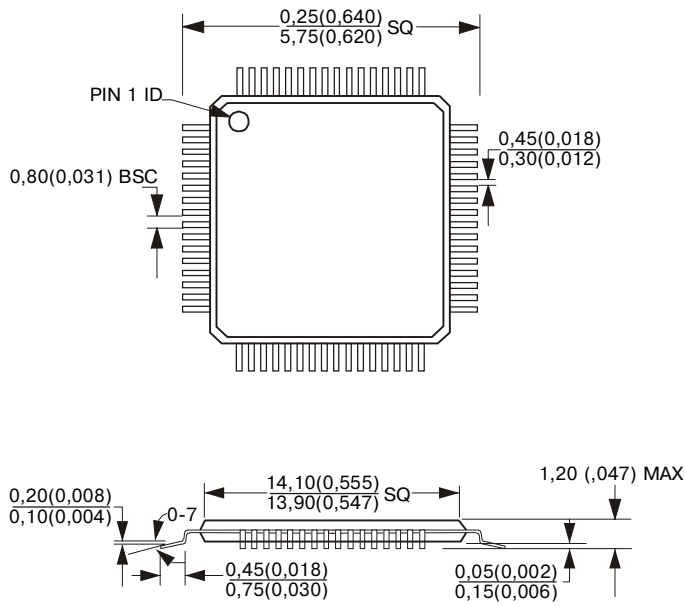
64A	64-выводной тонкий (1,0 мм) квадратный пластмассовый корпус TQFP (Plastic Gull Wing Quad Flat Package)
-----	--

Таблица регистров микроконтроллеров АТмега603/103

Адрес	Обознач.	Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	Стр.
\$3F (\$5F)	SREG	I	T	H	S	V	N	Z	C	
\$3E (\$5E)	SPH	SP15	SP14	SP13	SP12	SP11	SP10	SP9	SP8	
\$3D (\$5D)	SPL	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0	
\$3C (\$5C)	XDIV	XDIVEN	XDIV6	XDIV5	XDIV4	XDIV3	XDIV2	XDIV1	XDIV0	
\$3B (\$5B)	RAMPZ	-	-	-	-	-	-	-	RAMPX0	
\$3A (\$5A)	EICR	ISC71	ISC70	ISC61	ISC60	ISC51	ISC50	ISC41	ISC40	
\$39 (\$59)	EIMSK	INT7	INT6	INT5	INT4	INT3	INT2	INT1	INT0	
\$38 (\$58)	EIFR	INTF7	INTF6	INTF5	INTF4					
\$37 (\$57)	TIMSK	OCIE2	TOIE2	TICIE1	OCIE1A	OCIE1B	TOIE1	OCIE0	TOIE0	
\$36 (\$56)	TIFR	OCF2	TOV2	ICF1	OCF1A	OCF1B	TOV1	OCF0	TOV0	
\$35 (\$55)	MCUCR	SRE	SRW	SE	SM1	SM0	-	-	-	
\$34 (\$54)	MCUSR	-	-	-	-	-	-	EXTRF	PORF	
\$33 (\$53)	TCCR0	-	PWM0	COM01	COM00	CTC0	CS02	CS01	CS00	
\$32 (\$52)	TCNT0	Таймер/счетчик0 (8-разрядный)								
\$31 (\$51)	OCR0	Регистр сравнения выхода Таймера/счетчика0								
\$30 (\$50)	ASSR	-	-	-	-	AS0	TCN0UB	OCR0UB	TCR0UB	
\$2F (\$4F)	TCCR1A	COM1A1	COM1A0	COM1B1	COM1B0	-	-	PWM11	PWM10	
\$2E (\$4E)	TCCR1B	ICNC1	ICES1	-	-	CTC1	CS12	CS11	CS10	
\$2D (\$4D)	TCNT1H	Старший байт регистра счетчика Таймера/счетчика1								
\$2C (\$4C)	TCNT1L	Младший байт регистра счетчика Таймера/счетчика1								
\$2B (\$4B)	OCR1AH	Старший байт регистра сравнения А выхода Таймера/счетчика1								
\$2A (\$4A)	OCR1AL	Младший байт регистра сравнения А выхода Таймера/счетчика1								
\$29 (\$49)	OCR1BH	Старший байт регистра сравнения В выхода Таймера/счетчика1								
\$28 (\$48)	OCR1BL	Младший байт регистра сравнения В выхода Таймера/счетчика1								
\$27 (\$47)	ICR1H	Старший байт регистра захвата входа Таймера/счетчика1								
\$26 (\$46)	ICR1L	Младший байт регистра захвата входа Таймера/счетчика1								
\$25 (\$45)	TCCR2	-	PWM2	COM21	COM20	CTC2	CS22	CS21	CS20	
\$24 (\$44)	TCNT2	Таймер/счетчик2 (8-разрядный)								
\$23 (\$43)	OCR2	Регистр сравнения выхода Таймера/счетчика2								
\$21 (\$41)	WDTCSR	-	-	-	WDT0E	WDE	WDP2	WDP1	WDP0	
\$1F (\$3F)	EEARH	-	-	-	-	EEAR11	EEAR10	EEAR9	EEAR8	
\$1E (\$3E)	EEARL	Младший регистр адреса EEPROM								
\$1D (\$3D)	EEDR	Регистр данных EEPROM								
\$1C (\$3C)	EEDCR	-	-	-	-	EERIE	EEMWE	EEWE	EERE	
\$1B (\$3B)	PORTA	PORTA7	PORTA6	PORTA5	PORTA4	PORTA3	PORTA2	PORTA1	PORTA0	
\$1A (\$3A)	DDRA	DDA7	DDA6	DDA5	DDA4	DDA3	DDA2	DDA1	DDA0	
\$19 (\$39)	PINA	PINA7	PINA6	PINA5	PINA4	PINA3	PINA2	PINA1	PINA0	
\$18 (\$38)	PORTB	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	
\$17 (\$37)	DDRB	ddb7	ddb6	ddb5	ddb4	ddb3	ddb2	ddb1	ddb0	
\$16 (\$36)	PINB	PINB7	PINB6	PINB5	PINB4	PINB4	PINB2	PINB1	PINB0	
\$15 (\$35)	PORTC	PORTC7	PORTC6	PORTC5	PORTC4	PORTC3	PORTC2	PORTC1	PORTC0	
\$12 (\$32)	PORTD	PORTD7	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0	
\$11 (\$31)	DDRD	DDD7	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	
\$10 (\$30)	PIND	PIND7	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	
\$0F (\$2F)	SPDR	Регистр данных SPI								
\$0E (\$2E)	SPSR	SPIF	WCOL	-	-	-	-	-	-	
\$0D (\$2D)	SPCR	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0	
\$0C (\$2C)	UDR	Регистр данных UART I/O								
\$0B (\$2B)	USR	RXC	TXC	UDRE	FE	OR	-	-	-	
\$0A (\$2A)	UCR	RXCIE	TXCIE	UDRIE	RXEN	TXEN	CHR9	RXB8	TXB8	
\$09 (\$29)	UBRR	Регистр управления скоростью UART								
\$08 (\$28)	ACSR	ACD	-	ACO	ACI	ACE	ACIC	ACIS1	ACIS0	
\$07 (\$27)	ADMUX	-	-	-	-	-	MUX2	MUX1	MUX0	
\$06 (\$26)	ADCSR	ADEN	ADCS	ADRF	ADIF	ADIE	ADPS2	ADPS1	ADPS0	
\$05 (\$25)	ADCH	-	-	-	-	-	-	ADC9	ADC8	
\$04 (\$24)	ADCL	ADC7	ADC6	ADC5	ADC4	ADC3	ADC2	ADC1	ADC0	
\$03 (\$23)	PORTE	PORTE7	PORTE6	PORTE5	PORTE4	PORTE3	PORTE2	PORTE1	PORTE0	
\$02 (\$22)	DDRE	DDE7	DDE6	DDE5	DDE4	DDE3	DDE2	DDE1	DDE0	
\$01 (\$21)	PINE	PINE7	PINE6	PINE5	PINE4	PINE3	PINE2	PINE1	PINE0	
\$00 (\$20)	PINF	PINF7	PINF6	PINF5	PINF4	PINF3	PINF2	PINF1	PINF0	

Информация по корпусу

64A, 64-выводной очень тонкий (0,1 мм)
 пластмассовый квадратный корпус TQFP
 (Plastic Gull Wing Quad Flat Package)



Размеры в миллиметрах и (дюймах)