

3-5 خصوصیات Atmega8L, Atmega8

* از معماری AVR RISC استفاده می کند.

- کارایی بالا توان مصرفی کم

- دارای 130 دستورالعمل با کارایی بالا که اکثر در یک کلاک سیکل اجرا می شوند.

8*32 رجیستر کاربردی

- سرعتی تا 16MIPS در فرکانس 16MHZ

حافظه < برنامه و داده غیر فرار

8K بایت حافظه FLASH داخلی قابل برنامه ریزی

پایداری حافظه FLASH و قابلیت 10000 بار نوشتن و پاک کردن (WRITE/ERASE)

1024 بایت حافظه EEPROM داخلی قابل برنامه ریزی

پایایی حافظه EEPROM قابلیت 100,000 بار نوشتن و پاک کردن (WRITE/ERASE)

فعل برنامه FLASH حفاظت داده EEPROM

* خصوصیات جانبی

دو تایمر - کانتر (TIMER/COUNTER) 8 بیتی با PRESCALER مجزا و دارای نذ

CAPUTER و COMARER

- 3 کانال PWM

8 - کانال مد آنالوگ به دیجیتال در بسته بندی های TQFP و MLF

6 کانال با دقت 10 بیتی

2 کانال با دقت 8 بیتی

6 - کانال مبدل آنالوگ به دیجیتال در بسته بندیهای PDIP

4 کانال با دقت 10 بیتی

2 کانال با دقت 8 بیتی

- دارای (REAL-TIME CLOCK) یا اسیلاتور مجزا

- یک مقایسه کننده آنالوگ داخلی

- USART سریال قابل برنامه ریزی

- WATCHDOG قابل برنامه ریزی با اسیلاتور داخلی

- ارتباط سریال SPI برای برنامه ریزی داخل مدار (IN SYSTEM PROGRAMING)

- قابلیت ارتباط سریال (SERIAL PERIPHERAL INTERFACE) SPI به صورت

MASTER یا SLAVE

- قابلیت ارتباط با پروتکل سریال دو سیمه (TOW-WIRE)

* خصوصیات ویژه میکروکنترلر

- POWER- ON RESET CIRCUIT

- دارای 5 حالت - IDLE ADC NOISE REDUCCTION, POWER- SAVE, POWER-

(STANDBY و DOWN)

- منابع وقفه (INTERPUT) داخلی و خارجی

- دارای اسپلاتور RC داخلی کالبره شده

- عملکرد کاملاً ثابت

- توان مصرفی پائین و سرعت بالا توسط تکنولوژی CMOS

* توان مصرفی در 3V, 4MHZ و 25°C

- حالت فعال (ACTIVE MODE) 3.6 Ma

- در حالت بیکاری (IDELMODE) 1.0, Ma

- در حالت POWER- DOWN : $5\mu A >$

* ولتاژ های عملیاتی (کاری)

- 2.7V تا 5.5 برای (Atmega 8L)

- 4.5V تا 5.5V برای (Atmega8)

* فرکانسهای کاربری

- 0MHZ تا 8MHZ برای (Atmega 8L)

- 0MHZ تا 16MHZ برای (8 Atmega)

* خطوط 1/0 و انواع بسته بندی

- 23 خطوط ورودی / خروجی (I/O) قابل برنامه ریزی

- 28 پایه PDIP و 32 پایه TAFP و MLF

* ترکیب پایه ها

فیوز بیت های ATMEFGA8

ATMEFGA8 برای دو بایت فیوز بیت است که در دو جدول نشان داده شده اند. منطق 0 به معنای

برنامه ریزی شدن و I به معنای برنامه ریزی نشدن بیت است.

RSTDISBL: در حالت پیش فرض PC6 پایه ریس ست است. با برنامه ریزی این بیت پایه PC6 به

عنوان پایه I/O استفاده می شود.

WDTON: در حالت پیش فرض WATCHDOG غیرفعال و کاربرد بایستی نرم افزاری

WATCHDOG را راه اندازی کند ولی زمان که این بیت برنامه ریزی شود WATCHDOG همیشه

روشن است.

SPINE: در حالت پیش فرض برنامه ریزی شده و میکرو از طریق سریال SPI برنامه ریزی می شود.

این بیت در مد برنامه ریزی سریال قابل دسترسی نمی باشد.

CKOPT: بیت انتخابی کلاک که به صورت پیش فرض برنامه ریزی نشده است. عملکرد این بیت

بستگی به بیت های CKSEL دارد که در بخش 3-14 در انتهای همین فصل آمده است.

EESAVE: در حالت پیش فرض برنامه ریزی نشده و در زمان پاک شدن EEPROM در زمان پاک شدن میکرو، محفوظ می ماند.

BOOTSZ0,BOOTSZ1 برای انتخاب مقدار حافظه BOOT طبق جدول زیر برنامه ریزی می شوند و در زمان برنامه ریزی شدن فیوز بیت BOOTRST اجرای برنامه از آدرس حافظه BOOT آغاز خواهد شد.

BOOTRST: پیش انتخاب برای ری ست BOOT که در حالت پیش فرض برنامه ریزی نشده و آدرس بردار ری ست \$0000 است و در صورت برنامه یزی آدرس بردار ری ست به آدرسی که فیوز بیت های BOOTSZ0 و BOOTSZ1 مشخص کرده اند تغییر می یابد.

BODLEVEL: زمانی که این بیت برنامه ریزی نشده (پیش فرض) باشد اگر ولتاژ پایه VCC از 2.7V پائین تر شود ری ست داخلی میکرو فعال شده و سیستم را ری ست می کند. زمانی که این بیت برنامه ریزی شده باشد اگر ولتاژ پایه VCC از 4V پائین تر شود ری ست داخلی میکرو فعال شده و میکرو را طبق شکل 3-1 ری ست می کند.

BODEN: برای فعال کردن عملکرد مدار BROWN-OUT این بیت بایستی برنامه ریزی شده باشد این بیت به صورت پیش فرض برنامه ریزی نشده است.

SUT1,SUT0 برای انتخاب زمان START-UP بکار برده می شوند.

14-3 کلاک سیستم (1)

توزیع کلاک

کلاک سیستم میکرو طبق شکل 3-3 توزیع شده است.

کلاک CPU – CLK cpu

این کلاک برای انجام عملیات AVR بطور مثال رجیستر ها استفاده می شود توقف وبه مکت بردن این

کلاک باعث می شود که عملیات و محاسبات VAR انجام نگیرد.

کلاک I/O-I/O-CLK

این کلاک توسط بسیاری از ماژول های I/O بطور مثال تایمرها، کانترها، SPI و uart استفاده می گردد.

کلاک FLASH-FLASH-CLK

این کلاک عملیات ارتباطی با حافظه FLASH را کمتر می کند. کلاک FLASH معمولاً با کلاک CPU

فعال می شود.

شکل 3-3 نمودار توزیع کلاک سیستم میکرو

کلاک غیرهمزمان تایمر – CLK ASY

با این کلاک تایمر / کانتر به صورت غیرهمزمان توسط کریستال ساعت 33768 HZ کار می کند. حتی

اگر سیستم در حالت SLEPP باشد.

کلاک ADC-CLKADC

ADC از یک کلاک جداگانه حساس استفاده می کند که باعث می شود کلاک های CPU و I/O به حالت ایست (HALT) رفته تا نویز حاصل از مدار دیجیتال داخلی کاهش یافته و در نتیجه عملیات تبدیل به دقت بیشتری انجام یابد.

منابع کلاک (CLOCKSOURCE)

میکرو دارای انواع منابع کلاک اختیاری است که می توان انواع آن را بوسیله بیت های قابل برنامه ریزی FLASH (FLASH FUSE BITE) انتخاب کرد. کلاک انتخاب شده بعنوان ورودی کلاک AVR طبق جدول زیر در نظر گرفته شده و کلاک مناسب به هر قسمت سیستم داده می شود.

نکته: در تمام جداول فیوز بیت ها، 0 به معنای بیت برنامه ریزی شده (PROGRAMMED) و 1 به معنای بیت برنامه ریزی نشده (UNPROGRAMMED) است.

CE CLOCKING OPTION	L3...0
INTERNAL CRYSTAL/CLOCK GENERATOR	1010
INTERNAL - LOW FREQUENCY CRYSTAL	
INTERNAL RC OSCILLATOR	0101
PROGRAMMED INTERNAL CRYSTAL OSCILLATOR	0001
INTERNAL CLOCK	

جدول انتخاب انواع کلاک سیستم میکرو

هنگامی که CPU از مد POWER-DOWN یا POWER-SAVE خارج می شود زمانی به نام زمان شروع (START-UP) برای رسیدن به کریستال به شرایط پایدار ایجاد و سپس دستورات برنامه اجرا می شود و هنگامی که CPU از ری ست شروع به کار می کند، تأخیری اضافه (DELAY) برای رسیدن ولتاژ به سطح پایدار ایجاد شده و سپس اجرای برنامه آغاز می شود. برای ایجاد زمانبندی های مذکور از اسیلاتور WATCHDOG استفاده می گردد.

اسیلاتور کریستالی (EXTERNAL CRYSTAL/CERAMIC RESONATOR)

در این حالت کریستالی یا نوسانگر سرامیکی (CERAMIC RESONATOR) با کریستالی کوارتز (QUARTZ CRYSTAL) همانطور که در شکل زیر نشان داده شده است به دو پایه XTAL1 و XTAL2 وصل شود.

فیوز بیت CKOPT می تواند برای دو حالت مختلف استفاده شود زمانی که محیط بسیار نویزی باشد، این بیت برنامه ریزی می شود که رنج وسیعی از فرکانسها را شامل می شود. برنامه ریزی نکردن CKOPT باعث کاهش مصرف شده و برخلاف قبل رنج محدودی از فرکانسها را شامل می شود. خازنهای C1 و C2 برای کریستالها و نوسانگرها بایستی یک مقدار باشند و مقادیر آنها بستگی به کریستال، نوسانگر و نویزهای الکترومغناطیسی محیط دارد. بعضی از خازنهای مورد استفاده برای کریستال های مختلف در جدول زیر آمده است. برای نوسانگرهای سرامیکی بایستی مقدار خازنهایی که توسط کارخانه پیشنهاد می گردد استفاده می شود.

شکل اتصال کریستال به میکرو در حالت اسیلاتور کریستالی

CKOPT	CKSEL 3..1	FREQUENCY RECOMMENDED RANGE(MHZ)	RANG FOR CAPITORS C1 AND C2 FOR WITH CRYSTAL(PF)
1	101 ¹	0.4-0.9	
1	110	0.9-3.0	12-22
1	111	3.0-8.0	12-22
0	101,110,111	$1.0 \leq$	12-22

جدول مدهای عملیاتی اسیلاتور کریستالی

توسط فیوز بیت CKSEL0 و SUT1..0 زمان آغاز (START- UP) را میتوان طبق جدول زیر انتخاب کرد.

اسیلاتور کریستالی فرکانس پائین (EXTERNAL- LOW FREQUENCY CRYSTAL)

برای استفاده از کریستالی ساعت 32.768KHZ فیوز بیت های CKSEL با 1001 برنامه ریزی می شوند و کریستال طبق شکل صفحه قبل به پایه های XTAL1 و XTAL2 متصل می شوند. با برنامه ریزی

¹ - این انتخاب برای نوسانگر سرامیکی استفاده می شود و نباید آنرا برای کریستال بکار برد.

کردن CKOPT می توان خازنهای داخلی را فعال نمود و در نتیجه خازنهای خارجی را برداشت. مقدار نامی خازنهای داخلی 36PF است.

هنگامی که این نوع کریستال انتخاب می شود زمان شروع (START-UP) توسط فیوز بیتهای SUT طبق جدول زیر قابل انتخاب است.

..0	T - UP TIME	TIONAL	RECOMMENDED
	ER- DOWN	RESET	GE
	ER- SAVE	=5.0V)	
			RISING POWER
			OD ENBAL
			WLY RISING
			ER
	CK		BLE FREQUENCY
			T - UP
RESERVED			

جدول انتخاب زمان START - UP برای کلاک اسپلاتور کریستالی فرکانس پائین

اسیلاتور RC خارجی (EXTERNAL RC OSCILLATOR)

اتصال RC به پایه های XTAL1 در شکل صفحه بعد آمده است. فرکانس تقریبی توسط معادله

$f = 1/(3RC)$ بدست می آید. مقدار خازن بایستی حداقل 22PF باشد. با برنامه ریزی کردن فیوز بیت

CKOPT کاربر می تواند خازنهای داخلی 36PF را بین XTAL1 و GND راه اندازی کند و در نتیجه

دیگر نیازی به خازن خارجی نیست.

شکل اتصال RC به میکرو در حالت اسیلاتور RC خارجی

اسیلاتور می تواند در 4 مد فرکانسی کار کند که این فرکانس طبق فیورز بیت های 3..0 CKSEL طبق

جدول زیر قابل انتخاب است.

CKSEL3..0	FREQUENCY RANG(MHZ)
0101	≤ 0.9
0110	0.9-0.3
0111	3.0-8.0
1000	8.0-12.0

جدول مدهای عملیاتی اسیلاتور RC خارجی

هنگامی که فرکانس کاری انتخاب می شود، زمان شروع (START-UP) توسط فیوز بیت های DUT طبق جدول زیر قابل انتخاب است.

SUT1.0	START-UP TIME	OPTIONAL DELAY	RECOMMENDED USAGE
00	18CK	4.1ms	BOD ENABLE
01	18CK	4.1ms	RISING POWER
10	18CK	65ms	SLOWLY RISING POWER
11	6CK ²	4.1ms	FAST RISING POWER BOD ENABLE

جدول انتخاب زمان START UP برای کلاک اسیلاتور RC خارجی

اسیلاتور RC کالیبره شده داخلی (CALIBRATED TED INTERNAL RC SCILLATOR)

اسیلاتور RC کالیبره شده داخلی، کلاک های نامی داخلی 1,2,4 و 8MHZ را در ولتاژ 5V و 25°C تولید می کند. این کلاک با برنامه ریزی کردن بیت های CKSEL می تواند به عنوان کلاک سیستم استفاده گردد که در این حالت نیازی به مدار خارجی نیست. زمانی که از این مدار استفاده می گردد فیوز بیت CKOPT همیشه بایستی برنامه ریزی شده باشد.

CKSEL	NOMONAL RANG(MHZ)
0001 ³	1.0
0010	2.0
0011	4.0
0100	8.0

جدول مدهای عملیاتی اسیلاتور RC کالیبره شده داخلی

هنگامی که در فرکانس کاری انتخاب می شود زمان شروع (START-UP) توسط فیوز بیت های SUT طبق جدول زیر قابل انتخاب است.

(3) برای میکرو به صورت پیش فرض این گزینه انتخاب شده است.

..0	- UP TIME FRCIONAL DELARMMENDED U		
	ER – DOWN AM RESET		
	ER – SAVE	=5.0V)	
			ENABLE
			RISING POWER
			VLY RISING PO
			RESERVED

جدول انتخاب زمان START- UP برای کلاک اسیلاتور RC کالیبره شده داخلی

کلاک خارجی (EXTERNAL CLOCK)

برای اندازه گیری میکرو توسط کلاک خارجی پایه XTAL1 طبق شکل زیر بایستی وصل شود برای کار

در این مد بیت های CKSEL با 0000 برنامه ریزی می شوند با برنامه ریزی کردن فیوز بیت

CKOPT خازن داخلی 36PF بین پایه های XTAL1 و GND می شود.

شکل اتصال کلاک خارجی به پایه میکرو در حالت کلاک خارجی

(4) برای میکرو به صورت پیش فرض این گزینه انتخاب شده است.

هنگامی که این نوع کلاک انتخاب می شود، زمان شروع (START-UP) توسط فیوز بیت های SUT طبق جدول زیر قابل انتخاب است.

..0	START-UP TIME	OPTIONAL DELAY	RECOMMENDED
	RESERVED - DOWN	RESET	RESERVED
	RESERVED - SAVE	(=5.0V)	
			ENABLE
			RISING POWER
			ONLY RISING POWER
			RESERVED

جدول انتخاب زمان START-UP برای کلاک خارجی

در این مد باید از تغییرات ناگهانی فرکانس کلاک خارجی برای اطمینان از انجام پایدار و صحیح عملیات میکروکنترلر (MCU) جلوگیری کرد. تغییرات بیشتر از 2% در فرکانس خارجی منعکس است باعث رفتارهای غیرقابل انتظار میکرو شود. زمانی که قصد تغییر فرکانس کلاک را دارید بایستی میکرو در حالت RESET نگهداشته شود.

اسیلاتور تایمر/کانتر

برای میکروکنترلرهایی که دارای پایه TOSC1 و TOCS2 هستند کریستال ساعت 32.768KHZ مستقیماً بین دو پایه قرار می گیرد و تایمر / کانتر 0 یا 2 به صورت آسنکرون از این دو پایه کلاک دریافت می کند.

پیکره بندی و کار با امکانات

VAR در BASCOM

امکانات تمام میکروهای AVR قبل از استفاده بایستی ابتدا پیکره بندی (CONFIG) شوند. در این فصل به پیکره بندی یا همان راه اندازی امکانات VAR پرداخته ایم. در ابتدا هر بخش رجیسترهای وسیله شرح داده شده اند و سپس نحوه پیکره بندی و دستورات مربوطه به وسیله کاملاً تشریح شده است و از آنجا که ارائه مثال در یادگیری بسیار مفید است در انتها نیز مثالهایی برای درک بیشتر با کاربرد و نحوه کار با وسیله آمده است. در این بخش سعی شده است که دستورات مربوط به امکانات AVR در کنار پیکره بندی قرار گیرد تا کاربر بتواند با مطالعه بخش مربوط بوسیله دلخواه بتواند براحتی وسیله را راه اندازی و با آن کار کند.

اهداف

1. آشنایی با تمام امکانات و رجیسترهای مربوطه میکروهای AVR

2. یادگیری کامل تمام پیکره بندی های امکانات VAR و دستورات

3. راه اندازی وسیله دلخواه و تحلیل نرم افزاری توسط تحلیل گر داخلی BASCOM

4. راه اندازی وسیله دلخواه و تحلیل نرم افزاری توسط تحلیل گر PROTEUS

1-6 پیکره بندی پورت ها

برای تعیین جهت پایه پورتها از این پیکربندی استفاده می نمائیم. جهت یک پایه می تواند ورودی یا خروجی باشد.

Config Portx=State

Config Pinx.y=State

X و Y بسته به میکرو می تواند به ترتیب پایه های 0 تا 7 پورت های A.B.C.D.E.F باشند. State نیز می تواند یکی از گزینه های زیر باشد.

INPUT یا 0: در این حالت رجیستر جهت داده پایه یا پورت انتخاب شده صفر (0) می شود و پایه یا

پورت بعنوان ورودی استفاده می شود.

OUTPUT یا 1: در این حالت رجیستر جهت داده پایه یا پورت انتخاب شده یک (1) می شود و پایه یا

پورت بعنوان خروجی استفاده می شود.

زمانی که بخواهید از پورتی بخوانید بایستی از رجیستر PIN پورت مربوطه استفاده کنید و هنگام نوشتن

در پورت بایستی در رجیستر PORT بنویسید.

* مثال:

Dim A As Byte , Count As Byte

'configure PORT D for input mode

Config Portd = Input

A=Pind 'Read Data on Ported

A= A And Ported 'A=A&PORTED

Print A print it

Bitwait pind .7. Reset wait until bit is low

Config portb= output

'assign value

Portb=10 ' set port B to 10

Portb = portb And 2

Set portb 0 ' set bit 0 of port B to 1

Incr portb

Count =0

Do

Incr count

Prtb=1

For A= 1 To 8

Rotate Portb , Left ' rotate bits left

Wait 1

Next

Loop Until count =10

End

بررسی پورتهای میکرو ATMEGA32

در این بخش سعی داریم برای آشنایی بیشتر با عملکرد پورت ها و رجیسترهای مربوط بطور نمونه به

بررسی پورتهای میکرو ATMEGA32 پردازیم:

پورت A

پورت 1 یک I/O دو طرفه 8 بیتی است. سه آدرس از مکان حافظه I/O اختصاص به PORTA دارد

یک آدرس برای رجیستر داده PORTA، دومی رجیستر جهت داده DDRA و سومی پایه ورودی

پورت PINA.A آدرس پایه های ورودی پورت A فقط قابل خواندن است. در صورتی که رجیستر داده

رجیستر جهت داده هم خواندی و هم نوشتنی هستند تمام پایه های پورت دارای مقاومت Pull-Up مجزا

هستند. بافر خروجی پورت A می تواند تا 20mA را Sink کند و در نتیجه LED را مستقیماً راه اندازی کند. هنگامی که پایه های PA0-PA7 با مقاومت های Pull-Down خارجی، خروجی استفاده می شوند، آنها SOURCE جریان می شوند زمانی که مقاومت های Pull-Up داخلی فعال باشند.

رجیسترهای پورت A

رجیسترهای پورت A عبارتند از:

رجیستر داده پورت A – A [PORT A DATA REGISTER]

Bit 7 6 5 4 3 2 1

0

A7	A6	A5	A4	A3	A2	A1	A0
----	----	----	----	----	----	----	----

Read/ Writ 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

0

رجیستر جهت داده پورت A – A [PORT A DATA DIRECTION REGISTER] DDRA

Bit 7 6 5 4 3 2 1

0

A7	A6	A5	A4	A3	A2	A1	A0
----	----	----	----	----	----	----	----

Read/ Writ 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

0

بایت آدرس پایه های ورودی پورت A-A [PORT A INPUT PINS ADDRESS] PIN

Bit 7 6 5 4 3 2 1

0

PORT A7	A6	A5	A4	A3	A2	A1	A0
---------	----	----	----	----	----	----	----

Read/ Writ R/W R/W R/W R/W R/W R/W

R/W R/w

Initial Value N/A N/A N/A N/A N/A N/A N/A

N/A N/A

PINA یک رجیستر نیست. این آدرس دسترسی به مقدار فیزیکی بر روی هر یک از پایه های پورت A را ممکن می سازد. زمانیکه پورت A (PORTA) خوانده می شود، داده لچ (Latch) پورت A خوانده می شود و زمانیکه از PINA خوانده می شود مقدار منطقی که بر روی پایه ها موجود است خوانده می شود.

استفاده از پورت A بعنوان یک I/O عمومی دیجیتال

DDAn	PORT An	I/O	Pull-up	Comment
0	0	Input	No	Tir- state
0	1	Input	Yes	Pan will source current if ext.pulled
1	0	Output	No	Push – pull Zero output
1	1	Output	No	Push – pull one output

جدول تأثیر تغییرات DDAn بر روی پایه های PORTA

تمام 8 پایه موجود زمانیکه بعنوان پایه های I/O دیجیتال استفاده می شوند دارای عملکرد مسوی هستند. Pan، پایه I/o عمومی: بیت DDAn در رجیستر DDRA مشخص کننده جهت پایه است. اگر DDAn یک باشند، Pan بعنوان یک پایه خروجی مورد استفاده قرار می گیرد و اگر DDAn صفر باشد Pan بعنوان یک پایه ورودی در نظر گرفته می شود. اگر Port An یک

باشد هنگامیکه پایه بعنوان ورودی تعریف شود. مقاومت PULL-UP فعال می شود. برای خاموش کردن مقاومت باید Port An صفر شود با اینکه پایه بعنوان خروجی تعریف شود پایه های پورت زمانیکه ری ست اتفاق می افتد به حالت Tri-state می روند.

دیگر کاربردهای پورت A

پورت A بعنوان ADC هم استفاده می شود. اگر تعدادی از پایه های پورت A خروجی تعریف شوند این نکته بسیار مهم است که در زمان نمونه برداری از سیگنال آنالوگ توسط ش ADC، سوئیچ نشوند. این کار ممکن است عملیات تبدیل ACD را نامعتبر کند.

پورت B

پورت B یک I/O دو طرفه 8 بیتی است. سه آدرس از مکان حافظه I/O اختصاص به PORTB دارد. یک آدرس برای رجیستر داده PORTB دومی رجیستر جهت داده DDRB و سومی پایه ورودی پورت B, PINE است. آدرس پایه های ورودی پورت B فقط قابل خواندن است. در صورتیکه رجیستر داده و رجیستر جهت داده هم خواندنی و هم نوشتنی هستند. پایه های پورت دارای مقاومت PULL-UP مجزا هستند. بافر خروجی پورت B می تواند تا 20mA را Sink کند و در نتیجه LED را مستقیماً راه اندازی می کند هنگامیکه PBO-PB7 با مقاومت های Pull-Down خروجی استفاده می شوند. آنها SOURCE جریان می شوند زمانی که مقاومت های Pull-Up داخلی فعال باشند.

رجیسترهای پورت B

رجیسترهای پورت B عبارتند از:

رجیستر داده پورت B – B [PORT B DATA REGISTER]PORTB

Bit 7 6 5 4 3 2 1

0

PORT	PORT	PORTE	PORTE	PORTE	PORTE	PORTE	PORTE
------	------	-------	-------	-------	-------	-------	-------

Read/ Writ 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

رجیستر جهت داده پورت B [PORT DATA DIRECTION REGISTER]DDRB-B

Bit 7 6 5 4 3 2 1

0

DDB7	DDB6	DDB5	DDB4	DDB3	DDB2	DDB1	DDB0
------	------	------	------	------	------	------	------

Read/ Writ 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

0

[PORT B INPUT PINS ADDRESS]PINB-B بیت آدرس پایه های ورودی پورت

Bit	7	6	5	4	3	2	1	0
	PINB7	PINB6	PINB5	PINB4	PINB3	PINB2	PINB1	PINB0

Read/ Writ R/W R/W R/W R/W R/W R/W R/W

R/W R/w

Initial Value N/A N/A N/A N/A N/A N/A N/A

N/A N/A

PINB یک رجیستر نیست. این آدرس دسترسی به مقدار فیزیکی بر روی هر یک از پایه های

پورت B را ممکن می سازد. زمانی که پورت B (PORTB) خوانده می شود. داده لچ (Latch)

پورت B خوانده می شود.

استفاده از پورت B بعنوان یک I/O عمومی دیجیتال

تمام 8 پایه موجود زمانی که بعنوان پایه های I/O دیجیتال استفاده می شوند دارای عملکرد مساوی

هستند. PBn، پایه I/O عمومی: بیت DDBn در رجیستر DDRB مشخص کننده جهت پایه

است. اگر DDBn یک باشد، PBn بعنوان یک پایه خروجی مورد استفاده قرار می گیرد و اگر DDBn صفر باشد، PBn بعنوان یک پایه ورودی در نظر گرفته می شود. اگر PortBn یک باشد هنگامی که پایه بعنوان ورودی تعریف شود، مقاومت Pull-Up فعال می شود برای خاموش کردن مقاومت Pull-Up باید Port Bn صفر شود یا اینکه پایه بعنوان خروجی تعریف شود. پایه های پورت زمانی که ری ست اتفاق می افتد به حالت Tri-state می روند.

DDAn	PORT An	I/O	Pull-up	Comment
0	0	Input	No	Tir- state
0	1	Input	Yes	PBn will source current if ext.pulled
1	0	Output	No	Push – pull Zero output
1	1	Output	No	Push – pull one output

جدول تأثیر تغییرات DDAn بر روی پایه های PORTA

دیگر کاربردهای پورت B

Port Pin	Alternat Functions
PB0	T0(Timer/Counter0External Counter Input)
PB1	T1(Timer/Counter External Counter Input)
PB2	AINO(Analog Comparator Positive Input)
PB3	AINO(Analog Comparator NegativeInput)
PB4	SS(SPI Slave Select Input)
PB5	SPI Bus MasterOutput/Slave Input)
PB6	SPI Bus MasterInput/Slave Input)
PB7	SCK(SPI Bus Serial Clock)

جدول دیگر کاربردهای پورت B

POETB.7-SCK*

SCK: کلاک خروجی Master و کلاک ورودی Slave برای ارتباط SPI است. زمانی که

SPI بعنوان Slave شکل دهی می شود این پایه با توجه به تنظیم DDB7 ورودی و در حالت

Master خروجی تعریف می شود.

PORT 6- MISO *

MISO: ورودی داده Master و خروجی داده Slave که برای ارتباط SPI استفاده می شود.

زمانی که SPI بعنوان Master شکل دهی می شود این پایه با توجه به تنظیمات DDB6 ورودی

و در حالت Slave بعنوان خروجی استفاده می شود.

PORTB5-MOSI *

MOSI: ورودی داده Slave و خروجی داده Master که برای ارتباط SPI استفاده می شود.

زمانی که SPI بعنوان Master شکل دهی می شود این پایه با توجه به تنظیمات DDB5 خروجی و در حالت Slave بعنوان ورودی استفاده می شود.

PORTB.4-SS *

SS: زمانی که SPI بعنوان Slave شکل دهی شود PB4 با توجه به DDB4 ورودی تعریف

می شود و در Slave با Low شدن این پایه SPI فعال می شود. این پایه در Master می تواند خروجی یا ورودی تعریف شود.

PORTB.3-OC0,AINI*

AINI: ورودی منفی مقایسه کننده آنالوگ است.

Oc0: دیگر کاربرد این پایه بعنوان خروجی مد مقایسه ای Timer/Counter0 است. پایه

PB3 با یک کردن DDD7 می تواند برای خروجی مد مقایسه ای Timer/Counter0 شکل

دهی شود.

PORTB.2-INT2,AIN0*

AIN0: ورودی مثبت مقایسه کننده آنالوگ است.

INT2: دیگر کاربرد این پایه بعنوان منبع وقفه خارجی دو است. پایه BP2 می تواند بعنوان

منبع وقفه خارجی برای میکرو (MCU) استفاده شود.

*PORTB.1-T1

T1: ورودی کلاک برای Timer/Counter1 است.

*PORTB.0-XCK, T0 *

T0: ورودی کلاک برای Timer/Counter0 است.

XCK: این پایه نیز می تواند بعنوان کلاک خارجی USART استفاده شود. این پایه فقط زمانی

که USART در مد آسنکرون کار می کند فعال می شود.

پورت C

پورت C یک I/O دو طرفه 8بیتی است. سه آدرس از مکان حافظه I/O اختصاص به PORTC

دارد. یک آدرس برای رجیستر داده PORTC، دومی رجیستر جهت داده DDRC و سومی پایه

ورودی پورت C، PINC، است. آدرس پایه های ورودی پورت C فقط قابل خواندن است. در

صورتی که رجیستر داده و رجیستر جهت داده هم خواندنی و هم نوشتنی هستند. تمام پایه های

پورت دارای مقاومت (Pull-Up) مجزا هستند. بافر خروجی پورت C می تواند تا 20mA را

Sink کند و در نتیجه LED را مستقیماً SOURCE جریان می شوند زمانی که مقاومت های

Pull-Up داخلی فعال باشند.

رجیسترهای پورت C

رجیسترهای پورت C عبارتند از:

[PORT C DATA REGISTER] PORTC-C رجیستر داده پورت

Bit 7 6 5 4 3 2 1

0

PORT	PORT	PORTE	PORTE	PORTE	PORTE	PORTE	PORTE
------	------	-------	-------	-------	-------	-------	-------

Read/ Writ R/W 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

[PORT C DATA DIRECTION REGISTER] DDRC-C رجیستر جهت داده پورت C

REGISTER] DDRC-C

Bit 7 6 5 4 3 2 1

0

DDC 7	DDC 6	DDC 5	DDC 4	DDC 3	DDC 2	DDC 1	DDC 0
-------	-------	-------	-------	-------	-------	-------	-------

Read/ Writ R/W 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

0

[PORT C INPUT PINS ADDRESS]PINC-C بایت آدرس پایه های ورودی پورت

Bit 7 6 5 4 3 2 1

0

PINC 7	PINC 6	PINC 5	PINC 4	PINC 3	PINC 2	PINC 1	PINC 0
--------	--------	--------	--------	--------	--------	--------	--------

Read/ Writ 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

0

PINC یک رجیستر نیست. این آدرس دسترسی به مقدار فیزیکی بر روی هر یک از پایه های

پورت C را ممکن می سازد. زمانی که پورت C (PORTC) خوانده می شود داده لچ (Latch)

پورت C خوانده می شود و زمانیکه از PINC خوانده می شود مقدار منطقی که بر روی پایه ها

موجود است خواند می شود.

استفاده از پورت C بعنوان یک I/O عمومی دیجیتال

تمام 8 پایه موجود زمانیکه بعنوان پایه های I/O دارای عملکرد مساوی هستند. PCn، پایه I/O عمومی بیت DDCn در رجیستر DDRC مشخص کننده جهت پایه است. اگر DDCn یک باشد، PCn بعنوان یک پایه خروجی مورد استفاده قرار می گیرد. و اگر DDCn صفر باشد، PCn بعنوان یک پایه ورودی در نظر گرفته می شود. برای خاموش کردن مقاومت Pull-Up باید PortCn صفر شود. یا اینکه پایه بعنوان خروجی تعریف شود. پایه های پورت در زمانی که ری ست اتفاق می افتد به حالت Tri-State می روند.

DDAn	PORT An	I/O	Pull-up	Comment
0	0	Input	No	Tri- state
0	1	Input	Yes	PCn will source current if ext.pulled
1	0	Output	No	Push – pull Zero output
1	1	Output	No	Push – pull one output

جدول تأثیر تغییرات DDAn بر روی پایه های PORTA

دیگر کاربردهای پورت C

Port Pin	Alternat Functions
PC0	TOSC2(Timer Oscillator Pin 2)
PC1	TOSC1 (Timer Oscillator Pin1)
PC2	TDI(JTAG Test Data In)
PC3	TDO(JTAG TEST Data Out)
PC4	TMS(JTAG Test Mode Select)
PC5	TCK(JTAG Test Clock)
PC6	SSDA(Tow – Wire Serial Bus Data Input /Output)
PC7	SCL(Tow- Wire serial Bus Clock Line)

جدول دیگر کاربردهای پورت C

PORTC.7-TOSC2*

TOSC2: زمانی که تایمر / کانتر 2 در مد آسنکرون کار می کند به این پایه و پایه TOSC1

کریستال ساعت متصل می شود. در این حالت دیگر نمی توان این پایه را بعنوان I/O استفاده

نمود.

PORTC.6 –TOSC1*

TOSC1: زمانی که تایمر / کانتر 2 در مد آسنکرون کار می کند به این پایه و پایه TOSC2

کریستال ساعت متصل می شود. در این حالت دیگر نمی توان این پایه را بعنوان I/O استفاده

نمود.

PORTC.5 –TDI*

TDI: در زمان ارتباط JTAG بعنوان ورودی داده سریال عمل می کند و دیگر نمی توان از این پایه بعنوان I/O استفاده نمود.

PORTC.4 – TDO*

TDO: در زمان ارتباط JTAG بعنوان خروجی داده سریال عمل می کند و دیگر نمی توان از این پایه بعنوان I/O استفاده نمود.

PORTC.3-TMS*

TMS: در زمان ارتباط JTAG استفاده می شود و دیگر نمی توان از این پایه بعنوان I/O استفاده نمود.

PORTC.1-SDA*

SDA: در زمان ارتباط 2-WIRE بعنوان خط داده استفاده می شود.

PORTC.0-SCL*

SCL: در زمان ارتباط 2-WIRE بعنوان خط کلاک استفاده می شود.

پورت:

پورت D یک I/O دو طرفه 8 بیت ی است. سه آدرس از مکان حافظه I/O اختصاص

به PORTD دارد. یک آدرس برای رجیستر داده PORTD، دومی رجیستر جهت داده DDRD

و سومی پایه ورودی پورت PIND_D است آدرس پایه های ورودی پورت D فقط قابل

خواندن است. در صورتی که رجیستر داده و رجیستر جهت داده هم خواندنی و هم نوشتنی هستند تمام پایه های پورت دارای مقاومت Pull-UP مجزا هستند. بافر خروجی پورت D می تواند تا 20mA Sink کند و در نتیجه LED را مستقیماً راه اندازی می کند هنگامیکه PD0-PD7 با مقاومت های Pull-Down خروجی استفاده می شوند. آنها SOURCE جریان می شوند زمانی که مقاومت های Pull-Up داخلی فعال باشد.

رجیستر های پورت D

رجیسترهای پورت D عبارتند از:

رجیستر داده پورت - [PORTD] [PORT D DATA REGISTER] ی

Bit 7 6 5 4 3 2 1

0

PORT	PORT	PORTI	PORTI	PORTI	PORTI	PORTI	PORTI
------	------	-------	-------	-------	-------	-------	-------

Read/ Writ 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

[PORT C DATA DIRECTION REGISTER] رجیستر جهت داده پورت D

REGISTER]DDRD-

Bit 7 6 5 4 3 2 1

0

DDD 7	DDD 6	DDD 5	DDD 4	DDD3	DDD2	1DDD	DDD0
-------	-------	-------	-------	------	------	------	------

Read/ Writ 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

0

[PORT C INPUT PINS ADDRESS]PINC-C بایت آدرس پایه های ورودی پورت

Bit 7 6 5 4 3 2 1

0

PIND 7	PIND 6	PIND 5	PIND 4	PIND 3	PIND2	PIND1	PIND 0
--------	--------	--------	--------	--------	-------	-------	--------

Read/ Writ R/W R/W R/W R/W R/W R/W

R/W R/w

Initial Value	N/A	N/A	N/A	N/A	N/A	N/A
N/A	N/A					

PIND یک رجیستر نیست. این آدرس دسترسی به مقدار فیزیکی بر روی هر یک از پایه های پورت D را ممکن می سازد. زمانیکه پورت D (PORTD) خوانده می شود ریال داده لچ (Latch) پورت D خوانده می شود و زمانی که از PIND خوانده می شود مقدار منطقی که بر روی پایه ها موجود است خوانده می شود.

استفاده از پورت D بعنوان یک I/O عمومی دیجیتال:

تمام 8 پایه موجود زمانیکه بعنوان پایه های I/O دیجیتال استفاده می شود دارای عملکرد مساوی هستند PDn، پایه I/O عمومی: بیت DDDn در رجیستر DDRD مشخص کننده جهت پایه است. اگر DDDn یک باشد، PDn یک پایه خروجی مورد استفاده قرار می گیرد و اگر DDDn صفر باشد، PDn، PDn بعنوان یک پایه ورودی در نظر گرفته می شود. اگر PortDn یک باشد هنگامیکه پایه بعنوان ورودی تعریف شود مقاومت Pull-Up فعال می شود. برای خاموش کردن مقاومت Pull-Up باید Port Dn صفر شود یا اینکه پایه بعنوان خروجی تعریف می شود. پایه های پورت در زمانیکه رس ست اتفاق می افتد به حالت Tri-State می روند.

دیگر کاربردهای پورت D

Port Pir	Alternat Functions
PD0	RDX(UART INPUT LINE)
PD1	TDX (UART OUTPUT LIN)
PD2	INT0(EXTRNAL INTERRUPT 0 INPUT)
PD3	INT(EXTERNAL INTERRUPT 1 INPUT)
PD4	OC1B(T/C1 OUTPUT COMPAREB MATCH OUT
PD5	SOC1A (T/C 1 OUTPUT COMPAREA MATCH JT
PD6	IICP (T/C INPUT CAPTUR PIN)
PD7	OC2(T/C2 OUTPUT COMPARE MATCH OUTPUT

جدول دیگر کاربردهای پورت D

PORTD,7-OC2 *

OCS: خروجی مد مقایسه ای تایمر / کانتر PD7.2 با یک شدن DDD7 می تواند بعنوان پایه خروجی مد

مقایسه ای Timer/Counter2 شکل دهی شود این پایه همچنین برای خروجی PWM تایمر استفاده

می شود.

PORTD.6-ICP *

ICP: PD6 می تواند بعنوان پایه ورودی CAPTURE تایمر / کانتر 1 عمل کند.

PORT.5-OC1A

OC1A: خروجی مد مقایسه ای Timer/Counter1 پایه PD5 با یک شدن DDD5 می تواند برای

خروجی مد مقایسه ای Timer/Counter1 شکل دهی شود. این پایه همچنین برای خروجی PWM

تایمر 1 استفاده می شود.

PORTD.4-OC1B*

OC1B: خروجی مد مقایسه ای Timer/Counter 1

پایه PD4 با یک شدن DDD4 می تواند برای خروجی مد مقایسه ای Timer/Counter1 شکل دهی

شود این پایه همچنین برای خروجی PWM تایمر استفاده می شود.

PORTD.3-INT1*

INT1: منبع وقفه خارجی یک.

پایه PD3 می تواند بعنوان منبع وقفه خارجی برای میکرو استفاده شود.

PORTD.2-INT0

INT0: منبع وقفه خارجی صفر

پایه PD2 می تواند بعنوان منبع وقفه خارجی برای میکرو استفاده شود.

PORTD.1-TXD*

TDX: ارسال داده (پایه خروجی داده برای USART)

زمانی که ارسال USART فعال می شود پایه با توجه به DDD1 بعنوان خروجی شکل دهی می شود.

PORTD.0-RXD*

RXD: دریافت داده (پایه ورودی داده برای USART)

زمانی که دریافت USART فعال می شود پایه با توجه به DDD0 بعنوان ورودی شکل دهی می شود.

مختصری راجع به AVR

زبانهای سطح بالا یا همان HLL (HIGH LEVEL LANGUAGES) به سرعت در حال تبدیل شدن به زبان برنامه نویسی استاندارد برای میکروکنترلرها (MCU) حتی برای میکروهای 8 بیتی کوچک هستند. زبان برنامه نویسی BASIC و C بیشترین استفاده را در برنامه نویسی میکروها دارند ولی در اکثر کاربردها کدهای بیشتری را نسبت به زبان برنامه نویسی اسمبلی تولید می کنند. ATMEL ایجاد تحولی در معماری، جهت کاهش کد به مقدار مینیمم را درک کرد که نتیجه این تحول میکروکنترلرهای AVR هستند که علاوه بر کاهش و بهینه سازی مقدار کدها به طور واقع عملیات را تنها در یک کلاک سیکل توسط معماری (REDUCED RISC (INSTRUCTION SET COMPUTER انجام می دهند و از 32 رجیستر همه منظوره

(ACCUMULATORS) استفاده می کنند که باعث شده 4 تا 12 بار سریعتر از میکروهای مورد استفاده کنونی باشند.

تکنولوژی حافظه کم مصرف غیر فرار شرکت ATMEL برای برنامه ریزی AVR ها مورد استفاده قرار گرفته است در نتیجه حافظه های FLASH و EEPROM در داخل مدار قابل برنامه ریزی (ISP) هستند. میکروکنترلرهای اولیه AVR دارای 1، 2 و 8 کیلو بایت حافظه FLASH و به صورت کلمات 16 بیتی سازماندهی شده بودند.

AVR ها به عنوان میکروهای RISC با دستورات فراوان طراحی شده اند که باعث می شود حجم کد تولید شده کم و سرعت بالاتری بدست آید.

عملیات تک سیکل

با انجام تک سیکل دستورات، کلاک اسیلاتور با کلاک داخلی سیستم یکی می شود. هیچ تقسیم کننده ای در داخل AVR قرار ندارد که ایجاد اختلاف فاز کلاک کند. اکثر میکروها کلاک اسیلاتور به سیستم را با نسبت 1:4 یا 1:12 تقسیم می کنند که خود باعث کاهش سرعت می شود. بنابراین AVR ها 4 تا 12 بار سریعتر و مصرف آنها نیز 4-12 بار نسبت به میکروکنترلرهای مصرفی کنونی کمتر است زیرا در تکنولوژی CMOS استفاده شده در میکروهای AVR، مصرف توان منطقی متناسب با فرکانس است.

نمودار زیر افزایش MIPS (MILLION INSTRUCTION PER SECONDS) را به علت انجام عملیات تک سیکل AVR (نسبت 1:1) در مقایسه با نسبت های 1:4 و 1:12 در دیگر میکروها را نشان می دهد.

نمودار مقایسه افزایش MIPS/POWER Consumption در AVR با دیگر میکروکنترلرها

طراحی برای زبانهای BASIC و C

زبانهای BASIC و C بیشترین استفاده را در دنیای امروز بعنوان زبانهای HLL دارند. تا امروزه معماری بیشتر میکروها برای زبان اسمبلی طراحی شده و کمتر از زبانهای HLL حمایت کرده اند.

هدف ATMEL طراحی معماری بود که هم برای زبان اسمبلی و هم زبانهای HLL مفید باشد. به طور مثال در زبانهای C و BASIC می توان یک متغیر محلی به جای متغیر سراسری در داخل زیر برنامه تعریف کرد، در این صورت فقط در زمان اجرای زیر برنامه مکانی از حافظه RAM برای متغیر اشغال می شود در صورتی که اگر متغیری به عنوان سراسری تعریف گردد در تمام وقت مکانی از حافظه FLASH ROM را اشغال کرده است.

برای دسترسی سریعتر به متغیرهای محلی و کاهش کد، نیاز به افزایش رجیسترهای همه منظوره است. AVR ها دارای 32 رجیستر هستند که مستقیماً به (UNIT

ARITHMETIC)LOGIC ALU متصل شده اند، و تنها در یک کلاک سیکل به این واحد

دسترسی پیدا می کنند. سه جفت از این رجیسترها می توانند به عنوان رجیسترهای 16 بیتی

استفاده شوند.

برنامه صفحه بعد نشان می دهد که چگونه تعداد مناسب رجیسترهای همه منظوره (در AVR

ها) می توانند با معماری CISC با یک ACCUMULATOR مقایسه گردند. برای این منظور

میخواهیم از معادله صفحه بعد A را بدست بیاوریم. می بینیم که با کدهای AVR این محاسبه در

عرض 4 کلاک سیکل و با کدهای CISC در عرض 48-96 کلاک سیکل انجام می گیرد.

نتیجه تمام موارد بحث شده، میکروکنترلرهای AVR با سرعت بالا و سازماندهی RISC هستند.

میکروکنترلرهای AVR به سه نوع AT90S یا AVR، TINYAVR و MEGAAVR

تقسیم بندی شده اند.