

# Микроконтроллер AT90S1200 фирмы Atmel

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

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

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

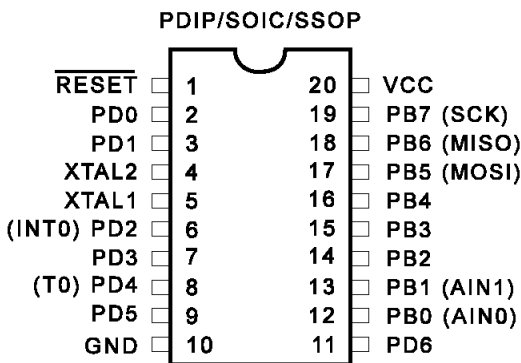
Холостой режим (Idle Mode) отключает ЦПУ, оставляя в рабочем состоянии регистры, таймер/счетчик, сторожевой таймер и систему прерываний.

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

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

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

## ОПИСАНИЕ ВЫВОДОВ



VCC - вывод источника питания

GND - земля

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

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

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

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

XTAL2 - Выход инвертирующего усилителя генератора.

## СПИСОК РЕГИСТРОВ

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

# Набор команд AT90S1200

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

## КОМАНДЫ ВЕТВЛЕНИЯ

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

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