

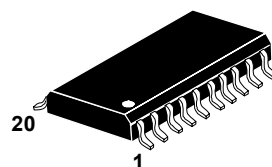
IN90S2313D

8 БИТНЫЙ МИКРОКОНТРОЛЛЕР С 2К БАЙТНОЙ ВСТРОЕННОЙ ФЛЭШ-ПАМЯТЬЮ

Описание

Функциональное назначение

IN90S2313 является низкомоощным КМОП 8-битным микроконтроллером, основанным на AVR улучшенной RISC архитектуре. Посредством выполнения мощных инструкций за один цикл генератора, IN90S2313 достигает пропускных способностей в 1 MIPS на МГц, что предоставляет возможность системному конструктору оптимизировать потребление энергии по отношению к скорости обработки. Ядро AVR сочетает богатый набор команд с 32 рабочими регистрами общего назначения. Все 32 регистра напрямую соединены с арифметическим логическим блоком (АЛБ), что позволяет получать доступ к двум независимым регистрам за одну команду, выполняемую в одном цикле генератора. Результирующая архитектура имеет более эффективный код при достижении пропускных способностей до десяти раз быстрее, чем обычные микроконтроллеры CISC.



Обозначение микросхемы
в корпусе
IN90S2313D
Пластмассовый SO
корпус

IN90S2313 обеспечивает следующие свойства: 2К байтов внутрисистемного программируемого флэша, 128 байтов EEPROM, 128 байтов SRAM, 15 линий входа/выхода общего назначения, 32 рабочих регистра общего назначения, гибкие таймеры / счетчики со сравнительными режимами, внутренними и внешними прерываниями, программируемым последовательным UART, программируемым сторожевым таймером с внутренним генератором, последовательным портом SPI для загрузки флэш памяти и два выбираемых программным обеспечением режима энергосбережения. Холостой режим останавливает центральный процессорный блок, позволяя продолжать функционировать SRAM, таймерам / счетчикам, порту SPI и системе прерываний. Режим снижения мощности сохраняет содержание регистра, но замораживает генератор, отключая все другие функции кристалла до следующего прерывания или сброса аппаратного обеспечения. прибор изготавливается посредством использования технологии высокоплотной неразрушающейся памяти Atmel. Внутрисистемный программируемый флэш на кристалле обеспечивает возможность программной памяти быть перепрограммированной внутри системы посредством последовательного интерфейса SPI или посредством обычного средства программирования неразрушающейся памяти. Посредством сочетания улучшенного RISC 8-битного центрального процессорного блока с внутрисистемным программируемым флэшем на монолитном кристалле, Atmel IN90S2313 является мощным микроконтроллером, который предоставляет высокогибкое и стоимостно-эффективное решение для многих встроенных применений управления.

IN90S2313D

IN90S2313 AVR поддерживается полным набором средств разработки программы и системы, включая: С компиляторы, макроассемблеры, средства отладки программы / моделирования, внутрисхемные эмуляторы, и комплекты оценки.

Свойства

AVR – Высокопроизводительная и низкоомная RISC архитектура;
118 мощных инструкций – большинство исполнения за один цикл генератора;
2К байтов внутрисистемного перепрограммируемого флэша;
SPI последовательный интерфейс для загрузки программы;
Продолжительность: 1,000 циклов записи / стирания;
128 байтов EEPROM;
Продолжительность: 100,000 циклов записи / стираний;
128 байтов внутренней RAM;
32 x 8 рабочих регистра общего назначения;
15 программируемых линий ввода/вывода;
 V_{CC} : 2.7 - 6.0В;
Полностью статическая операция;
0 - 10 МГц, 4.0 - 6.0В;
0 - 4 МГц, 2.7 - 6.0В;
До 10 MIPS пропускная способность при 10 МГц;
Один 8-битный таймер / счетчик с отдельным предварительным делителем частоты;
Один 16-битный таймер / счетчик с отдельным предварительным делителем частоты и режимами сравнения и захвата;
Полнодуплексный UART;
Выбираемая 8, 9 или 10 битная широтно-импульсная модуляция;
Источники внешних и внутренних прерываний;
Программируемый сторожевой таймер с генератором на кристалле;
Аналоговый компаратор на кристалле;
Режимы холостого хода низкой мощности и снижения мощности;
Программируемый замок для безопасности программного обеспечения;
20-выводной прибор.

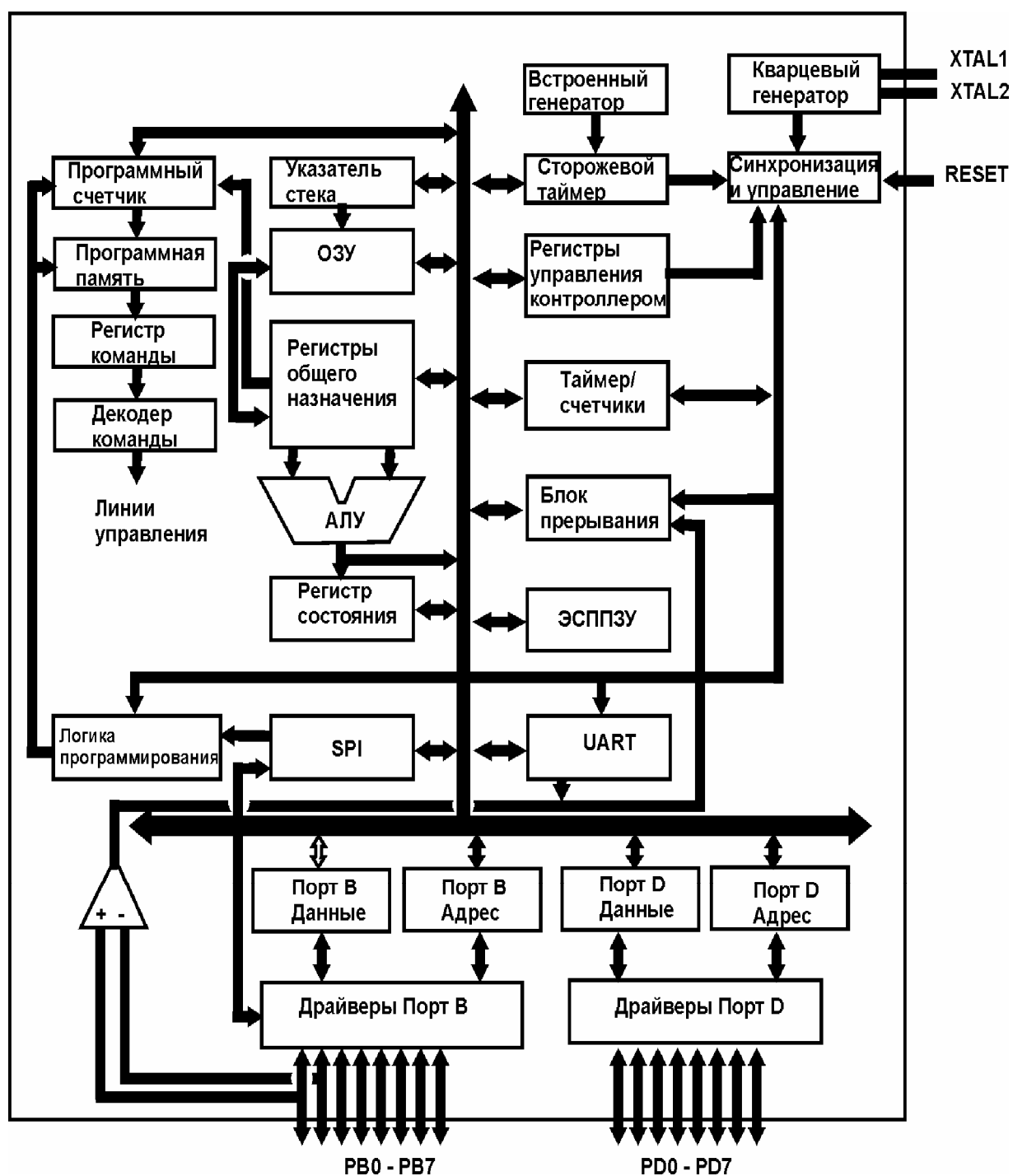


Рисунок 1 - Блок-схема

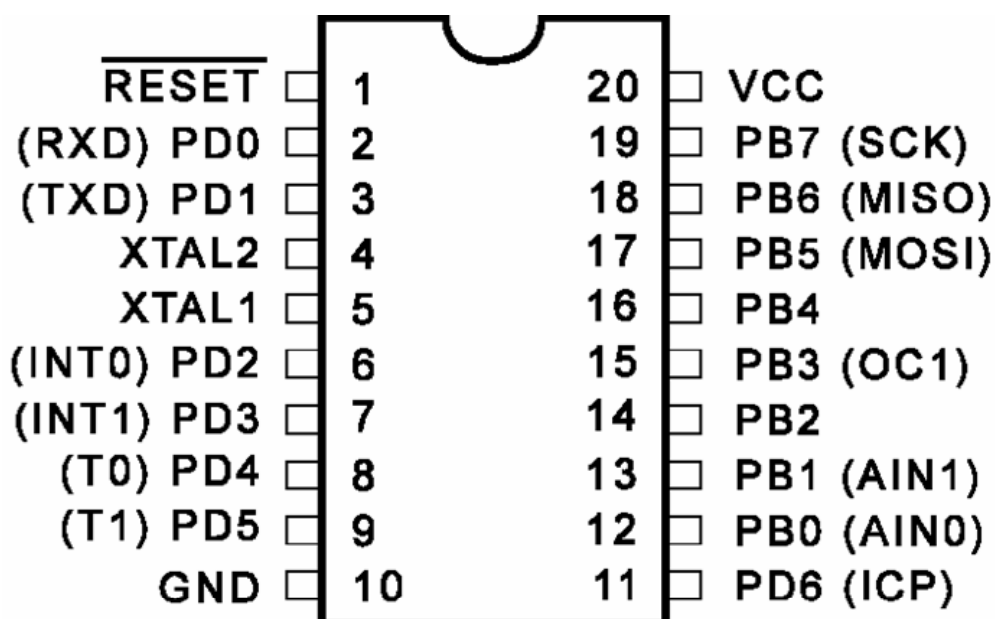


Рисунок 2 - Назначение выводов.

Описание выводов

VCC

Вывод питания напряжения.

GND

Вывод заземления.

Порт В (PB7..PB0)

Порт В является 8-битным двунаправленным портом ввода/вывода. Выводы порта могут обеспечить внутренние подтягивающие резисторы (выбираемые для каждого бита). PB0 и PB1 также служат как положительный вход (AIN0) и отрицательный вход (AIN1), соответственно, аналогового компаратора на кристалле. Выходные буфера порта В могут являться стоком 20 мА и могут быть прямым приводом дисплеев ЖКИ. Когда выводы от PB0 до PB7 используются в качестве входов и внешним образом стягиваются до нижнего уровня, они будут являться источником тока, если внутренние подтягивающие резисторы активированы. Порт В также выполняет функции различных специальных свойств IN90S2313, как указано на стр. 38.

Порт D (PD6..PD0)

Порт D имеет семь двунаправленных выводов ввода/вывода с внутренними подтягивающими резисторами, PD6..PD0. Выходные буферы порта D могут являться стоком 20 мА. Как входы, выводы порта D, которые внешним образом являются стянутыми до нижнего уровня, будут являться источником тока, если активируются подтягивающие резисторы.

Порт D также выполняет функции различных специальных свойств IN90S2313, как указано на стр. 43.

RESET

Вход сброса. Ноль на этом выводе для двух машинных циклов при работе генератора производит сброс прибора.

XTAL1

Вход на инвертирующий усилитель генератора и вход на инвертирующий усилитель генератора и вход на рабочую схему внутреннего генератора.

XTAL2

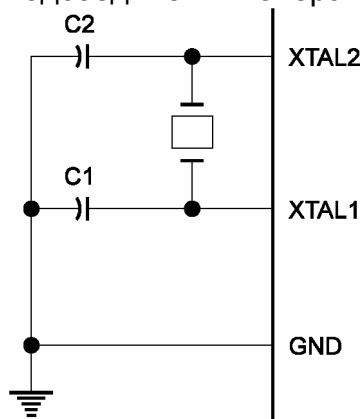
Выход с усилителя инвертирующего генератора.

Кристалльный генератор

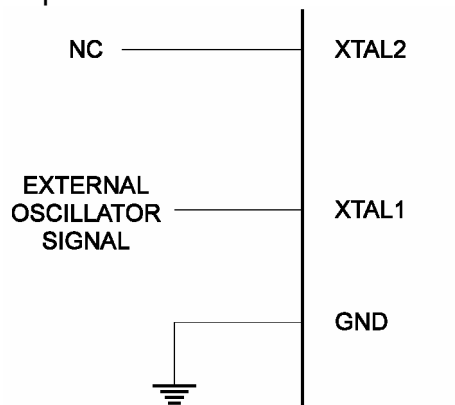
XTAL1 и XTAL2 являются соответственно входом и выходом инвертирующего усилителя, который может быть конфигурирован для использования как генератор на кристалле. Могут быть использованы либо кварцевый кристалл, либо керамический резонатор. Для привода прибора от внешнего источника генератора, XTAL2 должен быть оставлен неподключенным при приводе XTAL1.

Присоединение генератора

Подсоединения генератора



Конфигурация привода внешнего генератора

**Обзор архитектуры**

Концепция файла регистра быстрого допуска содержит 32 x 8-бит рабочие регистры общего назначения со временем допуска за одиночный цикл генератора. Это означает, что за время одиночного цикла генератора выполняется одна операция арифметического логического блока. Два операнда являются выходными от регистрового файла, операция выполняется и результат сохраняется обратно в регистровом файле – за один цикл генератора.

Шесть из 32 регистров могут использоваться как три 16-битных указателя регистров непрямого адреса для адресации пространства данных – задействуя вычисления эффективных адресов. Один из трех указателей адресов также используется как указатель адреса для функции нахождения константной таблицы. Этими регистрами добавочной функции являются 16-битные X-регистр, Y-регистр и Z-регистр. Арифметический логический блок поддерживает арифметические и

логические функции между регистрами или между константой и регистром. Одиночные регистровые операции также выполняются в арифметическом логическом блоке.

В дополнение к регистровой операции могут использоваться режимы адресации обычной памяти также на регистровом файле. Это задействуется тем, что регистровому файлу могут быть назначены 32 самых нижних адреса пространства данных (\$00 - \$1F), что обеспечивает к ним доступ, хотя они являются обычными ячейками памяти.

Пространство ввода/вывода содержит 64 адресов для периферийных функций центрального процессорного блока, таких как контрольные регистры, таймеры / счетчики, аналого-цифровые преобразователи и другие функции ввода / вывода. Доступ к памяти ввода/вывода может быть получен напрямую или как месторасположения пространства данных, следующие за пространствами регистрового файла, \$20 - \$5F.

AVR имеет архитектуру Гарварда – с отдельными памятью и шиной для программы и данных. Доступ к памяти программы достигается последовательно в два этапа. Тогда как выполняется одна команда, следующая программа предварительно доставляется из памяти программы. Эта концепция задействует команды для выполнения в каждом цикле генератора. Память программы является внутрисистемной программируемой флэш-памятью.

При командах относительного перехода и вызова обеспечивается прямой доступ к целому адресному пространству в 1К. Большинство команд AVR имеют одиночный формат слова в 16-бит. Каждый адрес программной памяти содержит 16- или 32-битную команду.

Во время прерываний и вызовов стандартной подпрограммы, на стеке сохраняется программный счетчик обратного адреса. Стек эффективно располагается в SRAM общих данных и, следовательно размер стека только ограничивается полным размером SRAM и использованием SRAM. Все пользовательские программы должны инициализировать стандартную подпрограмму при процедуре сброса (до выполнения стандартных подпрограмм или прерываний). Имеется доступ в режиме чтения/записи 8-битного стекового указателя стандартной подпрограммы в пространстве ввода/вывода.

128 байтный SRAM данных+ регистровый файл и регистры ввода/вывода могут быть легко доступны через пять различных адресных режимов, поддерживаемых архитектурой AVR.

Пространства памяти в архитектуре AVR являются все линейными и обычными картами памяти.

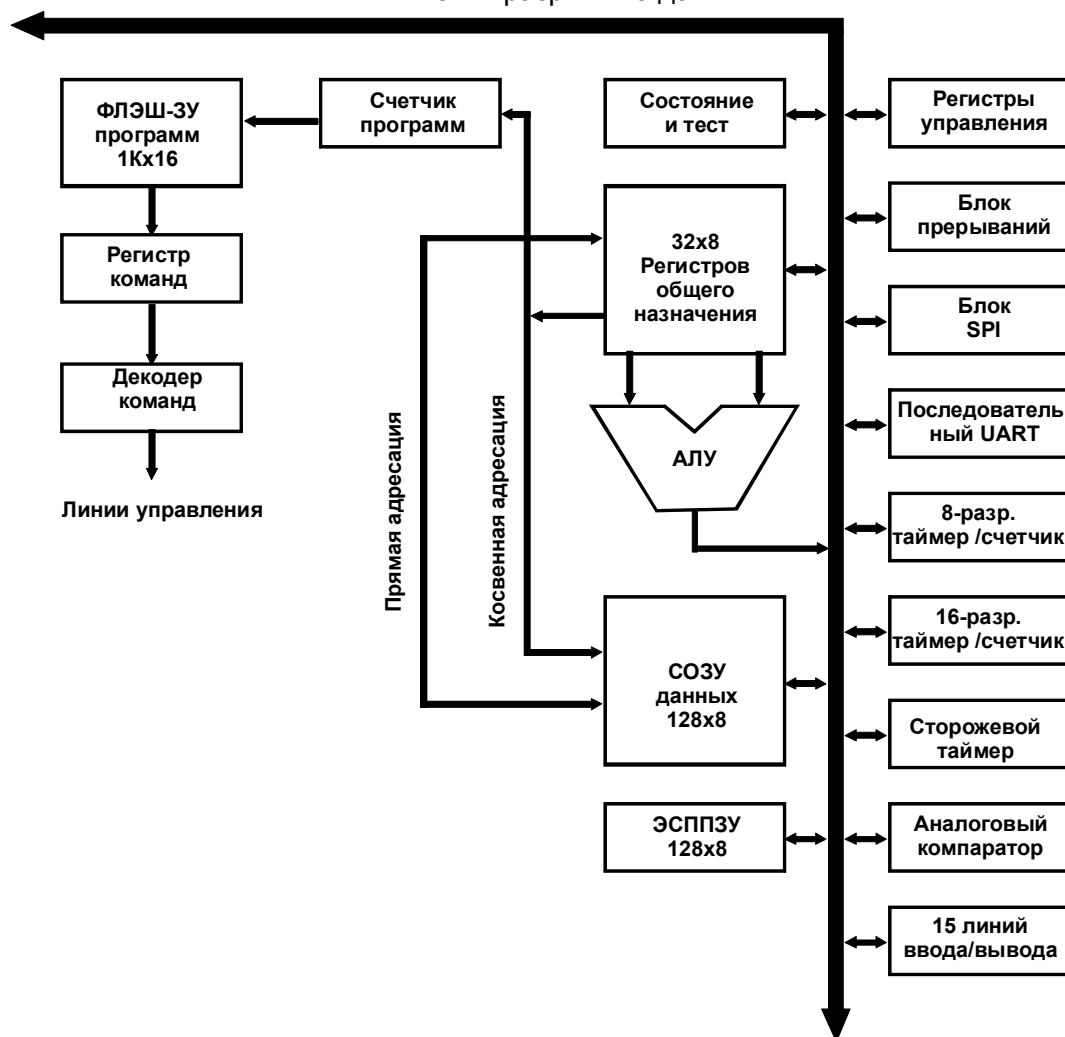


Рисунок 3 - AVR улучшенная RISC архитектура

IN90S2313D

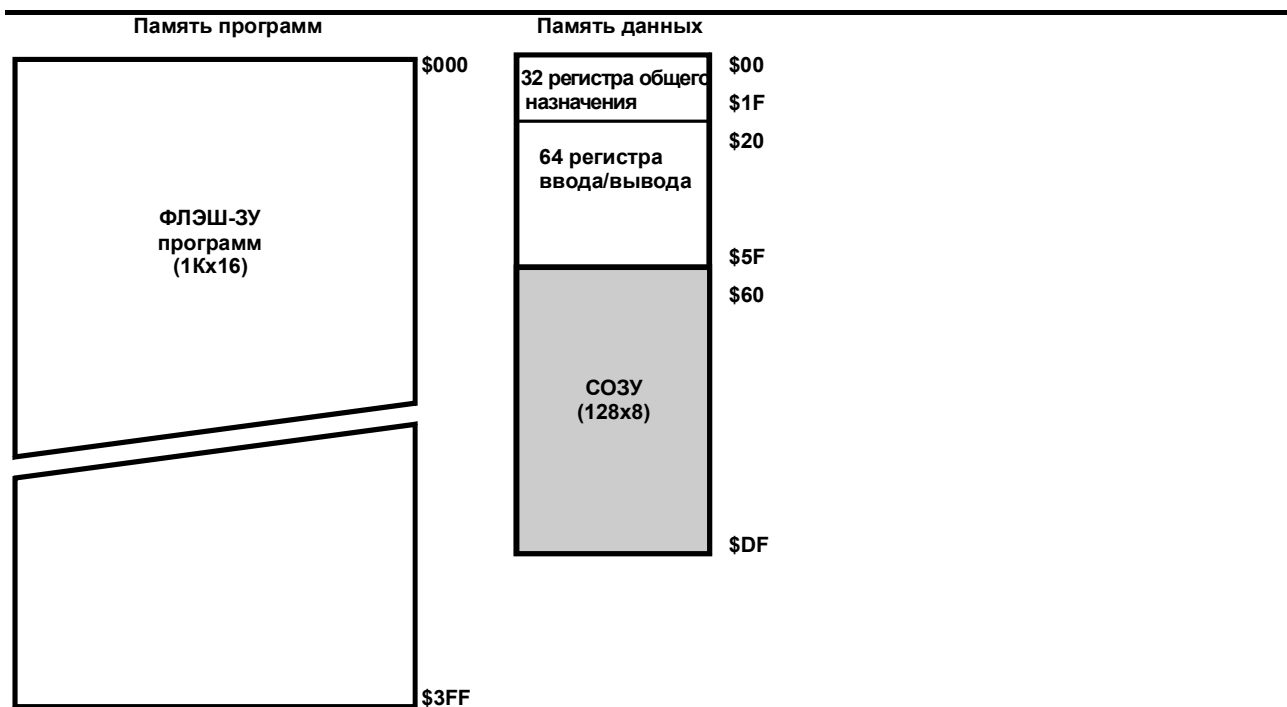


Рисунок 4 - Карта памяти

IN90S2313D

Резюме регистров

Адрес	Название	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	17
\$3E (\$5E)	Зарезервировано									
\$3D (\$5D)	SPL	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0	18
\$3C (\$5C)	Зарезервировано									
\$3B (\$5B)	GIMSK	INT1	INT0	-	-	-	-	-	-	23
\$3A (\$5A)	GIFR	INTF1	INTF0							23
\$39 (\$59)	TIMSK	TOIE1	OCIE1A	-	-	TICIE1	-	TOIE0	-	23
\$38 (\$58)	TIFR	TOV1	OCF1A	-	-	ICF1	-	TOV0	-	24
\$37 (\$57)	Зарезервировано									
\$36 (\$56)	Зарезервировано									
\$35 (\$55)	MCUCR	-	-	SE	SM	ISC11	ISC10	ISC01	ISC00	25
\$34 (\$54)	Зарезервировано									
\$33 (\$53)	TCCR0	-	-	-	-	-	CS02	CS01	CS00	28
\$32 (\$52)	TCNT0	Таймер/Счетчик0 (8 бит)								29
\$31 (\$51)	Зарезервировано									
\$30 (\$50)	Зарезервировано									
\$2F (\$4F)	TCCR1A	COM1A1	COM1A0	-	-	-	-	PWM11	PWM10	30
\$2E (\$4E)	TCCR1B	ICNC1	ICES1	.	-	CTC1	CS12	CS11	CS10	31
\$2D (\$4D)	TCNT1H	Timer/Counter1 – Единичный байт регистра счетчика								32
\$2C (\$4C)	TCNT1L	Таймер/счетчик1 – нулевой байт регистра счетчика								32
\$2B (\$4B)	OCR1AH	Таймер/счетчик1 – Единичный байт регистра сравнения								32
\$2A (\$4A)	OCR1AL	Таймер/счетчик1 – нулевой байт регистра сравнения								32
\$29 (\$49)	Зарезервировано									
\$28 (\$48)	Зарезервировано									
\$27 (\$47)	Зарезервировано									
\$26 (\$46)	Зарезервировано									
\$25 (\$45)	ICR1H	Таймер/счетчик1 – единичный байт регистра захвата входа								33

IN90S2313D

Адрес	Название	Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	Стр.
\$24 (\$44)	ICR1L	Таймер/счетчик1 – нулевой байт регистра захвата входа								33
\$23 (\$43)	Зарезерви ровано									
\$22 (\$42)	Зарезерви ровано									
\$21 (\$41)	WDTCSR	-	-	-	WDTOE	WDE	WDP2	WDP1	WDP0	35
\$20 (\$40)	Зарезерви ровано									
\$1F (\$3F)	Зарезерви ровано									
\$1E (\$3E)	EEAR	-	EEPROM адресный регистр							36
\$1D (\$3D)	EEDR	EEPROM регистр данных								37
\$1C (\$3C)	EECR	-	-	-	-	-	EEMWE	EEWE	EERE	37
\$1B (\$3B)	Зарезерви ровано									
\$1A (\$3A)	Зарезерви ровано									
\$19 (\$39)	Зарезерви ровано									
\$18 (\$38)	PORTB	PORTB7	PORTB6	PORTB5	PORTB4	PORTB3	PORTB2	PORTB1	PORTB0	46
\$17 (\$37)	DDRB	DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0	46
\$16 (\$36)	PINB	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0	46
\$15 (\$35)	Зарезерви ровано									
\$14 (\$34)	Зарезерви ровано									
\$13 (\$33)	Зарезерви ровано									
\$12 (\$32)	PORTD	-	PORTD6	PORTD5	PORTD4	PORTD3	PORTD2	PORTD1	PORTD0	51
Адрес	Назва-ние	Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	Стр.
\$11 (\$31)	DDRD	-	DDD6	DDD5	DDD4	DDD3	DDD2	DDD1	DDD0	51
\$10 (\$30)	PIND	-	PIND6	PIND5	PIND4	PIND3	PIND2	PIND1	PIND0	51
\$0F (\$2F)	Зарезерви ровано									
\$0E (\$2E)	Зарезерви ровано									
\$0D (\$2D)	Зарезерви ровано									
\$0C (\$2C)	UDR	UART I/O Дрегистр данных ввода/вывода								40
\$0B (\$2B)	USR	RXC	TXC	UDRE	FE	OR	-	-	-	40
\$0A (\$2A)	UCR	RXCIE	TXCIE	UDRIE	RXEN	TXEN	CHR9	RXB8	TXB8	41
\$09 (\$29)	UBRR	UART регистр степени Baud Rate								43

Адрес	Название	Бит 7	Бит 6	Бит 5	Бит 4	Бит 3	Бит 2	Бит 1	Бит 0	Стр.
\$08 (\$28)	ACSR	ACD	-	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0	44
...	Зарезервировано									
\$00 (\$20)	Зарезервировано									

Резюме набора команд

Мнемоника	Операнды	Описание	Операция	Флаги	#Синхросигналов
АРИФМЕТИЧЕСКИЕ И ЛОГИЧЕСКИЕ КОМАНДЫ					
ADD	Rd, Rr	Добавить два регистра	$Rd \leftarrow Rd + Rr$	Z,C,N,V,H	1
ADC	Rd, Rr	Добавить с переносом два регистра	$Rd \leftarrow Rd + Rr + C$	Z,C,N,V,H	1
ADIW	Rdl, K	Добавить немедленно к слову	$Rdh:Rdl \leftarrow Rdh:Rdl + K$	Z,C,N,V,S	2
SUB	Rd, Rr	Вычесть два регистра	$Rd \leftarrow Rd - Rr$	Z,C,N,V,H	1
SUBI	Rd, K	Вычесть константу из регистра	$Rd \leftarrow Rd - K$	Z,C,N,V,H	1
SBIW	Rdl, K	Свычесть немедленно из слова	$Rdh:Rdl \leftarrow Rdh:Rdl - K$	Z,C,N,V,S	2
SBC	Rd, Rr	Вычесть с переносом двух регистров	$Rd \leftarrow Rd - Rr - C$	Z,C,N,V,H	1
SBCI	Rd, K	Вычесть с переносом константы из регистра.	$Rd \leftarrow Rd - K - C$	Z,C,N,V,H	1
AND	Rd, Rr	Логические регистры AND	$Rd \leftarrow Rd \bullet Rr$	Z,N,V	1
ANDI	Rd, K	Логический регистр AND и константа	$Rd \leftarrow Rd \bullet K$	Z,N,V	1
OR	Rd, Rr	Логические регистры OR	$Rd \leftarrow Rd \vee Rr$	Z,N,V	1
ORI	Rd, K	Логический регистр OR и константа	$Rd \leftarrow Rd \vee K$	Z,N,V	1
EOR	Rd, Rr	Эксклюзивные регистры OR	$Rd \leftarrow Rd \oplus Rr$	Z,N,V	1
COM	Rd	Одополнение одного	$Rd \leftarrow \$FF - Rd$	Z,C,N,V	1
NEG	Rd	Дополнение двух	$Rd \leftarrow \$00 - Rd$	Z,C,N,V,H	1
SBR	Rd, K	Установить биты в регистре	$Rd \leftarrow Rd \vee K$	Z,N,V	1
CBR	Rd, K	Очистить биты в регистре	$Rd \leftarrow Rd \bullet (\$FF - K)$	Z,N,V	1
INC	Rd	Увеличение	$Rd \leftarrow Rd + 1$	Z,N,V	1
DEC	Rd	Уменьшение	$Rd \leftarrow Rd - 1$	Z,N,V	1
TST	Rd	Тест для нуля или минуса	$Rd \leftarrow Rd \bullet Rd$	Z,N,V	1
CLR	Rd	Очистить регистр	$Rd \leftarrow Rd \oplus Rd$	Z,N,V	1
SER	Rd	Установить регистр	$Rd \leftarrow \$FF$	Нет	1
ВКОМАНДЫ УСЛОВНЫХ ПЕРЕХОДОВ					
RJMP	k	Относительный скачок (переход)	$PC \leftarrow PC + k + 1$	Нет	2
IJMP		Непрямой переход к (Z)	$PC \leftarrow Z$	Нет	2
RCALL	k	Относительный вызов стандартной подпрограммы	$PC \leftarrow PC + k + 1$	Нет	3
ICALL		Непрямой вызов к (Z)	$PC \leftarrow Z$	Нет	3
RET		Возврат стандартной подпрограммы	$PC \leftarrow STACK$	Нет	4
RETI		Возврат прерывания	$PC \leftarrow STACK$	I	4

Мнемо-ника	Операнды	Описание	Операция	Флаги	#Синхро-сигналов
CPSE	Rd,Rr	Сравнить, пропустить, если равен	if (Rd = Rr) $PC \leftarrow PC + 2 \text{ or } 3$	Нет	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 регистр с немедленным	$Rd - K$	Z, N,V,C,H	1
SBRC	Rr, b	Пропустить, если бит очищен в регистре	if (Rr(b)=0) $PC \leftarrow PC + 2 \text{ or } 3$	Нет	1 / 2
SBRB	Rr, b	Пропустить, если бит установлен в регистре	if (Rr(b)=1) $PC \leftarrow PC + 2 \text{ or } 3$	Нет	1 / 2
SBIC	P, b	Пропустить, если бит очищен в регистре ввода/вывода	if (P(b)=0) $PC \leftarrow PC + 2 \text{ or } 3$	Нет	1 / 2
SBIS	P, b	Пропустить, если бит установлен в регистре ввода/вывода	if (P(b)=1) $PC \leftarrow PC + 2 \text{ or } 3$	Нет	1 / 2
BRBS	s, k	Перейти, если установлен флаг состояния	if (SREG(s) = 1) затем $PC \leftarrow PC + k + 1$	Нет	1 / 2
BRBC	s, k	Перейти, если очищен флаг состояния	if (SREG(s) = 0) затем $PC \leftarrow PC + k + 1$	Нет	1 / 2
BREQ	k	Перейти, если равен	if (Z = 1) затем $PC \leftarrow PC + k + 1$	Нет	1 / 2
BRNE	k	Перейти, если не равен	if (Z = 0) затем $PC \leftarrow PC + k + 1$	Нет	1 / 2
BRCS	k	Перейти, если установлен перенос	if (C = 1) затем $PC \leftarrow PC + k + 1$	Нет	1 / 2
BRCC	k	Перейти, если очищен перенос	if (C = 0) затем $PC \leftarrow PC + k + 1$	Нет	1 / 2
BRSH	k	Перейти, если такой же или выше	if (C = 0) затем $PC \leftarrow PC + k + 1$	Нет	1 / 2
BRLO	k	Перейти, если ниже	if (C = 1) затем $PC \leftarrow PC + k + 1$	Нет	1 / 2
BRMI	k	Перейти, если минус	if (N = 1) затем $PC \leftarrow PC + k + 1$	Нет	1 / 2
BRPL	k	Перейти, если плюс	if (N = 0) затем $PC \leftarrow PC + k + 1$	Нет	1 / 2
BRGE	k	Перейти, если больше или равен, со знаком	if ($N \oplus V = 0$) затем $PC \leftarrow PC + k + 1$	Нет	1 / 2
BRLT	k	Перейти, если меньше, чем ноль, со знаком	if ($N \oplus V = 1$) затем $PC \leftarrow PC + k + 1$	Нет	1 / 2
BRHS	k	Перейти, если установлен флаг полупереноса	if (H = 1) затем $PC \leftarrow PC + k + 1$	Нет	1 / 2
BRHC	k	Перейти, если очищен флаг полупереноса	if (H = 0) затем $PC \leftarrow PC + k + 1$	Нет	1 / 2
BRTS	k	Перейти, если установлен флаг T	if (T = 1) затем $PC \leftarrow PC + k + 1$	Нет	1 / 2
BRTC	k	Перейти, если очищен флаг T	if (T = 0) затем $PC \leftarrow PC + k + 1$	Нет	1 / 2
BRVS	k	Вперейти, если установлен флаг переполнения	if (V = 1) затем $PC \leftarrow PC + k + 1$	Нет	1 / 2

Мнемоника	Операнды	Описание	Операция	Флаги	#Синхросигналов
BRVC	k	Перейти, если очищен флаг переполнения	if (V = 0) затем $PC \leftarrow PC + k + 1$	Нет	1 / 2
BRIE	k	Перейти, если задействовано прерывание	if (I = 1) затем $PC \leftarrow PC + k + 1$	Нет	1 / 2
BRID	k	Перейти, если отключено прерывание	if (I = 0) затем $PC \leftarrow PC + k + 1$	Нет	1 / 2
КОМАНДЫ ПЕРЕДАЧИ ДАННЫХ					
MOV	Rd, Rr	Сдвинуть между регистрами	$Rd \leftarrow Rr$	Нет	1
LDI	Rd, K	Загрузить немедленно	$Rd \leftarrow K$	Нет	1
LD	Rd, X	Непрямая загрузка	$Rd \leftarrow (X)$	Нет	2
LD	Rd, X+	Непрямая загрузка и Post-Inc.	$Rd \leftarrow (X), X \leftarrow X + 1$	Нет	2
LD	Rd, -X	Непрямая загрузка и Pre-Dec.	$X \leftarrow X - 1, Rd \leftarrow (X)$	Нет	2
LD	Rd, Y	Непрямая загрузка	$Rd \leftarrow (Y)$	Нет	2
LD	Rd, Y+	Непрямая загрузка и Post-Inc.	$Rd \leftarrow (Y), Y \leftarrow Y + 1$	Нет	2
LD	Rd, -Y	непрямая загрузка и Pre-Dec.	$Y \leftarrow Y - 1, Rd \leftarrow (Y)$	Нет	2
LDD	Rd, Y+q	Непрямая загрузка и перемещение	$Rd \leftarrow (Y + q)$	Нет	2
LD	Rd, Z	Непрямая загрузка	$Rd \leftarrow (Z)$	Нет	2
LD	Rd, Z+	Непрямая загрузка и Post-Inc.	$Rd \leftarrow (Z), Z \leftarrow Z + 1$	Нет	2
LD	Rd, -Z	Непрямая загрузка и Pre-Dec.	$Z \leftarrow Z - 1, Rd \leftarrow (Z)$	Нет	2
LDD	Rd, Z+q	Непрямая загрузка и перемещение	$Rd \leftarrow (Z + q)$	Нет	2
LDS	Rd, k	Прямая загрузка от SRAM	$Rd \leftarrow (k)$	Нет	2
ST	X, Rr	Непрямое сохранение	$(X) \leftarrow Rr$	Нет	2
ST	X+, Rr	Непрямое сохранение and Post-Inc.	$(X) \leftarrow Rr, X \leftarrow X + 1$	Нет	2
ST	-X, Rr	Непрямое сохранение и Pre-Dec.	$X \leftarrow X - 1, (X) \leftarrow Rr$	Нет	2
ST	Y, Rr	Непрямое сохранение	$(Y) \leftarrow Rr$	Нет	2
ST	Y+, Rr	Непрямое сохранение и Post-Inc.	$(Y) \leftarrow Rr, Y \leftarrow Y + 1$	Нет	2
ST	-Y, Rr	Непрямое сохранение и Pre-Dec.	$Y \leftarrow Y - 1, (Y) \leftarrow Rr$	Нет	2
STD	Y+q, Rr	Непрямое сохранение с перемещением	$(Y + q) \leftarrow Rr$	Нет	2
ST	Z, Rr	Непрямое сохранение	$(Z) \leftarrow Rr$	Нет	2
ST	Z+, Rr	Непрямое сохранение и Post-Inc.	$(Z) \leftarrow Rr, Z \leftarrow Z + 1$	Нет	2
ST	-Z, Rr	Непрямое сохранение и Pre-Dec.	$Z \leftarrow Z - 1, (Z) \leftarrow Rr$	Нет	2
STD	Z+q, Rr	Непрямое сохранение с перемещением	$(Z + q) \leftarrow Rr$	Нет	2
STS	k, Rr	Прямое сохранение в SRAM	$(k) \leftarrow Rr$	Нет	2

Мнемоника	Операнды	Описание	Операция	Флаги	#Синхросигналов
LPM		Загрузить программную память	$R0 \leftarrow (Z)$	Нет	3
IN	Rd, P	В порту	$Rd \leftarrow P$	Нет	1
OUT	P, Rr	Вне порта	$P \leftarrow Rr$	Нет	1
PUSH	Rr	Нажимной регистр на стеке	$STACK \leftarrow Rr$	Нет	2
POP	Rd	Всплывающий регистр из стека	$Rd \leftarrow STACK$	Нет	2
ПОБИТОВЫЕ ТЕСТОВЫЕ КОМАНДЫ					
SBI	P, b	Установить бит в регистре ввода/вывода	$I/O(P, b) \leftarrow 1$	Нет	2
CBI	P, b	Очистить бит в регистре ввода/вывода	$I/O(P, b) \leftarrow 0$	Нет	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	Вращать влево через перенос	$Rd(0) \leftarrow C,$ $Rd(n+1) \leftarrow Rd(n),$ $C \leftarrow Rd(7)$	Z, C, N, V	1
ROR	Rd	Вращать вправо через перенос	$Rd(7) \leftarrow C, Rd(n) \leftarrow$ $Rd(n+1), C \leftarrow Rd(0)$	Z, C, N, V	1
ASR	Rd	Арифметическое смещение вправо	$Rd(n) \leftarrow Rd(n+1),$ $n=0..6$	Z, C, N, V	1
SWAP	Rd	Сменить полубайты	$Rd(3..0) \leftarrow Rd(7..4),$ $Rd(7..4) \leftarrow Rd(3..0)$	Нет	1
BSET	s	Флаг установлен	$SREG(s) \leftarrow 1$	SREG(s)	1
BCLR	s	Флаг очищен	$SREG(s) \leftarrow 0$	SREG(s)	1
BST	Rr, b	Сохранить бит из регистра в T	$T \leftarrow Rr(b)$	T	1
BLD	Rd, b	Загрузка бита из T в регистр	$Rd(b) \leftarrow T$	Нет	1
SEC		Установить перенос	$C \leftarrow 1$	C	1
CLC		Очистить перенос	$C \leftarrow 0$	C	1
SEN		Установить отрицательный флаг	$N \leftarrow 1$	N	1
CLN		Очистить отрицательный флаг	$N \leftarrow 0$	N	1
SEZ		Установить нулевой флаг	$Z \leftarrow 1$	Z	1
CLZ		Очистить нулевой флаг	$Z \leftarrow 0$	Z	1
SEI		Задействовать общее (глобальное) прерывание	$I \leftarrow 1$	I	1
CLI		Отключить общее (глобальное) прерывание	$I \leftarrow 0$	I	1
SES		Установить тестовый флаг со знаком	$S \leftarrow 1$	S	1
CLS		Очистить тестовый флаг со знаком	$S \leftarrow 0$	S	1
SEV		Установить переполнение дополнения двух	$V \leftarrow 1$	V	1
CLV		Очистить переполнение дополнения двух	$V \leftarrow 0$	V	1

IN90S2313D

Мнемо-ника	Операнды	Описание	Операция	Флаги	#Синхро-сигналов
SET		Установить T в SREG	$T \leftarrow 1$	T	1
CLT		Очистить T в SREG	$T \leftarrow 0$	T	1
SEH		Установить флаг полупереноса в SREG	$H \leftarrow 1$	H	1
CLH		Очистить флаг полупереноса в SREG	$H \leftarrow 0$	H	1
NOP		Нет операции		Нет	1
SLEEP		Режим ожидания	(см. конкретное описание для функции ожидания)	Нет	3
WDR		Сброс сторожевого таймера	(см. конкретное описание для WDR / таймера)	Нет	1