

**Микропроцессорные**  
**устройства и системы**  
**(МПУиС)**

Вводная лекция  
2017

К.т.н. Бородин К.  
В.

# Микропроцессорные устройства и системы (МПУиС)

- Архитектура микроконтроллеров
- Компиляторы
- Программаторы
- Программирование и отладка контроллеров
- Порты ввода-вывода
- Таймеры/счетчики и ШИМ сигнал
- АЦП преобразователи
- UART и RS-232/RS-485

# Микропроцессорные устройства и системы

## Архитектура микроконтроллера

- **CISC (Complex Instruction Set Computer)** - выполняют более 200 команд разной степени сложности, которые имеют размер от 1 до 15 байт и обеспечивают более 10 различных способов адресации.
  - нефиксированное значение длины команды и инструкций;
  - арифметические действия кодируются в одной команде;
  - небольшое число регистров, каждый из которых выполняет строго определённую функцию.
  - ориентация на процессор

- **RISC (Reduced Instruction Set Computer)** - использование ограниченного набора команд фиксированного формата.
  - реализуют не более 100 команд, имеющих фиксированный формат длиной 4 байта.
  - фиксированная длина инструкций
  - значительно сокращено число используемых способов адресации
  - быстродействие увеличивается за счёт упрощения инструкций, но глубокого использования компилятора

- **VLW (Very Large Instruction Word)** - архитектура с очень длинными командами (128 бит и более), отдельные поля которых содержат коды, обеспечивающие выполнение различных операций

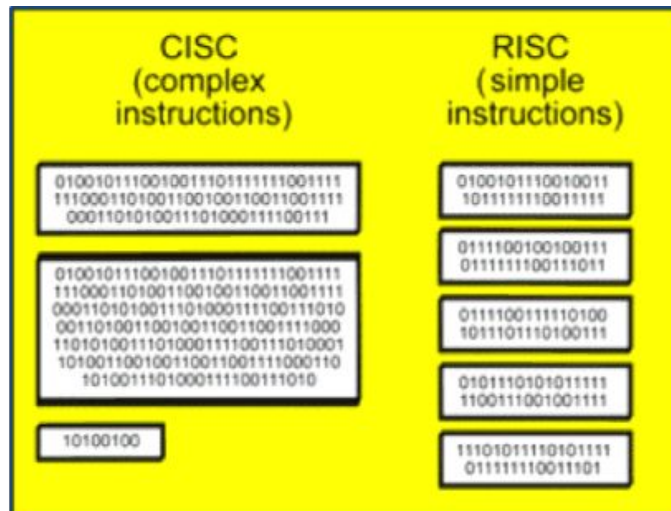
# Микропроцессорные устройства и системы

## Архитектура микроконтроллера

Одна большая или много маленьких команд выполнятся быстрее?



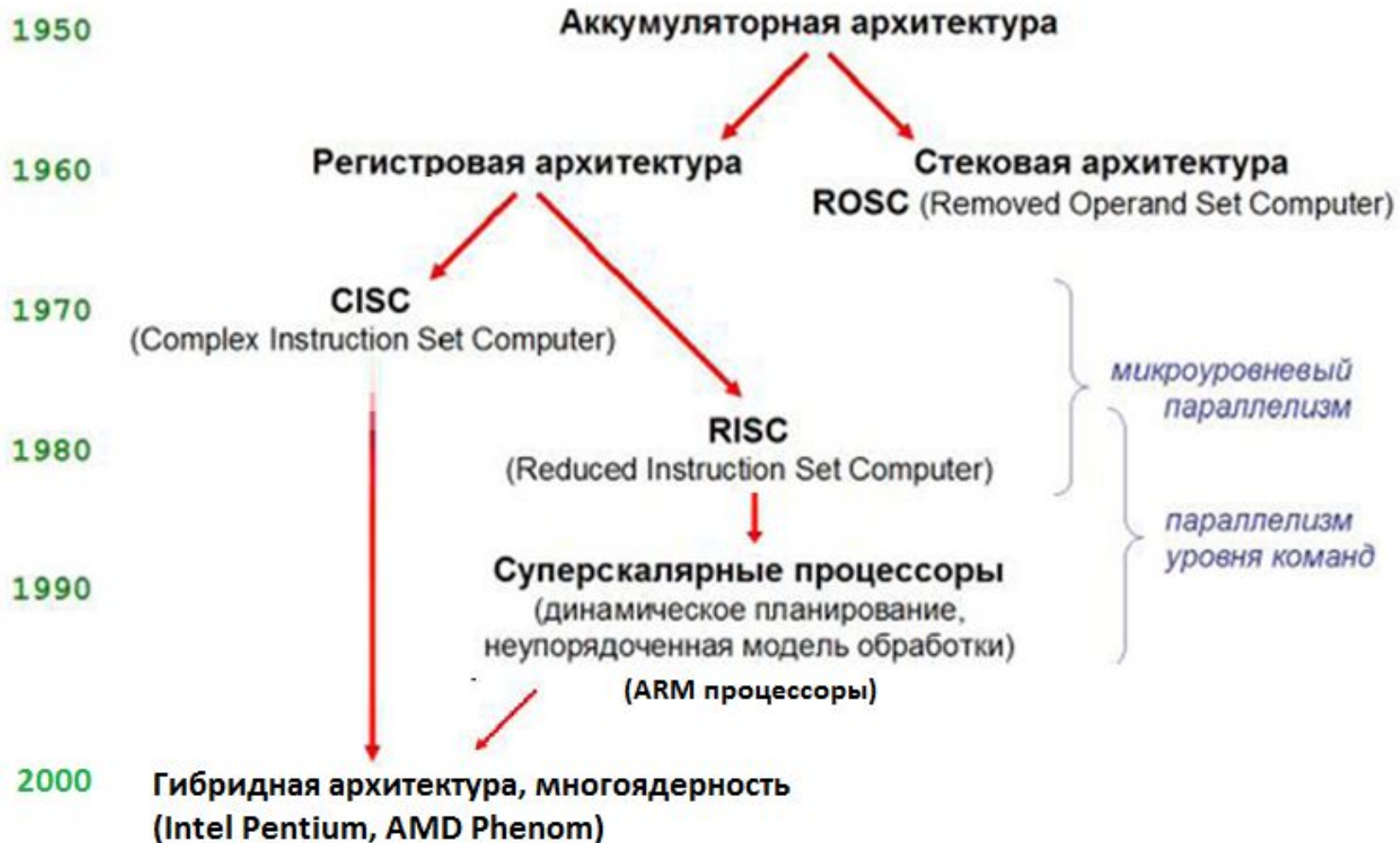
CISC – большие команды центральному процессору, которые разбиваются внутри на микроинструкции и в последствии выполняются разом



RISC – много отдельных коротких команд центральному процессору, которые сразу быстро выполняются

# Микропроцессорные устройства и системы

## Архитектура микроконтроллера



# Микропроцессорные устройства и системы

## Архитектура микроконтроллера

**Принстонская архитектура (архитектура Фон-Неймана)** - характеризуется использованием общей оперативной памяти для хранения команд (программ) и памяти данных, а также для организации стека. Для обращения к этой памяти используется общая системная шина, по которой в процессор поступают и команды, и данные

### **Достоинство:**

- ✓ наличие общей памяти
- ✓ общая шина данных для передачи команд и данных значительно упрощает отладку, тестирование и текущий контроль функционирования системы, повышает ее надежность

### **Недостаток:**

- ✓ необходимость последовательной выборки команд и обрабатываемых данных по общей системной шине, что ограничивает наращивание производительности цифровой системы



# Микропроцессорные устройства и системы

## Архитектура микроконтроллера

- **Гарвардская архитектура** - физическое разделение памяти команд (программ) и памяти данных.

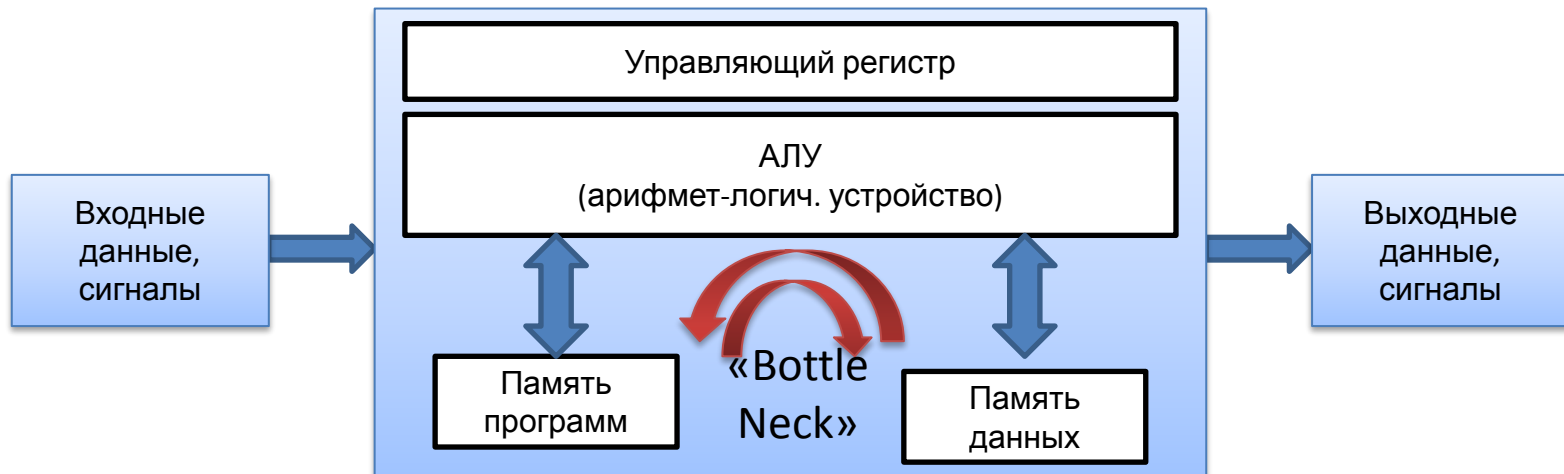
### Достоинство:

- ✓ Каждый внутренний блок памяти соединяется с процессорным ядром отдельной шиной, что позволяет одновременно с чтением-записью данных при выполнении текущей команды производить выборку и декодирование следующей команды
- ✓ более высокая производительность, чем при использовании Принстонской архитектуры

### Недостаток:

- ✓ сложность изготовления кристалла с большим количеством шин
- ✓ фиксированным объемом памяти, выделенной для команд и данных, значение которой не может оперативно перераспределяться в соответствии с требованиями решаемой задачи

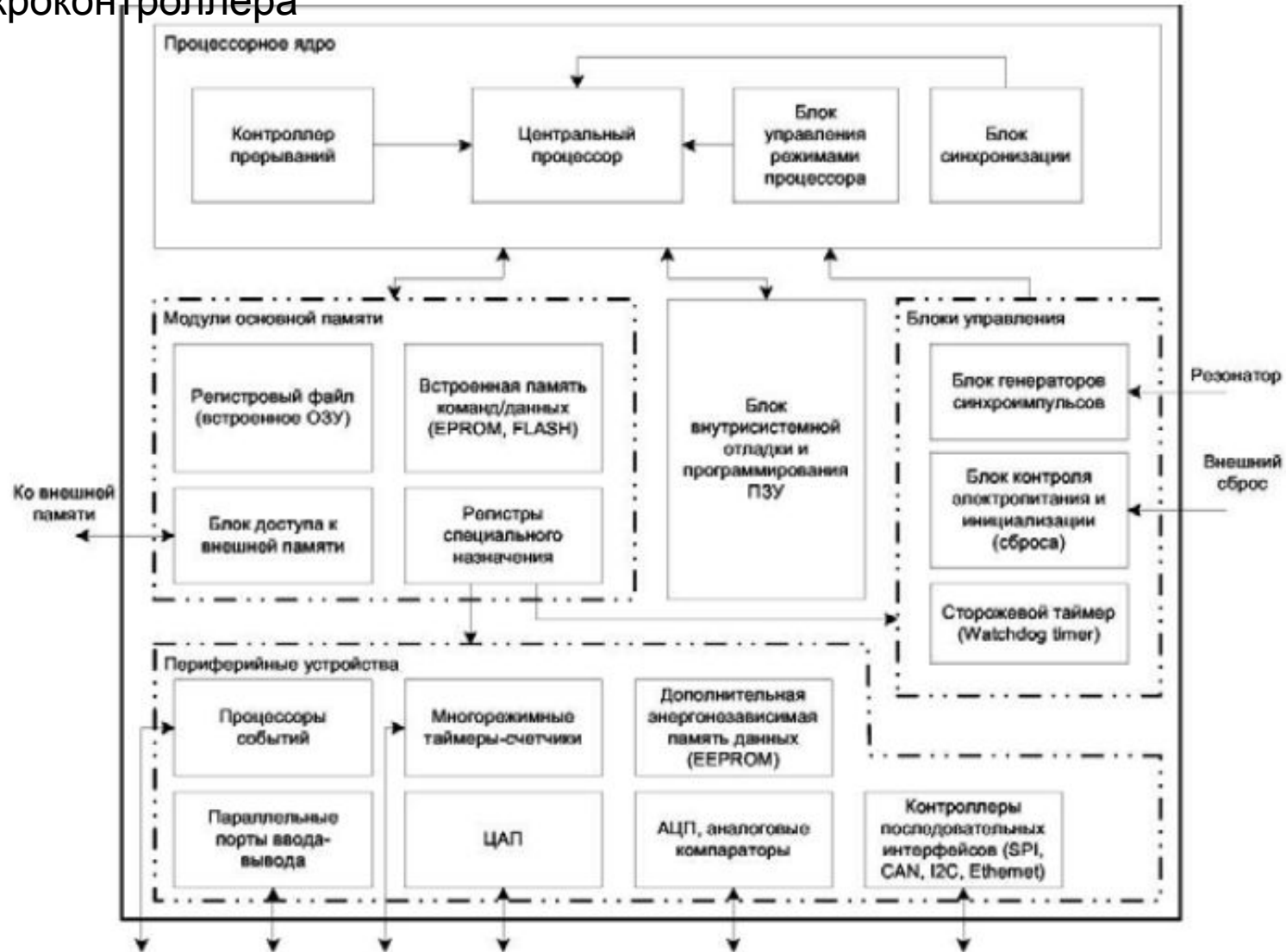
### Микроконтроллер



# Микропроцессорные устройства и системы

## Архитектура микроконтроллера

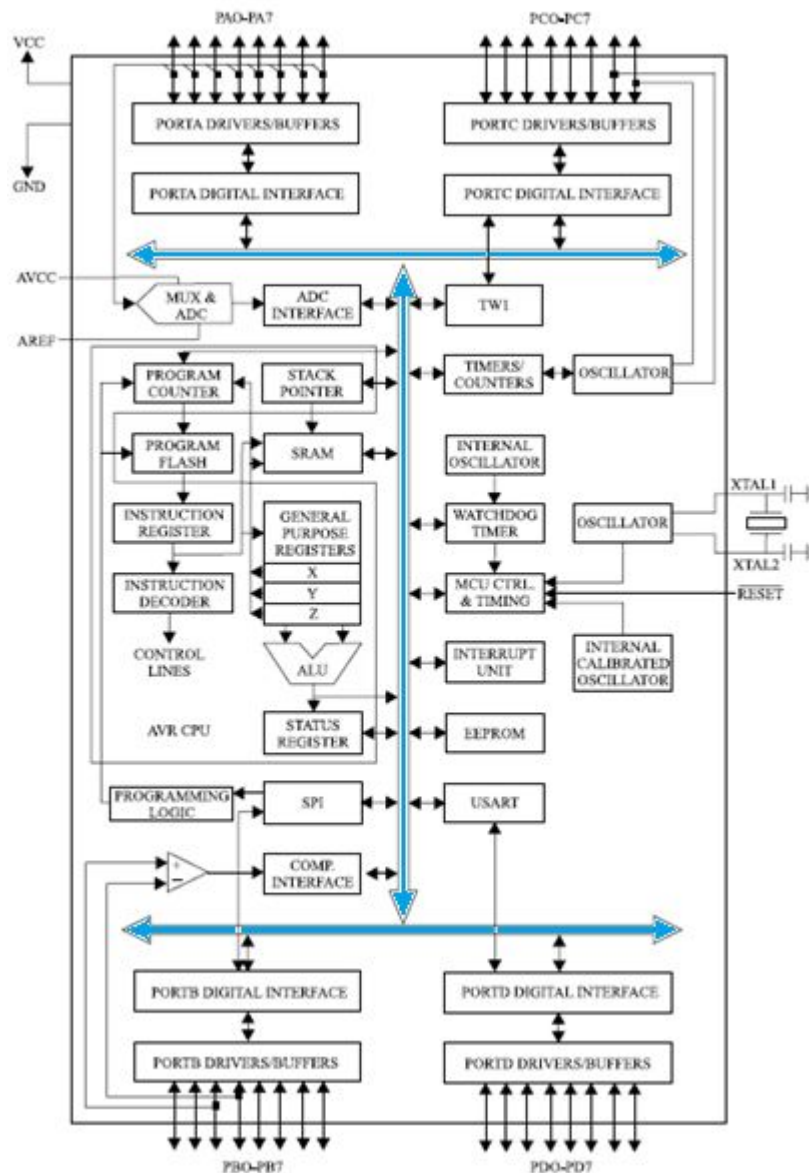
Типовая модульная внутренняя структура 8-разрядного микроконтроллера





# Микропроцессорные устройства и системы

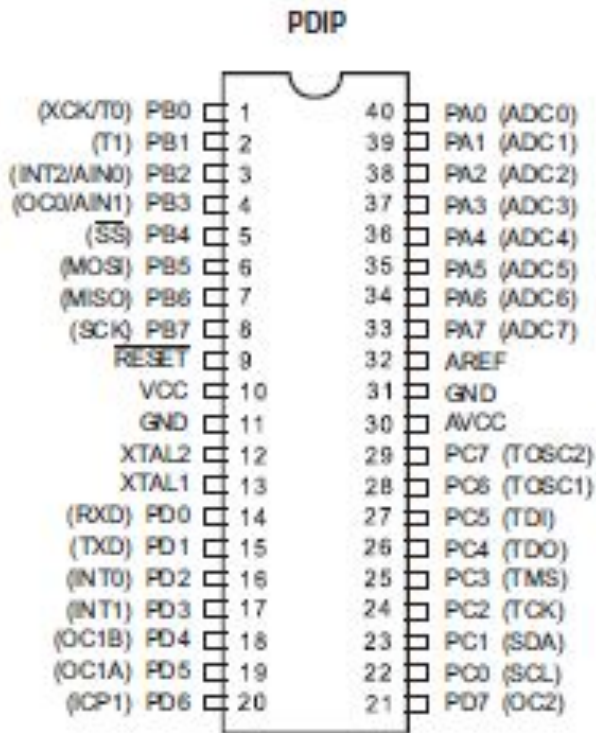
## Архитектура микроконтроллера AT Mega16



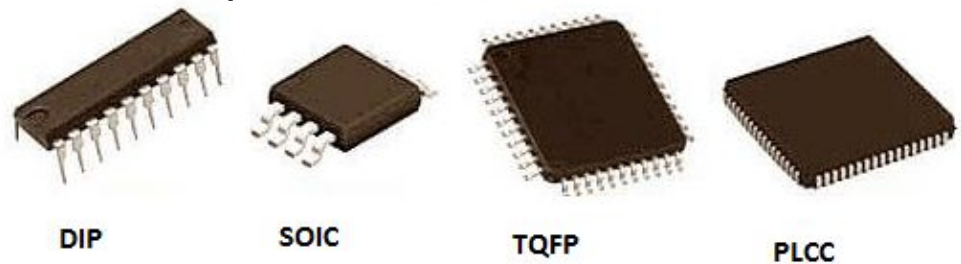
- На схеме показываются внутренние взаимосвязи модулей
- Общая шина данных объединяет все внутренние модули микроконтроллера
- На схеме явно обозначено 4 порта ввода-вывода (два сверху и два снизу), один АЦП, один USART и т.д.

# Микропроцессорные устройства и системы

## Архитектура микроконтроллера AT Mega16

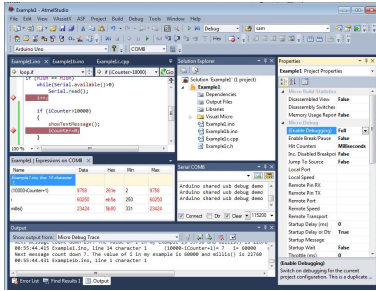


- Варианты корпусного исполнения:
- ✓ Прошивка контроллера обычно не зависит от корпуса – один и тот же код для разных вариантов расположения выводов.
  - ✓ Для изучения и отладки обычно выбирают корпус большего размера – удобно паять, подключать приборы, осциллографы..
  - ✓ Для финального варианта выбираю корпус меньшего габарита



Device	Flash ROM	EEPROM	RAM	Max I/O	F.max	Vcc	16-bit timer	8-bit timer	PWM	RTC	SPI	UART	TWI	AD	Int.	Ext Int.
ATmega16	16	0,5	1024	32	16	2,7-5,5	1	2	4	Yes	1	1	Yes	8	20	3

# Микропроцессорные устройства и системы. Программирование и отладка микроконтроллеров



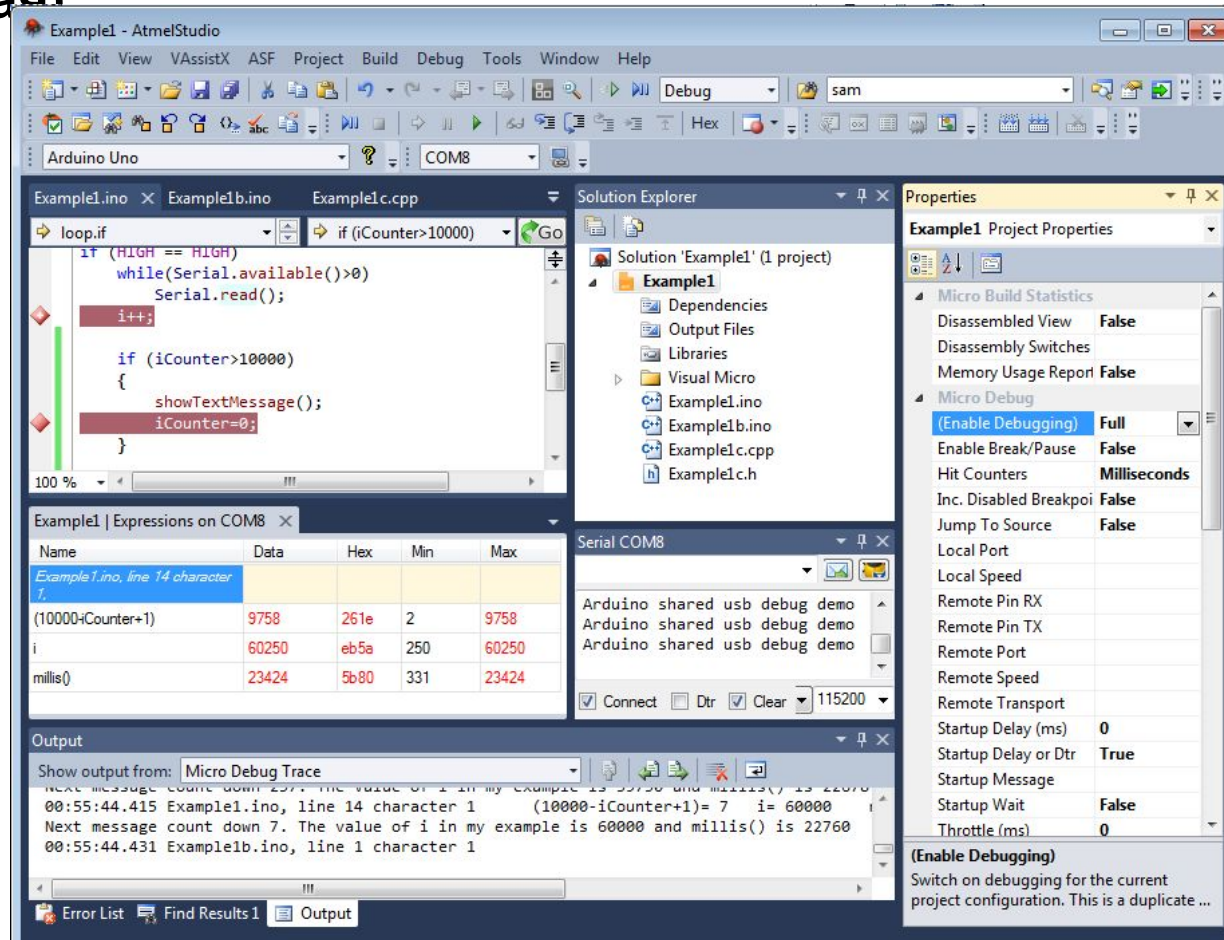
# Микропроцессорные устройства и системы.

## Компиляторы для микроконтроллеров - Atmel Studio7 (бесплатная)

Программный  
код

Значения  
переменных

сообщения



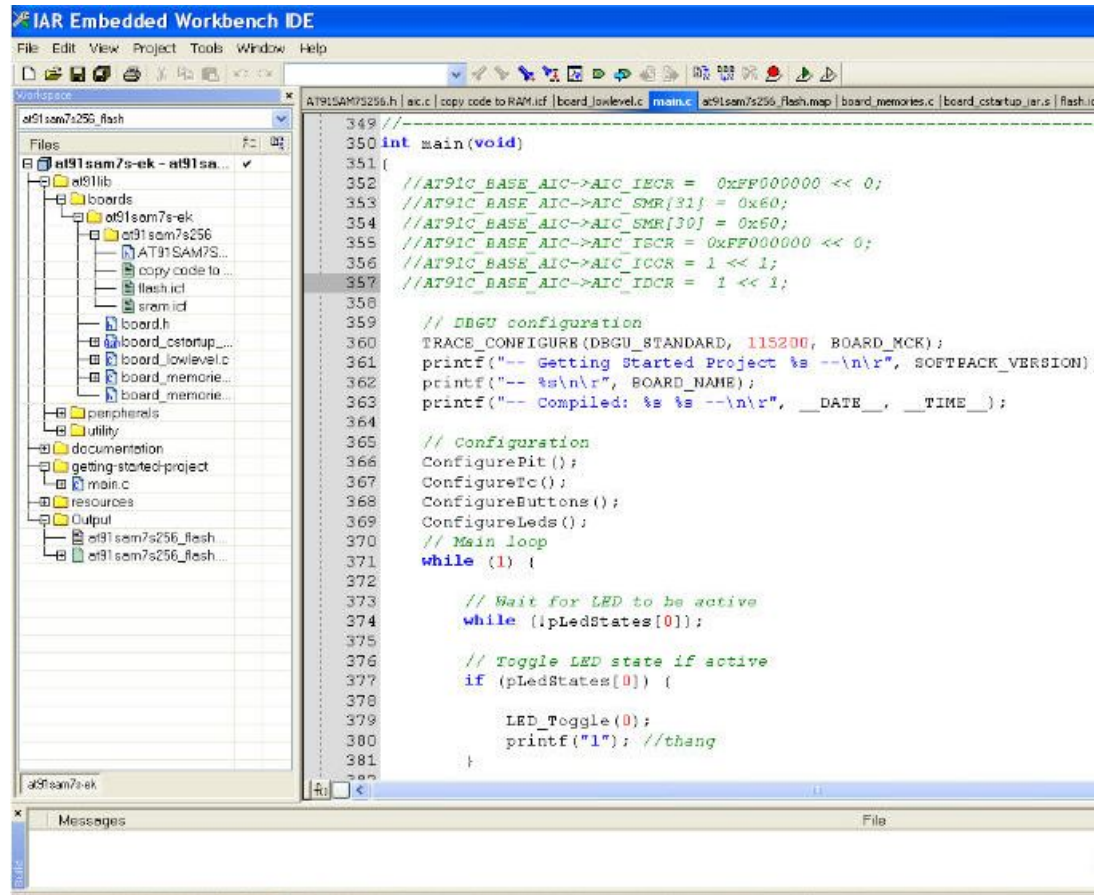
Настройк  
и  
проекта

- ✓Интегрированный компилятор C/C++;
- ✓Интегрированный симулятор;
- ✓При помощи плагина возможна поддержка компилятора GCC в виде сборки WinAVR;
- ✓Поддержка инструментов Atmel, совместимых с 8-разрядной AVR архитектурой0;
- ✓Поддержка плагина AVR RTOS;
- ✓Интерфейс командной строки с поддержкой TPI.



# Микропроцессорные устройства и системы.

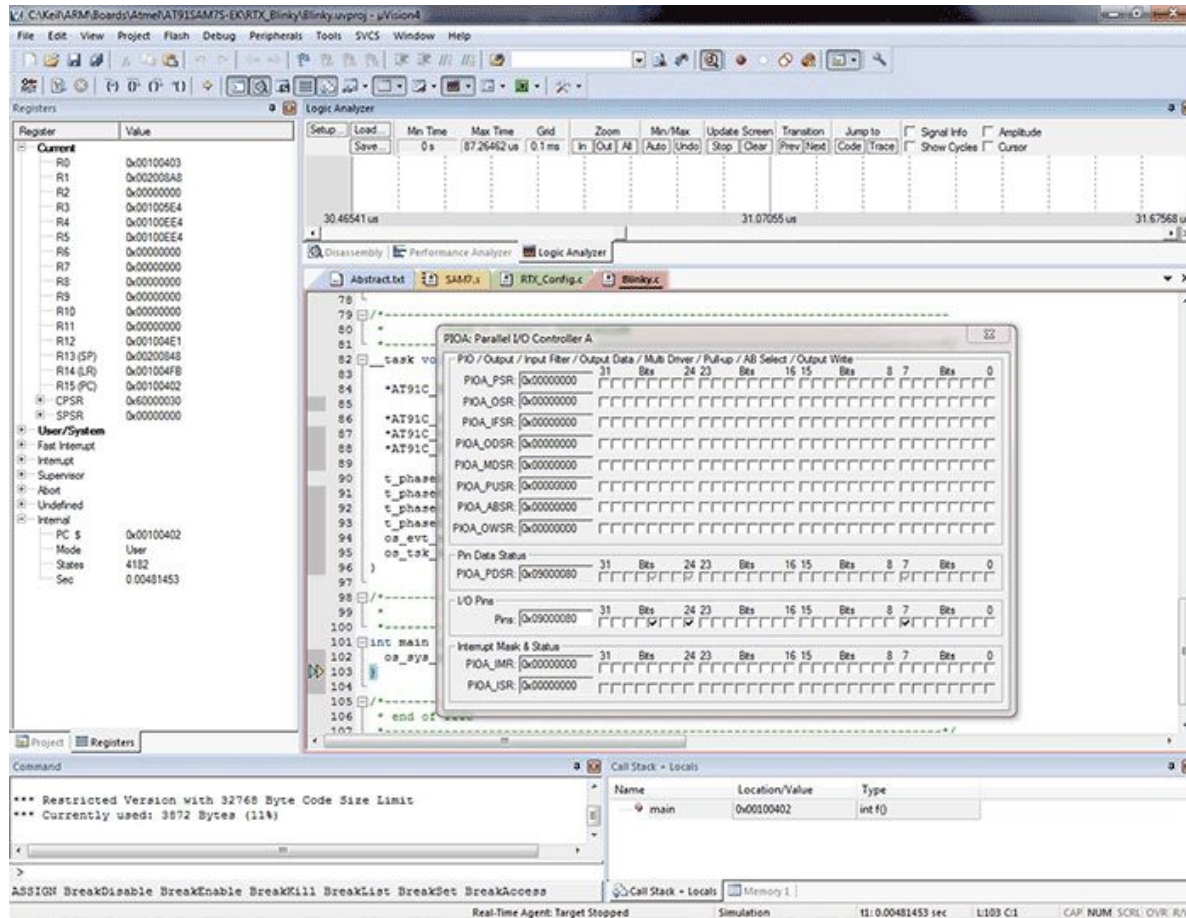
## Компиляторы для микроконтроллеров - IAR Embedded Workbench (платная)



- ✓ Многофункциональная среда разработки приложений на языках C, C++ и ассемблере для целого ряда микроконтроллеров от различных производителей.
- ✓ Интегрированный компилятор C/C++;
- ✓ Поддерживает работу с 8-, 16-, 32-разрядными микроконтроллерами от Atmel, ARM, NEC, Infineon, Analog Devices, Cypress, Microchip Technologies, Micronas, Dallas Semiconductor/Maxim, Ember, Luminary, NXP, OKI, Samsung, National Semiconductor, Texas Instruments, STMicroelectronics, TI/Chipcon, Silicon Labs...

# Микропроцессорные устройства и системы.

## Компиляторы для микроконтроллеров - Keil uVision (платная)



- ✓Среда программирования разработана компанией Keil, которая была основана в Мюнхене в 1982 году братьями Гюнтером и Рейнхардом. В октябре 2005 года Keil вошла в состав американской корпорации ARM.
- ✓Интегрированный компилятор C/C++;
- ✓Макроассемблер,
- ✓Отладчики, симуляторы, линкеры, IDE-приложения

# Микропроцессорные устройства и системы.

## Программаторы для микроконтроллеров AVR

Программирование и  
отладка



AVR ONE!

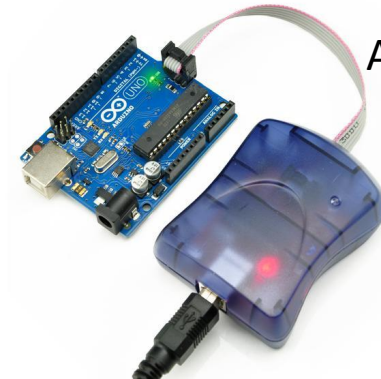
JTAG ICE mkII



Программировани  
е AVR ISP mkII



AVR ISP



Простейший AVR USB  
Программатор  
USBASP

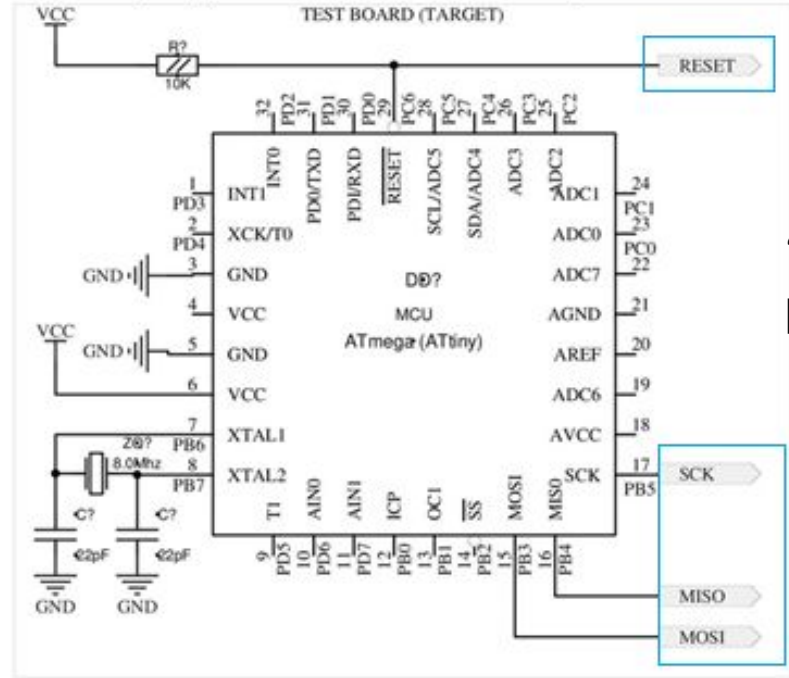


# Микропроцессорные устройства и системы.

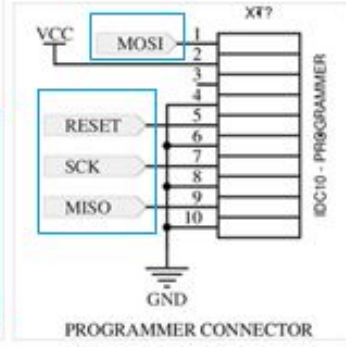
## Загрузка программы в микроконтроллер используя внутрисхемное программирование (ISP)



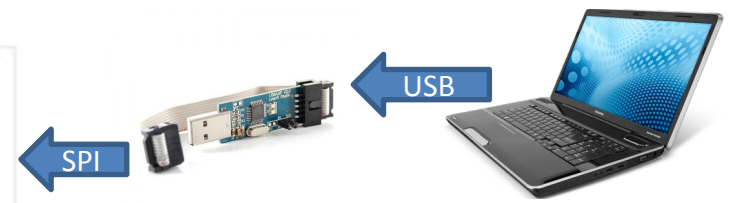
Пример подключения программатора USBASP к микроконтроллеру AVR



“ISP  
разъем”



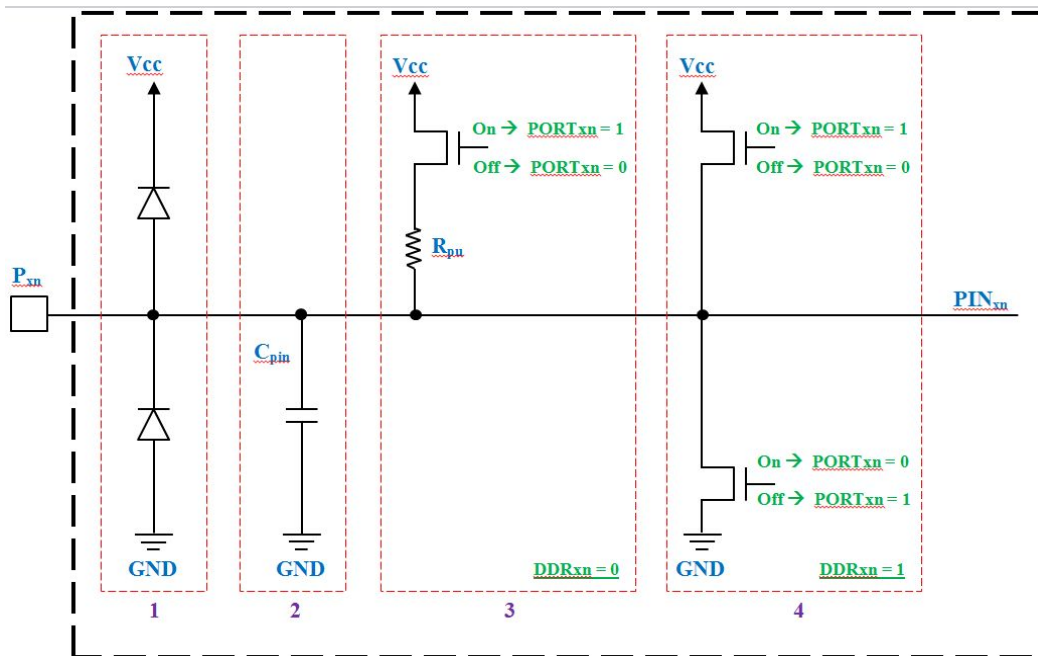
Программатор AVR ISP mkII  
USB AVR программатор  
(USBASP)





# Микропроцессорные устройства и системы.

Общие сведения о портах ввода/вывода atmega 16



DDRB = 0b10000001; или DDRB |= 0x01;  
PORTB |= (1<<PB0); или PORTB = 1;

MyKeyboard = PINB; // читаем состояние порта  
B

DDRx – Настройка разрядов  
порта x на вход или выход.

Весь порт настроен на

выл 

1	1	1	1	1	1	1	1
---	---	---	---	---	---	---	---

Вес 

0	0	0	0	0	0	0	0
---	---	---	---	---	---	---	---

 д  
=0

PORTx – Управление  
состоянием выходов порта x

PINx – Чтение логических

ур 

1	0	0	0	0	0	1	1
---	---	---	---	---	---	---	---

На вывод 0,1,7 порта подана 1

# Микропроцессорные устройства и системы.

## Общие сведения о Таймерах-счетчиках atmega 16

Считающий в “фоне” Таймер не нагружает ЦПУ и не сказывается на свободном процессорном времени

### Общие регистры, относящиеся ко всем трем таймерам ATme

TIMSK - конфигурационный регистр

TIFR - статусный регистр флагов

SFIOR - регистр специальных функций

### Для таймера 0 (8бит)

1. TCNT0 - счетный регистр

2. OCR0 - регистр сравнения

3. TCCR0 - конфигурационный регистр.

### Для таймера 1 (16бит)

1. TCNT1 Timer/Counter1 – регистр, содержащий текущее значение таймера счетчика 1

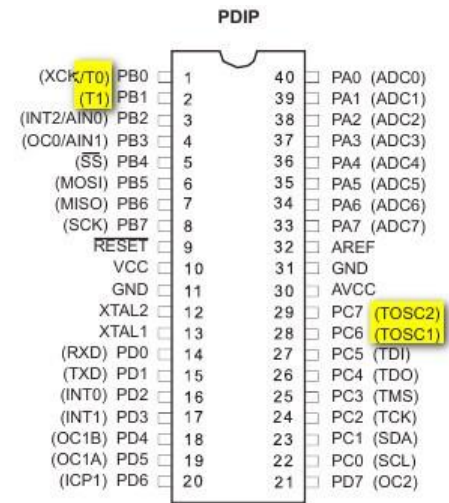
2. TCCR1A Timer/Counter1 Control Register A – регистр задания режимов таймера/счетчика 1

3. TCCR1B Timer/Counter1 Control Register B – регистр задания режимов таймера/счетчика 1

4. OCR1A Timer/Counter Output1 Compare Register A – выходной регистр компаратора A

5. OCR1B Timer/Counter Output1 Compare Register B – выходной регистр компаратора B

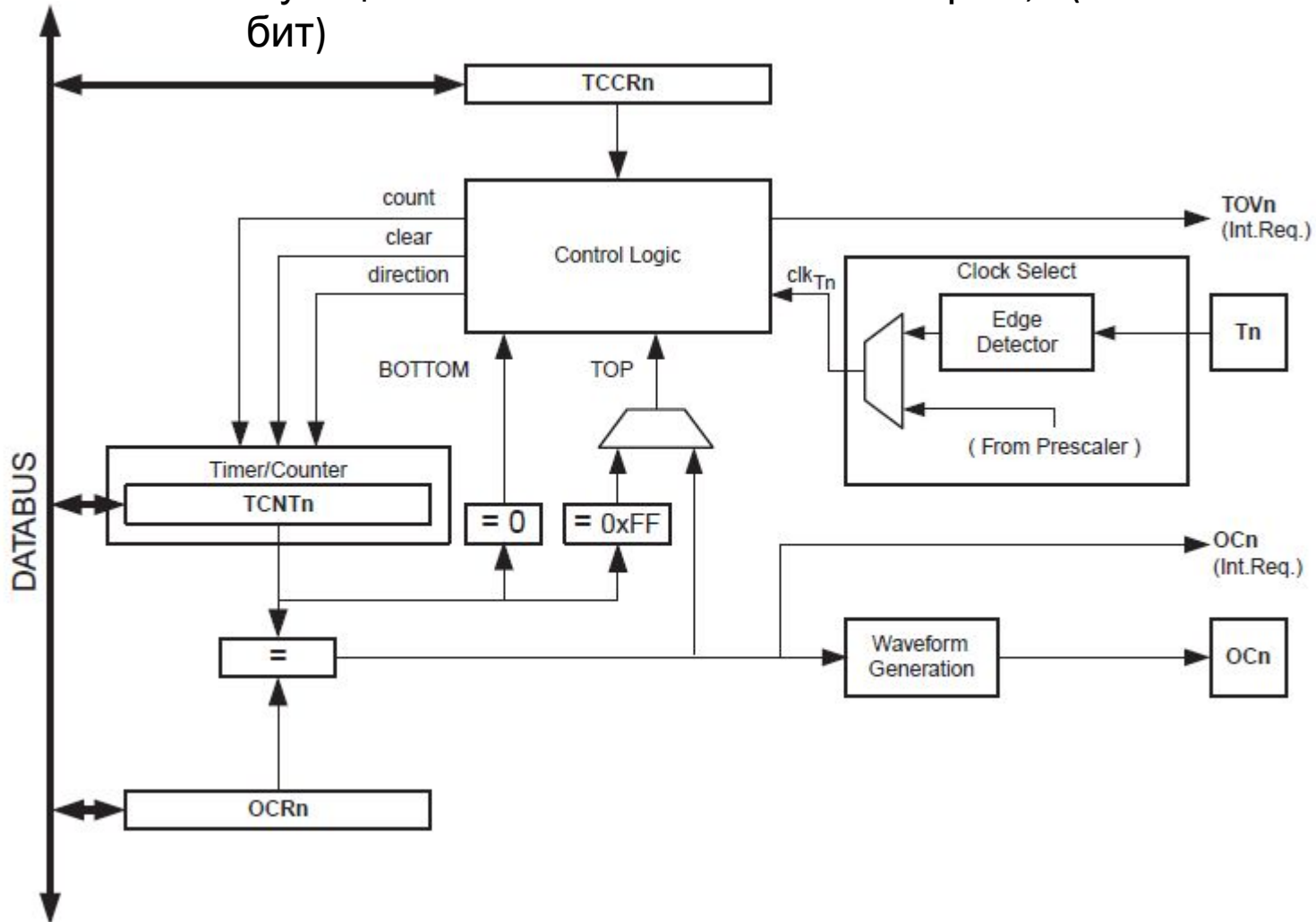
6. ICR1 Timer/Counter1 Input Capture Register1 – входной регистр защелки 1-го таймера



# Микропроцессорные устройства и системы.

## Общие сведения о Таймерах-счетчиках atmega 16

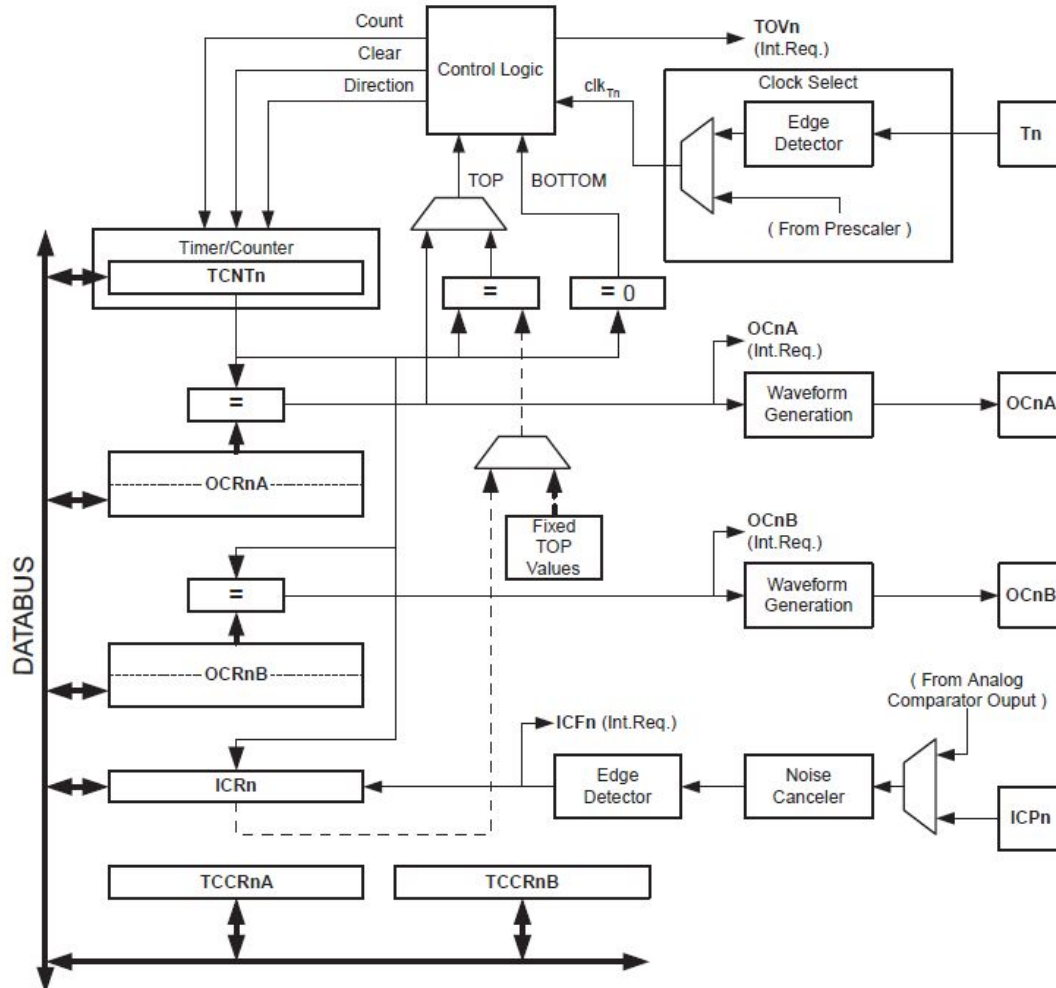
Функциональная блок схема Таймера 0,2 (8-бит)



# Микропроцессорные устройства и системы.

## Общие сведения о Таймерах-счетчиках atmega 16

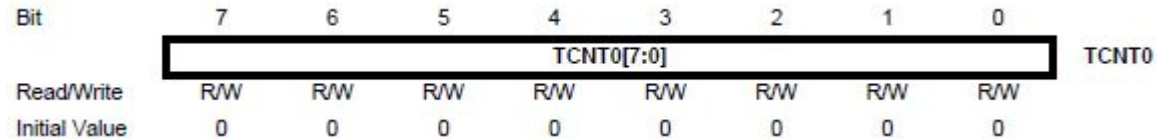
Функциональная блок схема Таймера 1 (16-  
бит)



# Микропроцессорные устройства и системы.

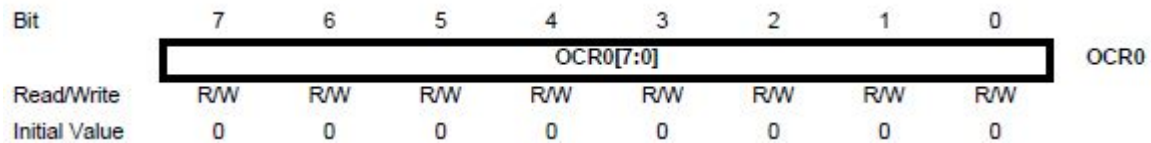
## Таймеры счетчики

### TCNT0



*Это 8-ми разрядный счетный регистр. Когда таймер работает, по каждому импульсу тактового сигнала значение TCNT0 изменяется на единицу.*

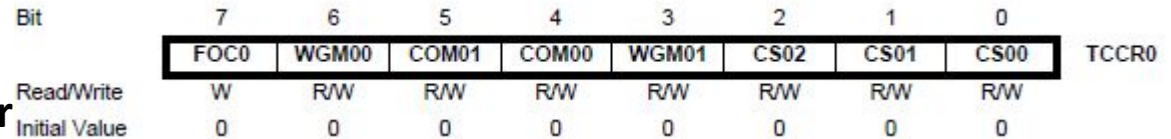
### OCR0



*Это 8-ми разрядный регистр сравнения. Его значение постоянно сравнивается со счетным регистром TCNT0, и в случае совпадения таймер может выполнять какие-то действия*

### TCCR0

**(Timer/Counter Control Register)**



*Это конфигурационный регистр таймера-счетчика T0*

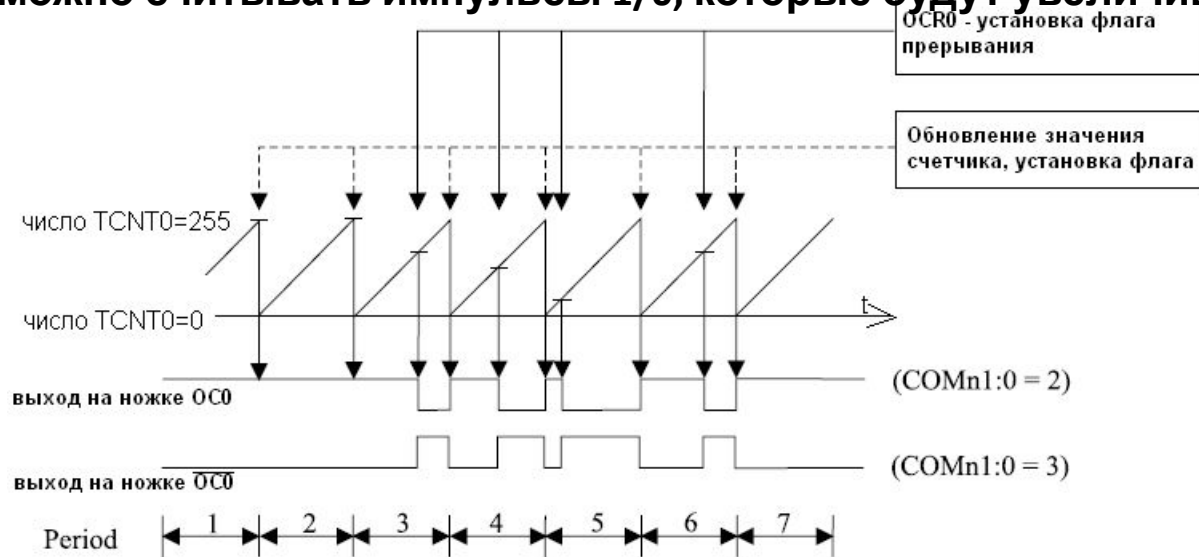
# Микропроцессорные устройства и системы.

## Таймеры счетчики

TCNT0, TCNT1, TCNT2 - При работающем таймере  $T0$  в TCNT0 будет прибавляться/вычитаться 1 (единица)  $TCNT0 = 0 \dots 255$  (1111 1111 b)

OCR0 Регистр сравнения. Число, записанное в OCR0 постоянно сравнивается с TCNT0.

1. Счет в TCNT0,1,2 происходит от мин. 0 до макс. 255 (для 8бит ) или 65 535(для 16бит таймеров).
2. Происходит сброс TCNT0,1,2 при достижении макс.значений + установка флага.
3. При равенстве TCNT0 = OCR0 выставляется флаг и/или прерывание.
4. На ножку OC0 можно назначить авто.вывод 1 или 0 если TCNT0 > OCR0 или TCNT0 < OCR0 (решим ШИМ).
5. С ножки OC0 можно считывать импульсы 1/0, которые будут увеличивать TCNT0.



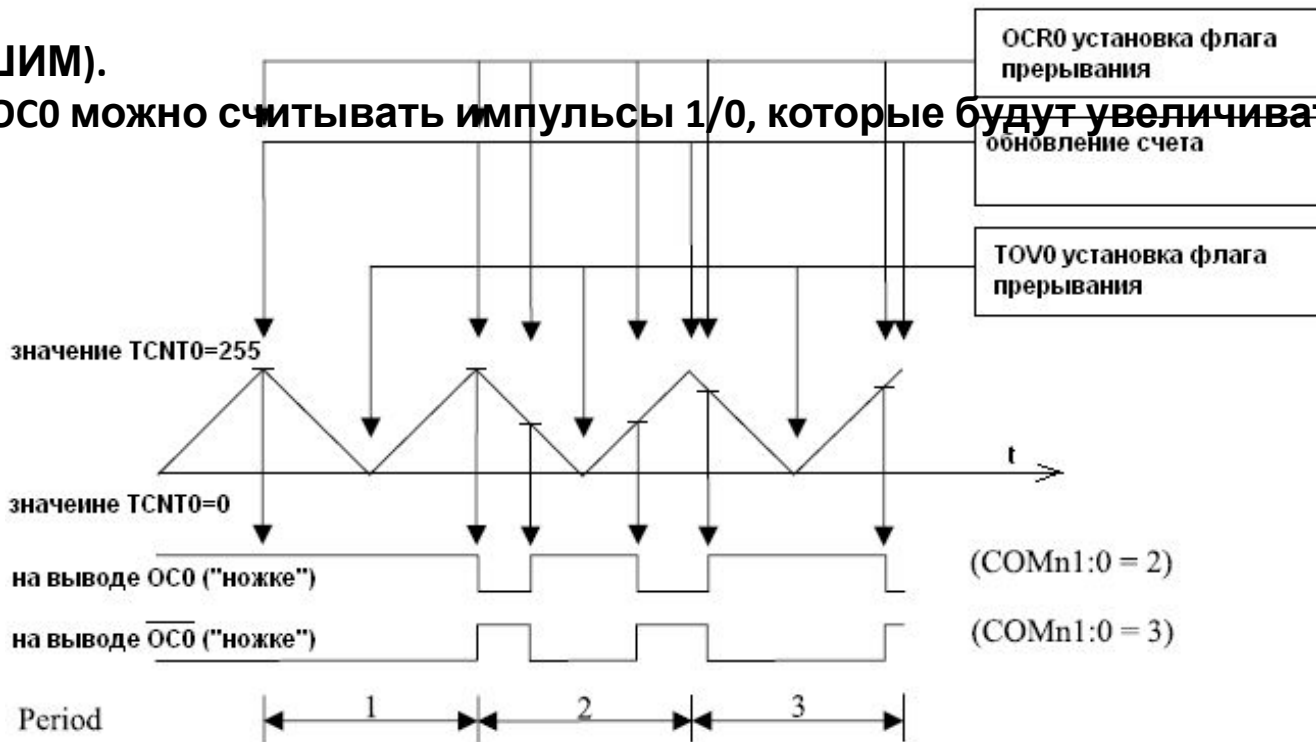
Нормальный режим,  
Режим быстрой широтно-импульсной  
модуляции

# Микропроцессорные устройства и системы.

## Таймеры счетчики

1. Счет в TCNT0 происходит от мин. 0 до макс. 255 (для 8бит ) или 65 535(для 16бит таймеров).
2. Происходит изменение счета TCNT0 при достижении макс.значений + установка флага.
3. При равенстве TCNT0 = OCR0 выставляется флаг и/или прерывание.
4. На ножку OC0 можно назначить авто.вывод 1 или 0 если TCNT0 > OCR0 или TCNT0 < OCR0 (режим ШИМ).

5. С ножки OC0 можно считывать импульсы 1/0, которые будут увеличивать TCNT0.

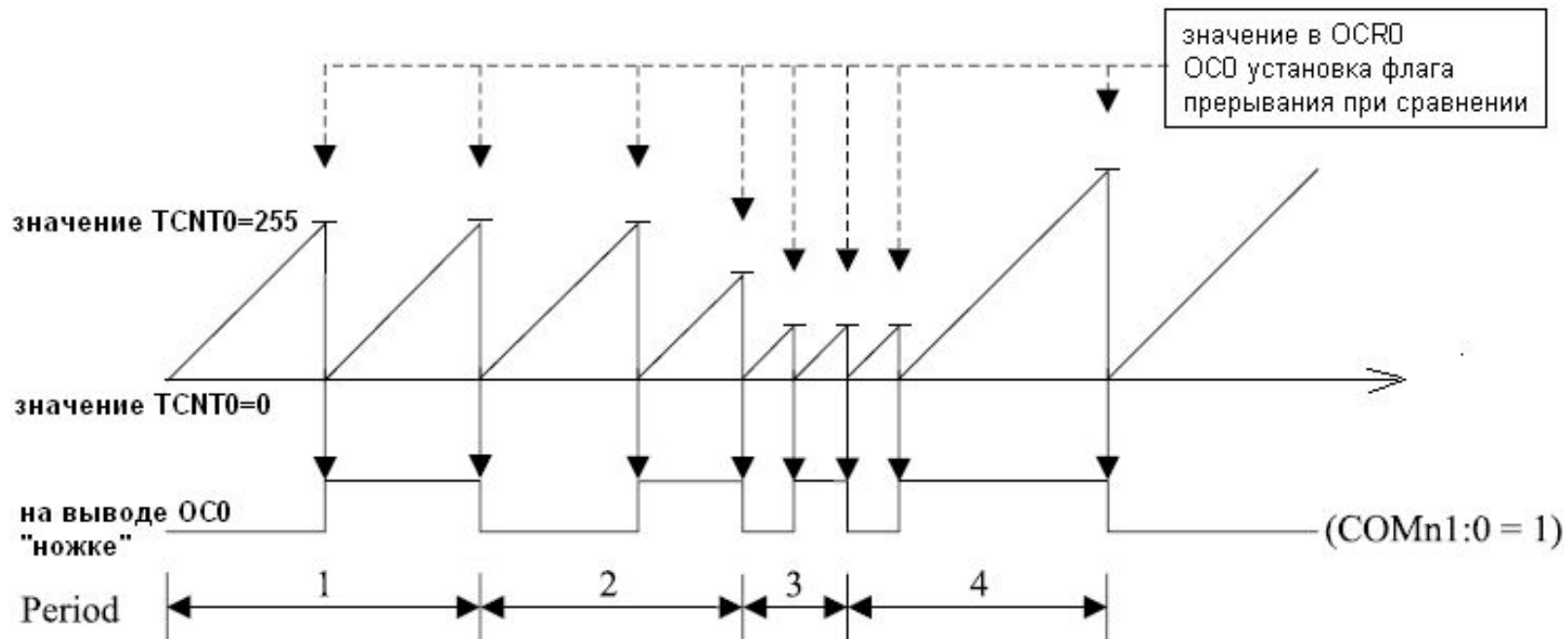


Режим ШИМ с фазовой коррекцией

# Микропроцессорные устройства и системы.

## Таймеры счетчики

1. Счет в TCNT0 происходит от мин. 0 до значения записанного в регистр OCR .
2. Происходит сброс счета TCNT0 при достижении значения OCR + установка флага.
3. При равенстве TCNT0 = OCR0 выставляется флаг и/или прерывание.
4. На ножку OC0 можно назначить авто.вывод 1 или 0 если TCNT0 > OCR0 или TCNT0 < OCR0  
(решим ШИМ).
5. С ножки OC0 можно считывать импульсы 1/0, которые будут увеличивать TCNT0.



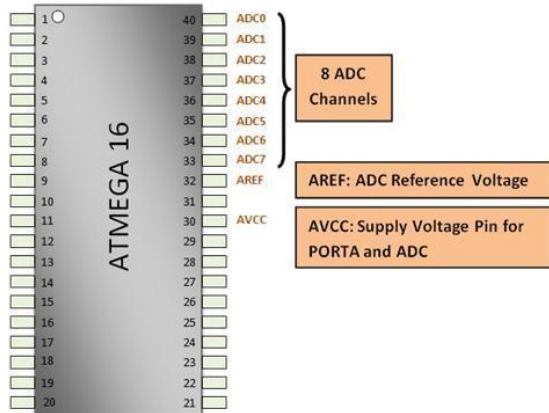
Режим сброса таймера при совпадении  
(CTC)



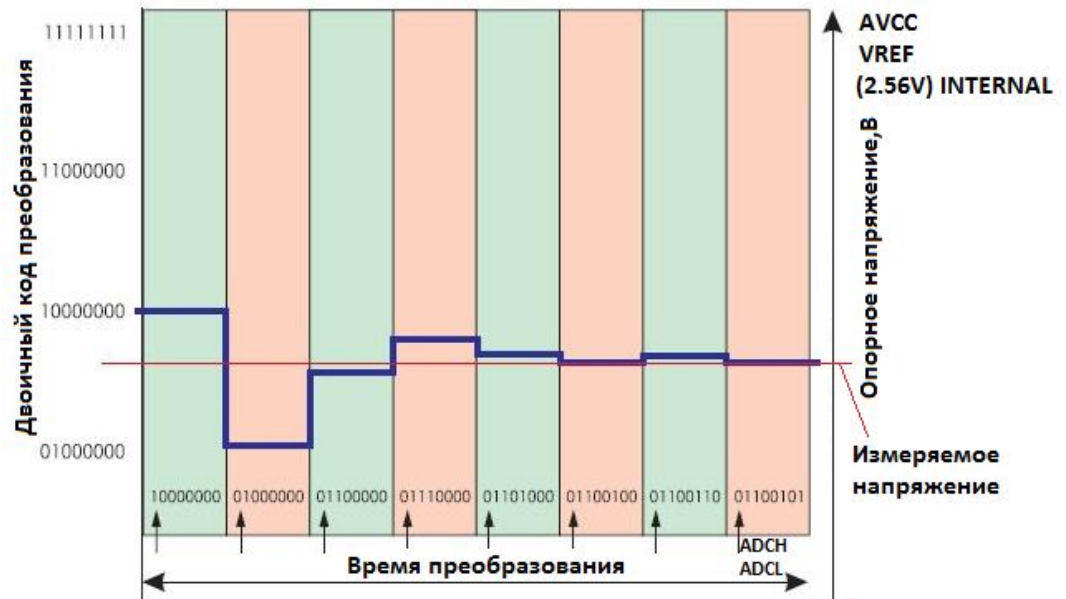
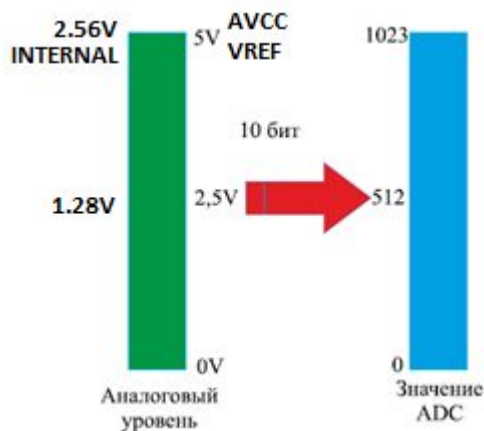
# Микропроцессорные устройства и системы.

## Модуль АЦП – аналого-цифровой преобразователь

(Analog to Digital Converter) AT mega16



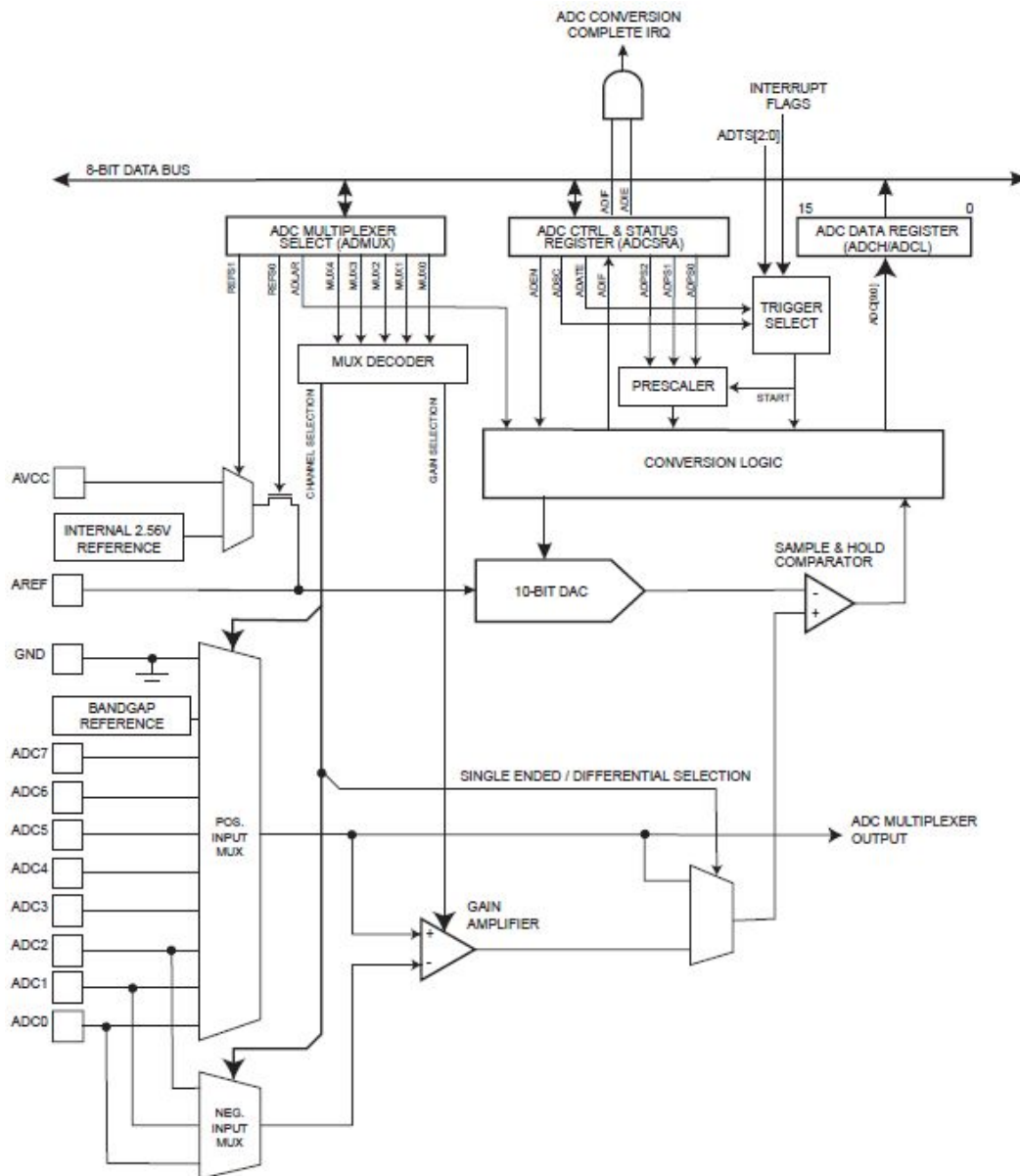
- ✓ В процессоре 1 АЦП модуль позволяет мультиплексировать 8 выводов
- ✓ Последовательное измерение 1 сигнала с выводов ADC0-ADC7



# Микропроцессорные устройства и системы.

## Модуль АЦП – аналого-цифровой преобразователь

(Analog to Digital Converter) AT mega16



Основные характеристики:

- 10-разрядное разрешение.
- Интегральная нелинейность 0.5 мл. разр.
- Абсолютная погрешность  $\pm 2$  мл. разр.
- Время преобразования 13–260 мкс.
- Частота преобразования до 15 тыс. преобразований в секунду при максимальном разрешении.
- 8 мультиплексированных однополярных каналов (входов).
- 7 дифференциальных каналов (входов).
- 2 дифференциальных канала (входа) с подключаемым усилением на 10 и 200.
- Представление результата с левосторонним или правосторонним выравниванием в 16-разрядном слове.
- Диапазон входного напряжения ADC  $0 \dots V_{CC}$ .
- Выборочный внутренний ИОН (Reference Voltage) на 2,56 В.
- Режимы одиночного преобразования и автоматического перезапуска.
- Прерывание по завершении преобразования ADC.
- Механизм подавления шумов в режиме сна.

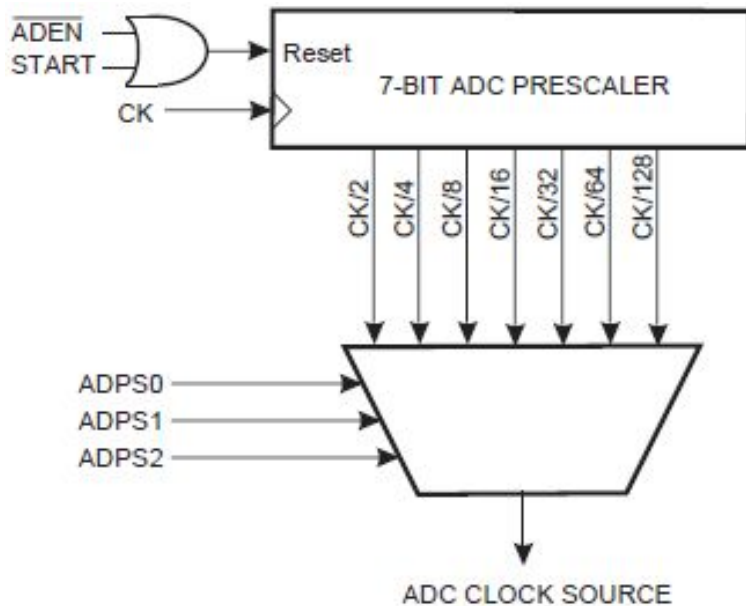


# Микропроцессорные устройства и системы.

## Модуль АЦП – аналого-цифровой преобразователь

(Analog to Digital Converter) AT mega16

### Изменение скорости работы АЦП



### ADCSRA (ADC Control and Status Register)

Bit	7	6	5	4	3	2	1	0	
	ADEN	ADSC	ADATE	ADIF	ADIE	ADPS2	ADPS1	ADPS0	ADCSRA
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

ADPS2	ADPS1	ADPS0	Division Factor
0	0	0	2
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

Узел тактового делителя АЦП позволяет изменить скорость/частоту работы АЦП. При более низкой скорости работы АЦП повышается точность измерений

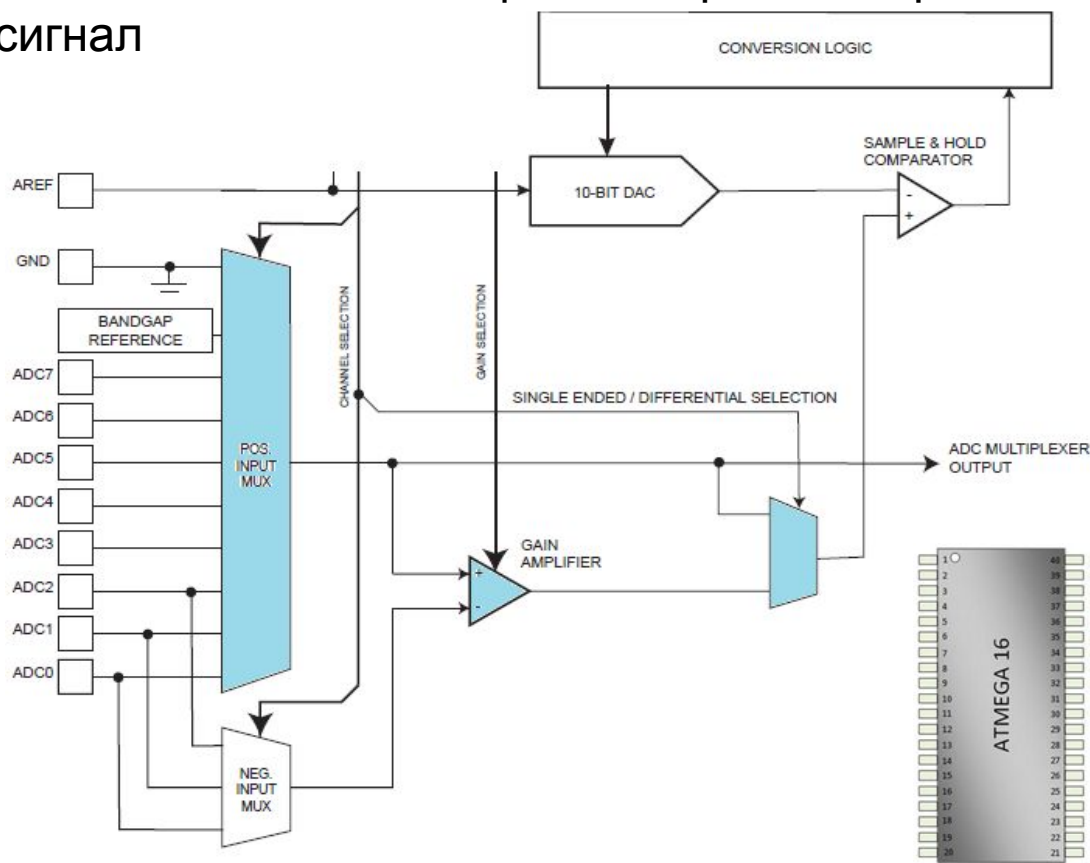
# Микропроцессорные устройства и системы.

## Модуль АЦП – аналого-цифровой преобразователь (Analog to Digital Converter) AT mega16

### ADMUX (ADC Multiplexer Select Register)

Bit	7	6	5	4	3	2	1	0	
	REFS1	REFS0	ADLAR	MUX4	MUX3	MUX2	MUX1	MUX0	ADMUX
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

### Усиление сигнала в АЦП и измерение диф. сигнал



MUX4..0	Single Ended Input	Positive Differential Input	Negative Differential Input	Gain	
00000	ADC0	N/A			
00001	ADC1				
00010	ADC2				
00011	ADC3				
00100	ADC4				
00101	ADC5				
00110	ADC6				
00111	ADC7				
01000	N/A	ADC0	ADC0	10x	
01001		ADC1	ADC0	10x	
01010 <sup>(1)</sup>		ADC0	ADC0	200x	
01011 <sup>(1)</sup>		ADC1	ADC0	200x	
01100		ADC2	ADC2	10x	
01101		ADC3	ADC2	10x	
01110 <sup>(1)</sup>		ADC2	ADC2	200x	
01111 <sup>(1)</sup>		ADC3	ADC2	200x	
10000		ADC0	ADC1	ADC1	1x
10001		ADC1	ADC1	ADC1	1x
10010		ADC2	ADC1	ADC1	1x
10011		ADC3	ADC1	ADC1	1x
10100		ADC4	ADC1	ADC1	1x
10101		ADC5	ADC1	ADC1	1x
10110		ADC6	ADC1	ADC1	1x
10111		ADC7	ADC1	ADC1	1x
11000	ADC0	ADC2	ADC2	1x	
11001	ADC1	ADC2	ADC2	1x	
11010	ADC2	ADC2	ADC2	1x	
11011	ADC3	ADC2	ADC2	1x	
11100	ADC4	ADC2	ADC2	1x	
11101	ADC5	ADC2	ADC2	1x	
11110	1.22 V (V <sub>BG</sub> )	N/A			
11111	0 V (GND)				

Узел внутреннего усиления АЦП позволяет программно усилить аналоговый сигнал

Узел дифференциального измерения позволяет анализировать разностный

# Микропроцессорные устройства и системы.

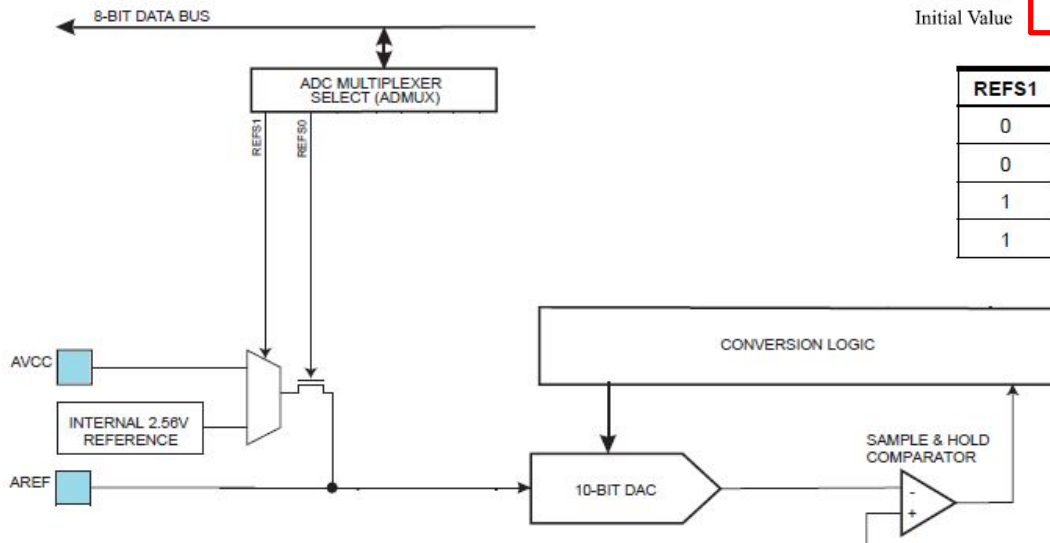
Модуль АЦП – аналого-цифровой преобразователь  
(Analog to Digital Converter) AT mega16

## Выбор опорного напряжения АЦП

### ADMUX (ADC Multiplexer Select Register)

Bit	7	6	5	4	3	2	1	0	ADMUX
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

REFS1	REFS0	Voltage Reference Selection
0	0	AREF, Internal Vref turned off
0	1	AVCC with external capacitor at AREF pin
1	0	Reserved
1	1	Internal 2.56V Voltage Reference with external capacitor at AREF pin



Источники опорного напряжения:

1. Питание АЦП – самое «шумное»
2. Внутреннее опорное = 2,56В. – стабильнее AVCC
3. Внешнее опорное, поданное на ножку AREF. – самое точное

Узел опорного напряжения АЦП позволяет выбрать опорный аналоговый сигнал



# Микропроцессорные устройства и системы.

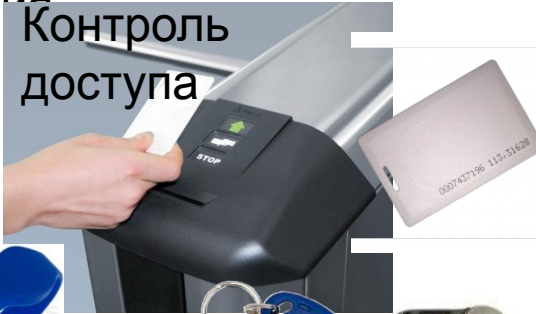
## Модуль UART ATmega16

Современные примеры  
использования

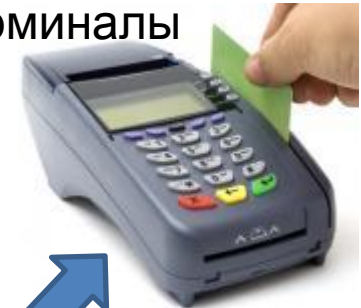
Считыватели магнитных



Контроль  
доступа



Банковские  
терминалы



Микроконтроллер с UART (AtMega, Pic, STM...)



GSM/GPRS чип

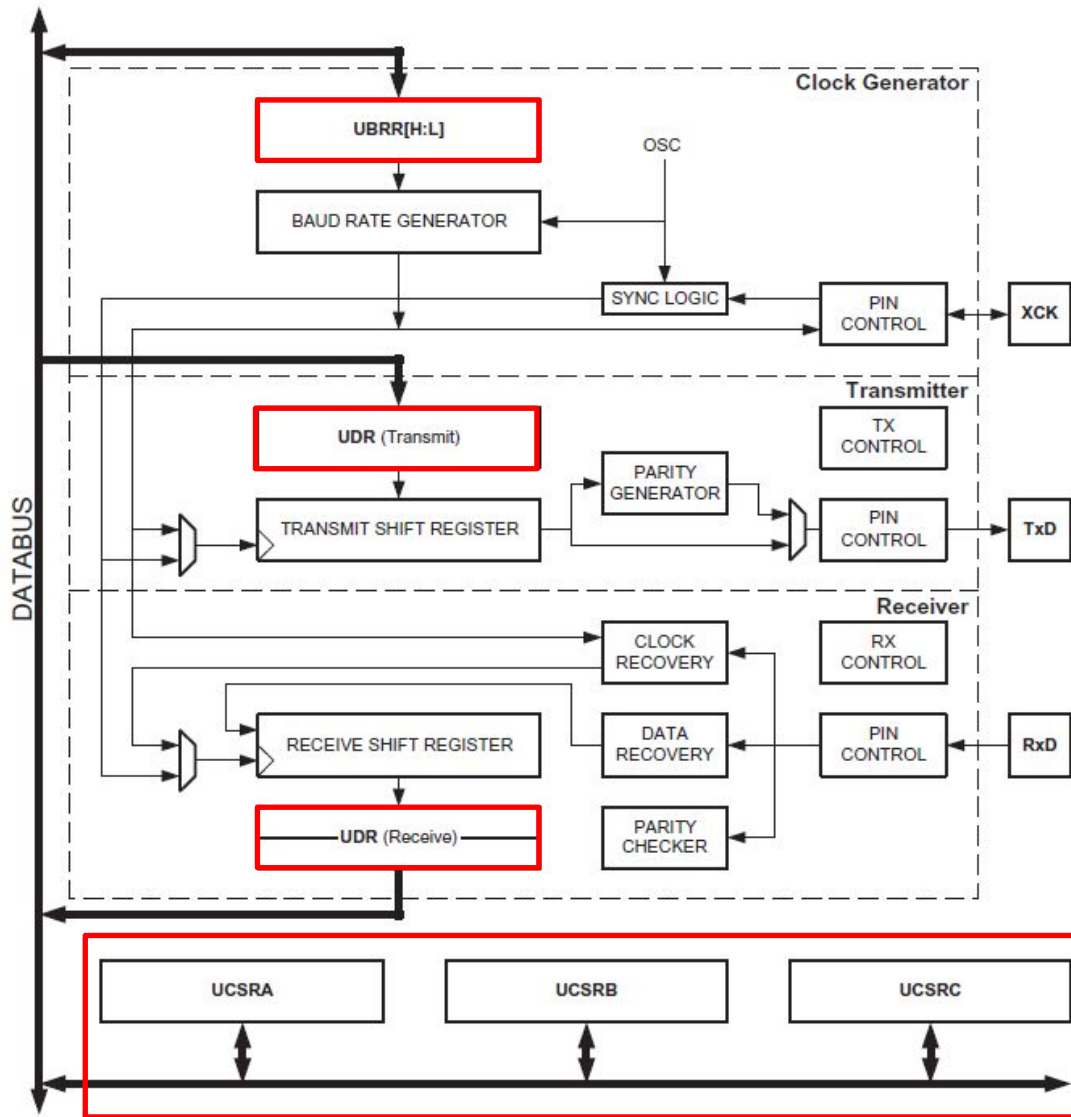


Купюроприемники



# Микропроцессорные устройства и системы.

## Модуль UART ATmega16

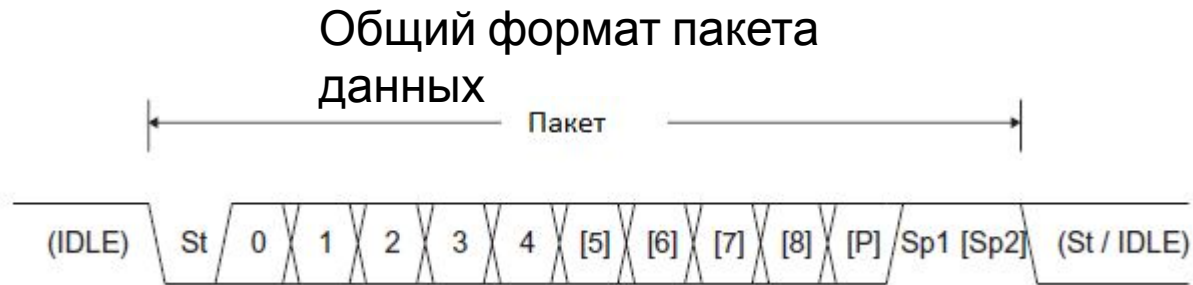


Отвечающие за работу регистры:  
**UDR** - Регистр данных  
**UCSRA** - Управляющий регистр  
**UCSRB** - Управляющий регистр  
**UCSRC** - Управляющий регистр  
**UBRRL** и **UBRRH** - Регистры скорости передачи



# Микропроцессорные устройства и системы.

## Модуль UART ATmega16



1 Старт бит

0,1,2,3,4 – обязательные биты данных

[5],[6],[7],[8],[9] – не обязательные биты данных

[SP2] – стоп бит может быть, либо нет

St - Старт бит, всегда 0

Sp - Стоп бит, всегда 1

P – Бит четности данных может быть либо нет

Типовой (стандартный) формат пакета данных



# Микропроцессорные устройства и системы.

## Модуль UART ATmega16

### Передача 8 бит

данных

```
void UART_Transmit( unsigned char data )
{
    /* Wait for empty transmit buffer */
    while ( !( UCSRA & (1<<UDRE)) )
        ;
    /* Put data into buffer, sends the data */
    UDR = data;
}
```

### Прием 8 бит

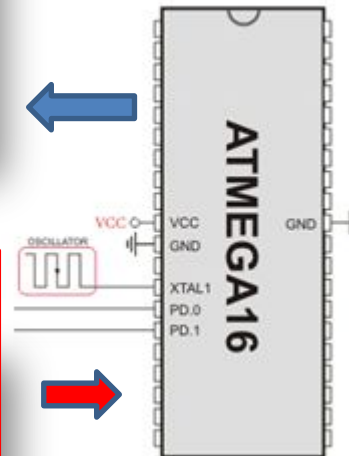
данных

```
unsigned char UART_Receive( void )
{
    /* Wait for data to be received */
    while ( !(UCSRA & (1<<RXC)) )
        ;
    /* Get and return received data from buffer */
    return UDR;
}
```

### Настройка

UART

```
void USART_Init( unsigned int baud )
{
    /* Set baud rate */
    UBRRH = (unsigned char) (baud>>8);
    UBRL = (unsigned char)baud;
    /* Enable receiver and transmitter */
    UCSRB = (1<<RXEN) | (1<<TXEN);
    /* Set frame format: 8data, 2stop bit */
    UCSRC = (1<<URSEL) | (1<<USBS) | (3<<UCSZ0);
}
```



# Микропроцессорные устройства и системы.

## Модуль UART ATmega16

### Интерфейс RS-232 (Com Port)

Скорость (Бод) / Длина неэкр.  
кабеля/

110 — 914,4 м

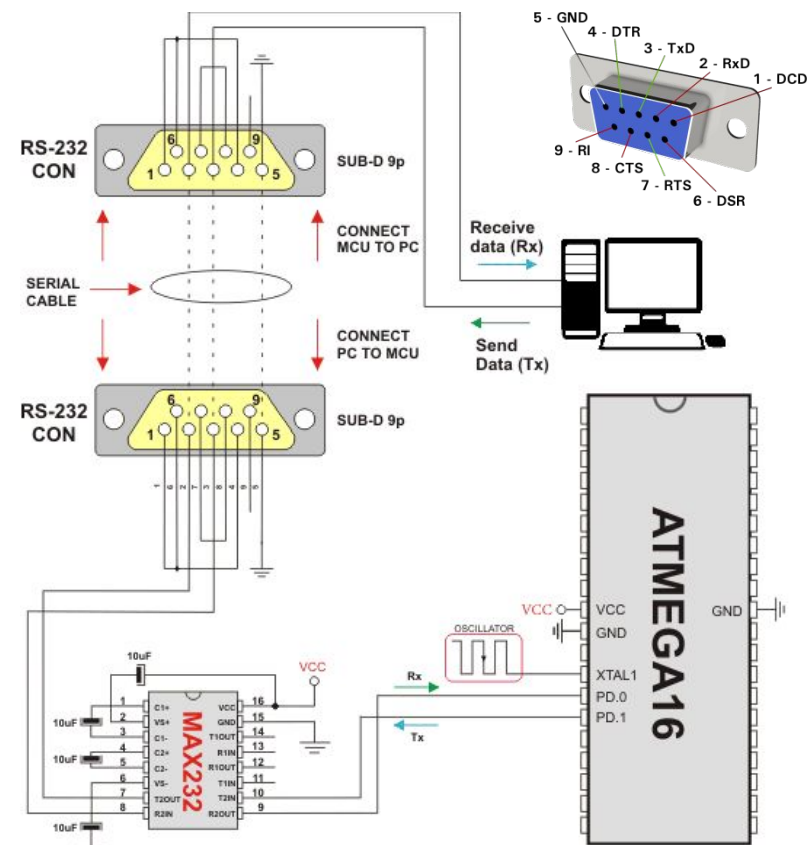
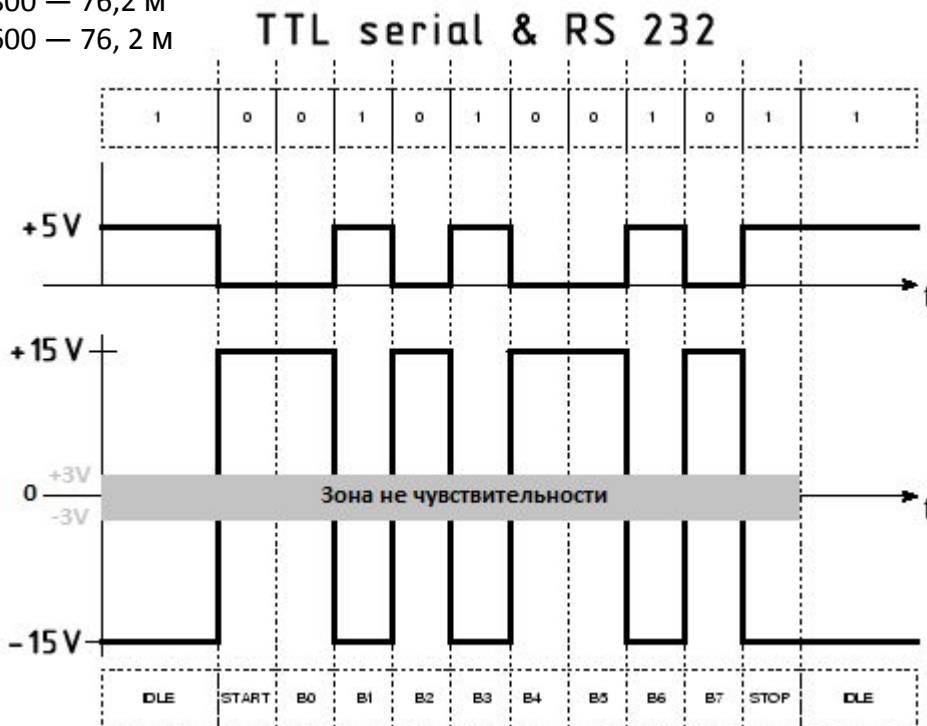
300 — 914,4 м

1200 — 914,4 м

2400 — 152,4 м

4800 — 76,2 м

9600 — 76,2 м

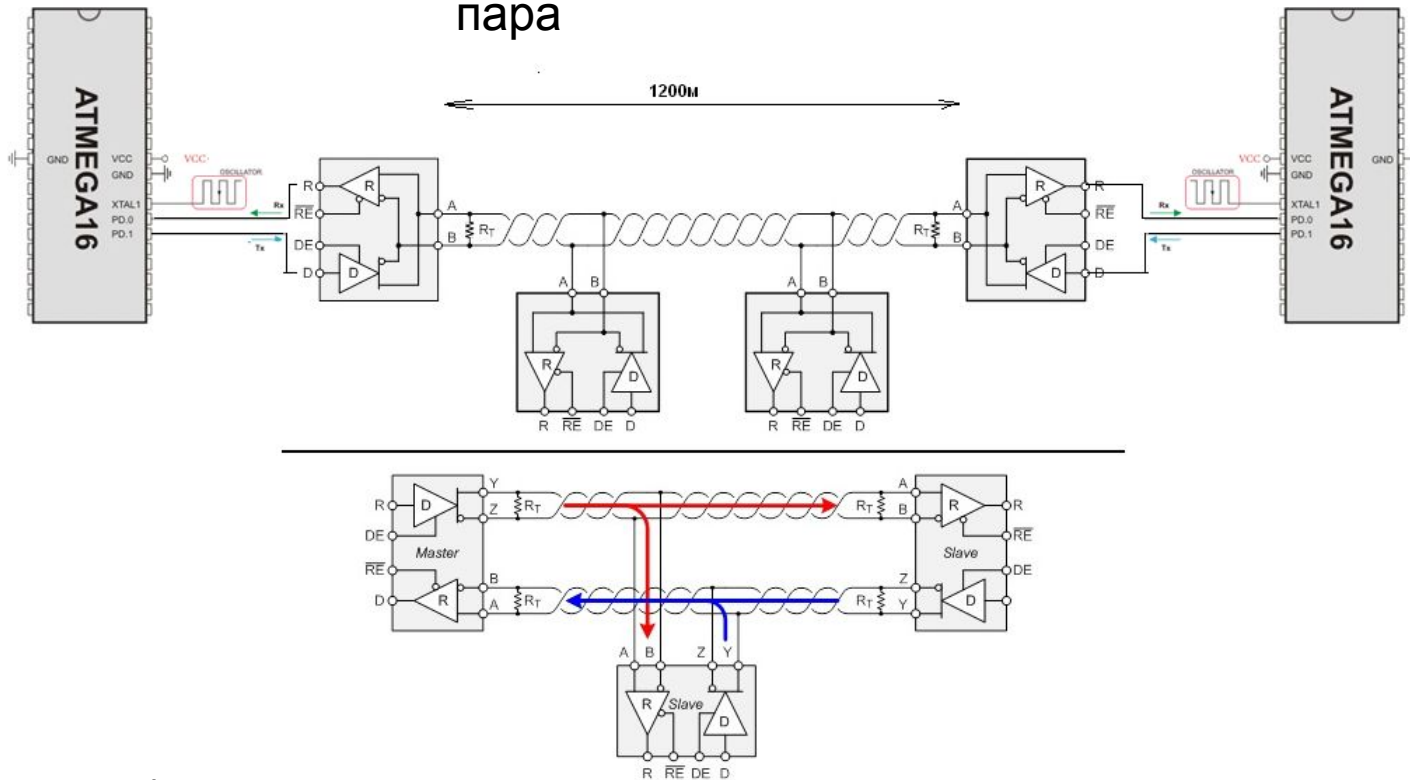


UDR = MyDataByte; // отправить 1байт данных, записав их в регистр UDR  
MyDataByte = UDR ; // получить 1байт данных, прочитав их из регистра UDR

# Микропроцессорные устройства и системы.

## Интерфейс RS-485/422. Прототип Ethernet

Среда передачи – витая пара



62,5 кбит/с 1200 м (одна витая пара),  
375 кбит/с 500 м (одна витая пара),  
500 кбит/с,  
1000 кбит/с,  
2400 кбит/с 100 м (две витых пары),  
10000 кбит/с 10 м

# Микропроцессорные устройства и системы.

## Интерфейс RS-485/422. Прототип Ethernet

Стандарт	EIA/TIA-485 (RS-485)
Физическая среда	Витая пара
Сетевая топология	Точка-точка, Multi-dropped, Multi-point
Максимальное количество устройств	32 — <b>256</b> устройств (32 нагруженных)
Максимальное расстояние	1200 метров
Режим передачи	Дифференциальный сигнал (балансный)
Максимальная скорость передачи	100 кбит/с — <b>10 Мбит/с</b>
Напряжение	-7 В до +12 В
(1)	(A-B) > +200 мВ (положительное напряжение)
(0)	(A-B) < -200 мВ (отрицательное напряжение)
Сигналы	Tx+/Rx+, Tx-/Rx- (Полудуплексный) Tx+, Tx-, Rx+, Rx- (Дуплексный)

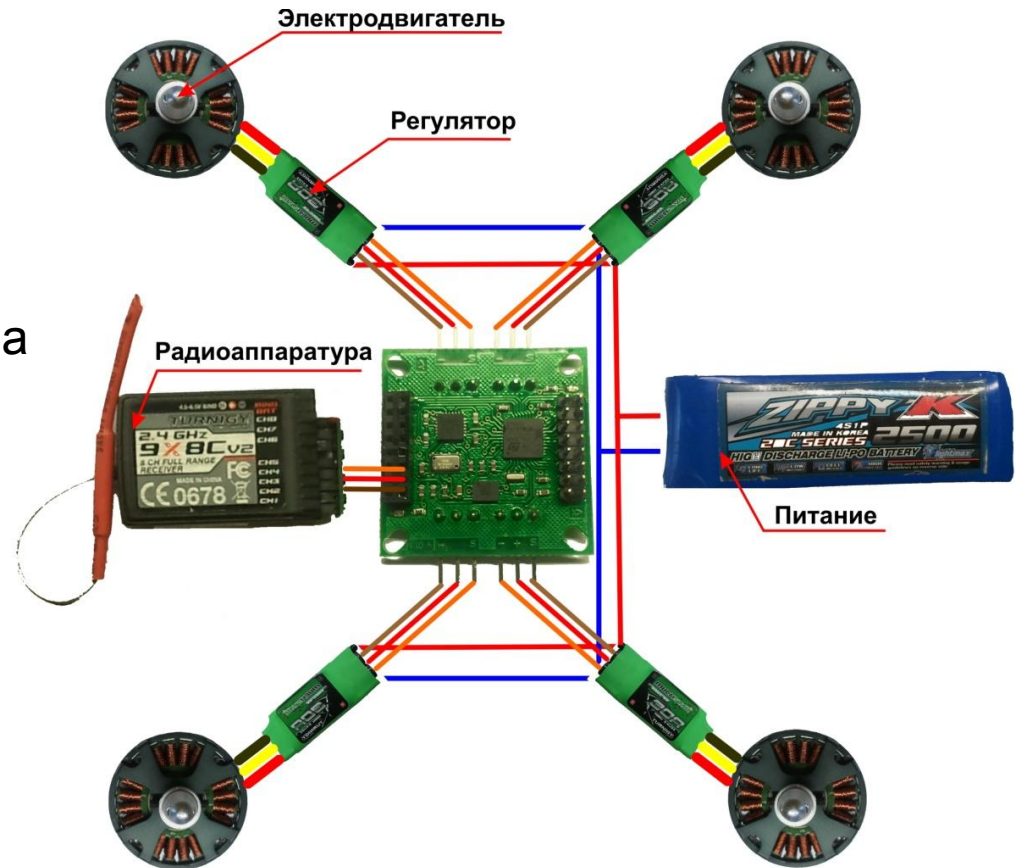
# Микропроцессорные устройства и системы.

## Многозадачность. Особенности. Рекомендации

На примере управления квадрокоптера.

Особенности работы:

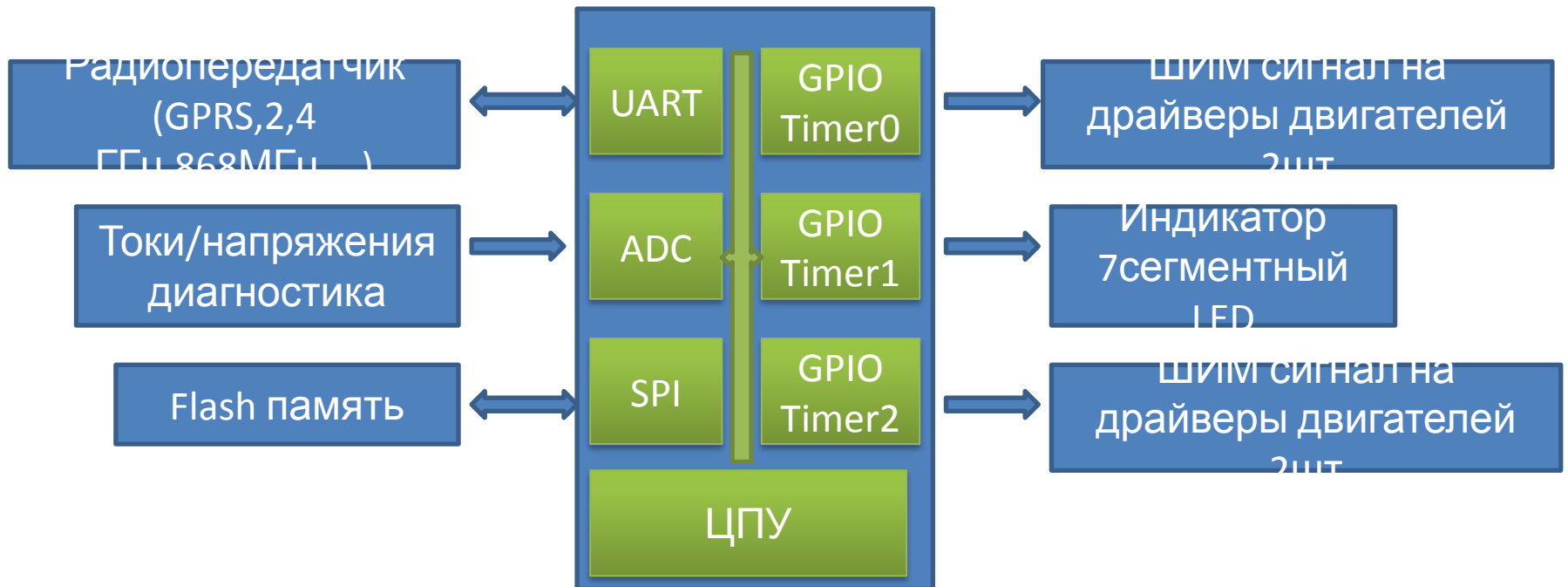
1. Непрерывная выдача сигналов на двигатели .  
*При сбое – падение аппарата.*
2. Непрерывный прием данных с пульта оператора.  
*При сбое – потеря управления.*
3. Мониторинг напряжения аккумулятора.  
*При сбое – разряд.*



# Микропроцессорные устройства и системы.

## Многозадачность. Особенности. Рекомендации

1. Работа с таймерами, UART, ADC.. на прерываниях.  
Максимальное задействие возможностей встроенных модулей.
2. Избегать программной реализации Таймеров, ШИМ, UART..
3. Избегать длительного ожидания флагов/данных, блокирующих работу основной программы
4. Уменьшать время выполнения в процедурах прерываний



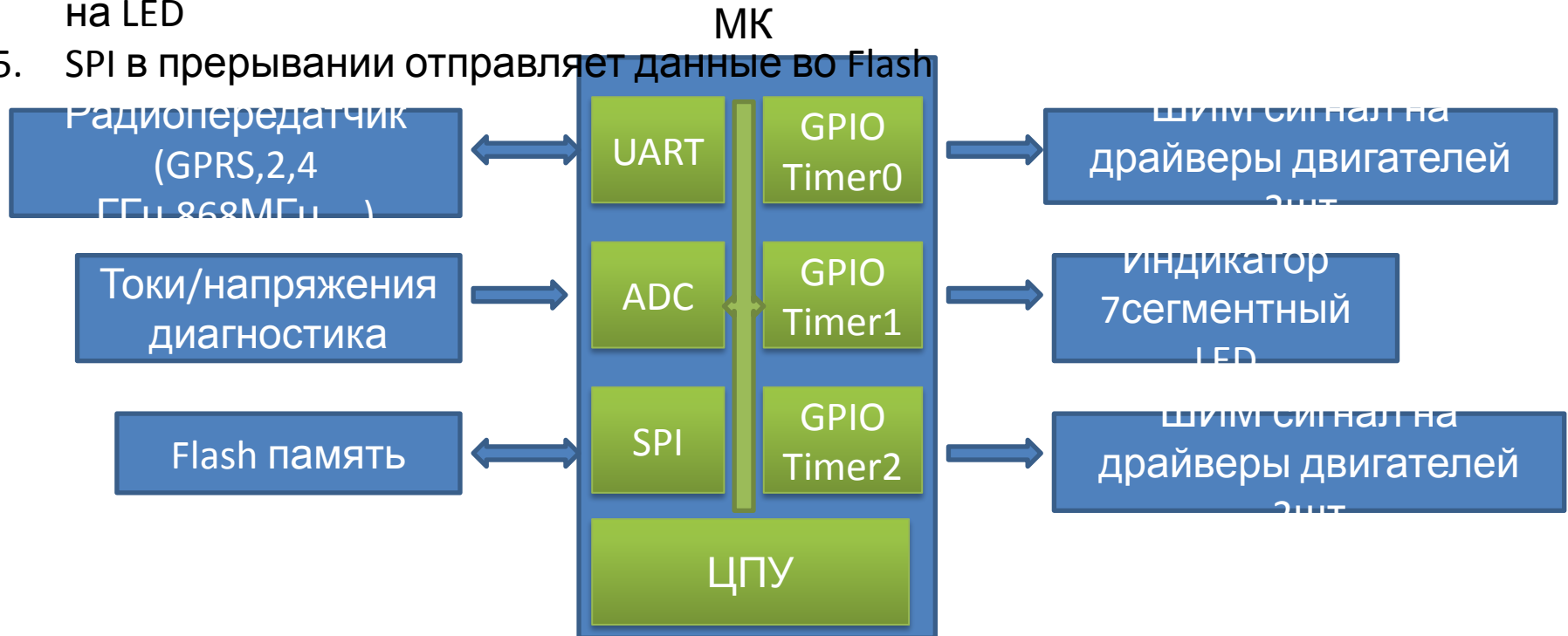
# Микропроцессорные устройства и системы.

## Многозадачность. Особенности. Рекомендации

### Прерывания – спасение в нагруженных

#### задачах

1. Таймер0 формирует ШИМ сигнал управления двигателями, в прерывании
2. UART при приеме данных управления переходит на свое прерывание
3. ADC при окончании измерений в своем прерывании хранит результаты
4. Таймер1, по окончании счета, в своем прерывании обновляет данные на LED
5. SPI в прерывании отправляет данные во Flash

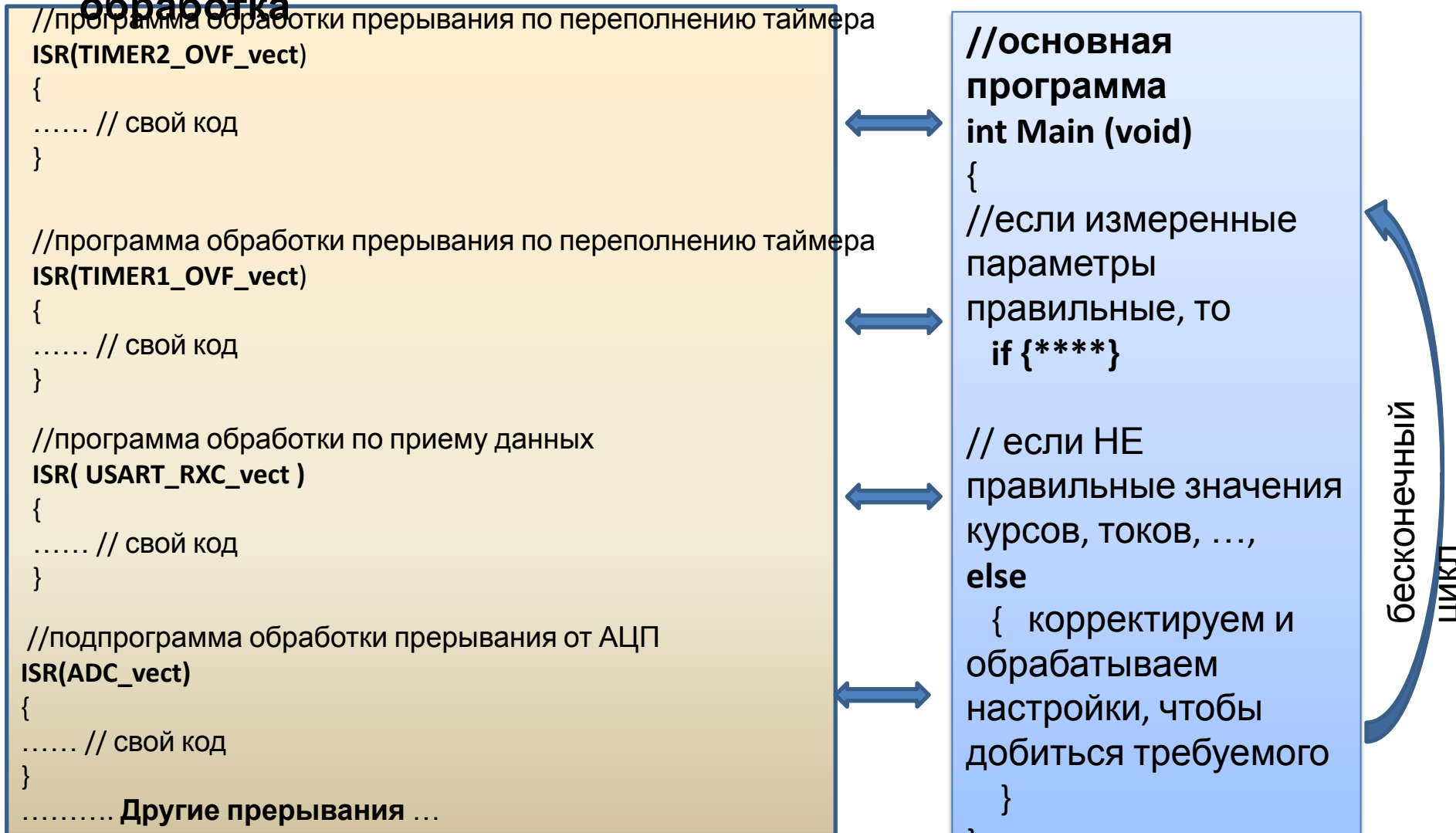




# Микропроцессорные устройства и системы.

Многозадачность. Особенности. Рекомендации

**Прерывания - получение информации, основная программа - обработка**



# Микропроцессорные устройства и системы.

## Заключение

Программирование микроконтроллеров сводится к настройке внутренних модулей через индивидуальные регистры:

1. Портов ввода/вывода (PORTA, PORTB...)
2. Таймеров/счетчиков (T0,T1,T2...)
3. Шин ввода вывода (UART, SPI, i2C, USB...)
4. Преобразователей сигнала (АЦП, ЦАП, Компаратор..)
5. Внутренней памяти (EEPROM, Flash..)
6. ...

и обмена информацией между модулями по различным условиям

- Спасибо за внимание!