عنوان:

طراحی و ساخت Logic Analyzer با کامپیوتر

فهرست:

عنوان	صفحه
مقدمه	۱
فصل ۱: طراحی بورد و سخت افزار	۲
نمونه برداری و ذخیره سازی	۵
مدار آشکار ساز TTL/CMOS	۵
طراحی قسمت نمونه برداری	۷
مدار منبع تغذیه	۱۱
نحوه کارکرد مدار	۱۲
فصل ۲: USB و مدارات مربوط به آن	۲۰
فصل ۳: نرم افزار کامپیوتری	۴۱
فصل۴: طرح های پیشنهادی	۴۶
مراجع	۵۵
ضمیمه	۵۶

مقدمه :

با پیشرفت علم و تولید سیستم دیجیتالی و استفاده و رواج آن ها باعث شد که برای عیب یابی یا کنترل طرز کار آنها مقادیر دیجیتالی مشاهده یا ثبت بشوند.یکی از راه های مشاهده سیگنال ها اسیلوسکوپ بود ولی تعداد کانالهای ورودی اسکوپ کم میباشد این وسیله برای این کار چندان کاربردی نداشت بنابریان باید دستگاهی با قابلیت نمایش تعداد ورودی های زیاد و ساده تر از اسکوپ ساخته میشد دستگاه ساخته شده لاجیک آنالایزر نام داشت این دستگاه علاوه بر نمایش ۰ و ۱ قابلیت تشخیص Z و نمایش آن نیز بود یک مشکل اساسی این دستگاه ها مانند اسکوپ ها حجم بزرگ انها و سنگینی آنها بود . برای رفع این مشکل میتوانستیم از یک واحد پردازشی نمایشی استفاده کنیم این مشکل با استفاده از کامپیوتر حل می شد .از انجا که امروزه کامپیوتر ها رایج شده اند و استفاده از انها عمومی شده این کار زیاد دور از ذهن نبود . بنابراین یک واحد نمونه برداری خارج از کامپیوتر وظیفه نمونه برداری و ارسال از طریق یک رابط به کامپیوتر را بر عهده داشت و کار نمایش و پردازش در کامپیوتر انجام می شد بنابراین دستگاه ساخته شده سبک و ارزان و کوچک بودو استفاده از ان به مراتب ساده تر شده بود.

لاجیک آنالایزر ساخته شده دارای ۳ بخش کامپیوتری . سخت افزار و ارتباط میباشد که به صورت مجزا در فصلهای بعدی بررسی می شوند.

در فصل اول طراحی کلی سیستم و طرز کار ان گفته خواهد شد.

در فصل دوم رابط کاربری USB و ویژگی ها و طرز کار و طریقه پیاده سازی آن مطرح می شود.

در فصل سوم برنامه کامپیوتری که کار نمایش و ارتباط با سخت افزار را بر عهده دارد توضیح داده می شود.

فصل چهارم پیشنهاداتی هستند که به دلیل دانشجویی بودن پروژه قابل انجام نبودند بررسی می شود.

فصل اول

طراحی بورد و سخت‌افزار

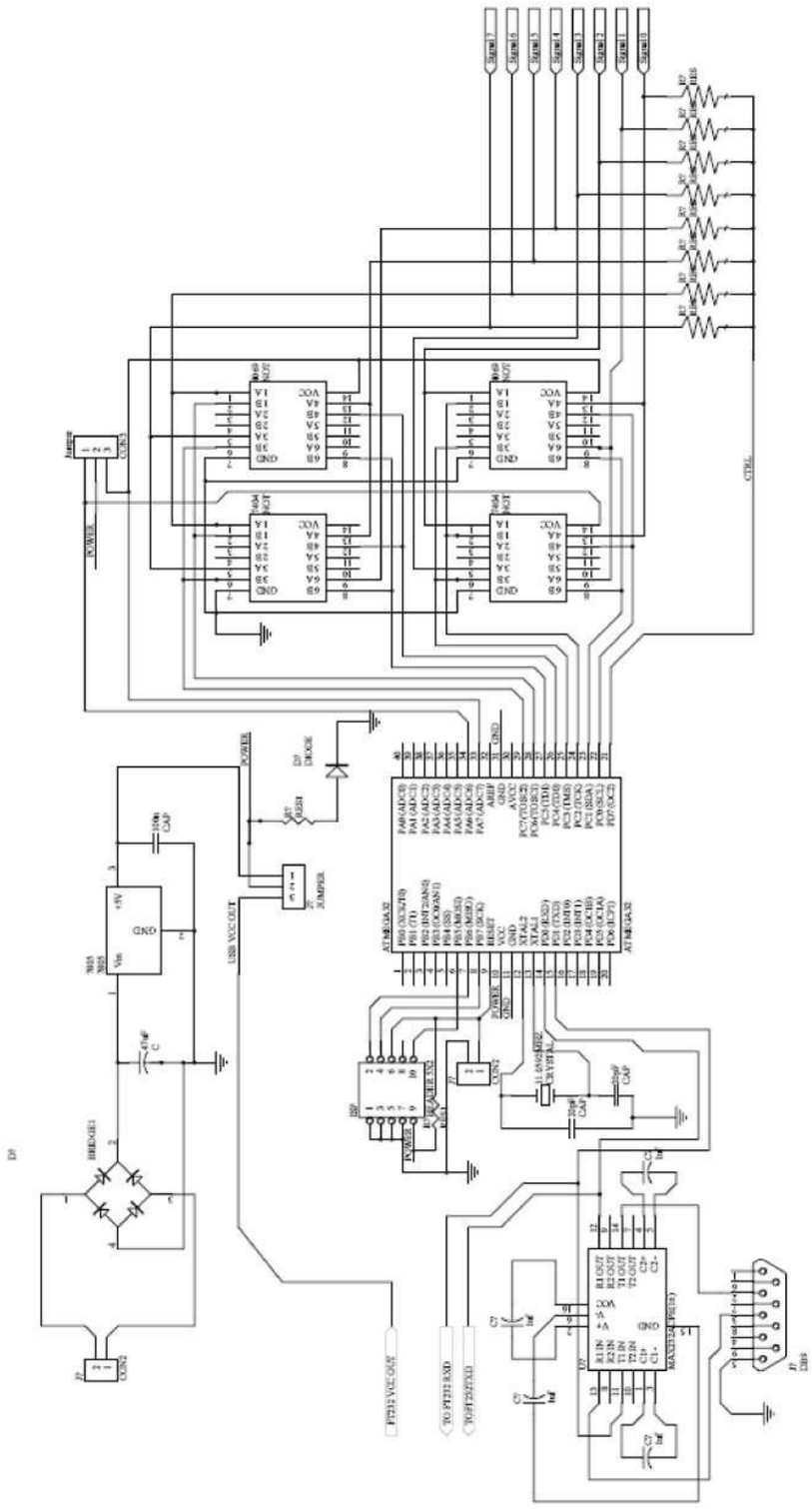
سخت افزار مربوط به لاجیک آنالایزراز سه بخش تشکیل شده است .

۱ -بخش نمونه برداری و ذخیره ساز ی

۲ -بخش تغذیه

۳ -بخش ارتباط با کامپیوتر که وظیفه تبدیل سریال به USB را دارد .

در شکل ۱ شماتیک مدار طراحی شده را می بینید.



شکل ۱: شماتیک مدار طراحی شده

نمونه‌برداری وذخیره‌سازی :

برای نمونه‌برداری نیاز به تشخیص حالتهای ۰ و ۱ و Z می‌باشد که ۰ و ۱ به راحتی قابل تشخیص هستند ولی حالت Z (High Impedance) که در نمونه‌گیری از BUS ها یا برای اطلاع از قطعی سیم موردنیاز است، می‌بایست با استفاده از یک مدار آشکار شود. برای آشکارسازی سه حالت ۰ و ۱ و Z نیاز به ۲ بیت می‌باشد ، بنابراین در حالت Z دوبار از سیگنال نمونه‌برداری می شود. با توجه به شکل ۲ این کار را می‌توان با یک مقاومت که در ۲ حالت Pull up و Pull Down قرار می‌گیرد انجام داد . پایه ctrl که از میکرو فرمان می‌گیرد در ۲ سیکل نمونه‌برداری ۲ مقدار ۱ و ۰ را به این قسمت اعمال می‌کند . بنابراین ورودی های میکرو طبق جدول ۱ خواهند بود.

Pull Down	Pull Up	
0	0	0
1	1	1
0	1	Z

جدول ۱: حالت های مختلف ورودی

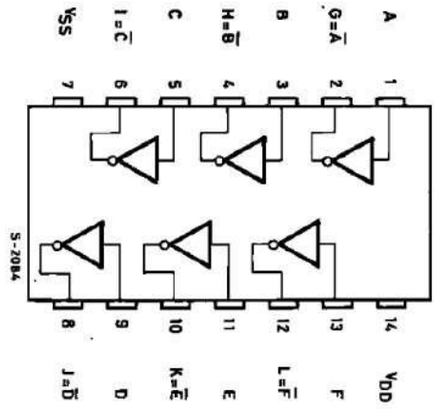
همانطور که در جدول مشاهده می‌شود حالت Z آشکار می‌شود یعنی در مدهای کاری که نیاز به آشکارسازی Z نمی‌باشد می‌توان پایه Ctrl را ۱ قرار بدهیم در نتیجه Z و ۱، یک حالت در نظر گرفته می‌شوند .

مدار آشکارساز CMOS/TTL :

در مدارهای الکترونیکی جدید دو خانواده بزرگ از آی سی های منطقی برتری خاصی پیدا کرده‌اند که ای سی های سری ۴۰۰۰ با سرعت متوسط و آی سی‌های سریع سری ۷۴ را شامل می شوند دو تکنولو ژی رقیب در ساخت این IC ها عبارتند از (TTL (Transistor Transistor Logic و CMOS (complementary mos – fet symmetry) که تاریخ همه آنها به دهه ۱۹۷۰ باز می‌گردد . به

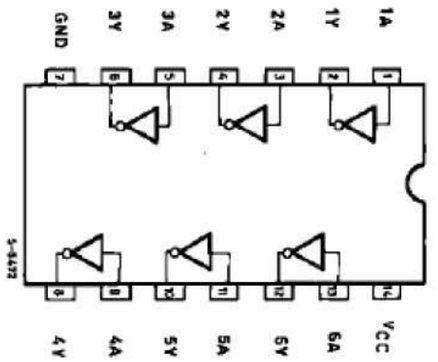
دلیل وجود تفاوت در سطح ولتا ژهای منطقی در این دو نوع IC ها و اهمیت استفاده از ۲ سری نیاز به یک مدار آشکارسازی این دو سری داریم و همانطور که می‌دانیم هر دو سری دارای خروجی CMOS می‌باشند . پس می‌توان از یک NOT به صورت ساده برای این کار استفاده کرد.

در شکل ۲ آی سی 4069 NOT که دارای ورودی و خروجی CMOS است را مشاهده می کنید .



شکل ۲: Pin out آی سی 4069

۷۴۰۴ مشابه ۴۰۶۹ می‌باشد که دارای PINOUT کاملا مشابه با IC ۴۰۶۹ می‌باشد ولی با این تفاوت که ورودی TTL و خروجی CMOS می‌باشد.



شکل ۳ : Pin Out آی سی ۷۴۰۴

بنابراین با توضیحات بالا مشاهده می‌شود که در صورت برقرار شدن مسیر ۴۰۶۹ سیگنال‌ها با سطح ولتا ژ CMOS آشکارسازی و با مسیر دوم یعنی از طریق ۷۴۰۴ به صورت TTL آشکارسازی می‌شوند.

طراحی قسمت نمونه‌برداری :

برای این بخش یک قسمت مرکزی برای نمونه‌گیری و پردازش و ذخیره‌سازی و ارسال اطلاعات لازم است یعنی این بخش وظیفه هماهنگ سازی و ارتباط و کنترل تمام قسمت‌ها را بر عهده دارد که این قسمت یک کنترل کننده مرکزی با قابلیت برنامه‌پذیری می‌بایست باشد که می‌توان از میکروکنترلر یا FPGA استفاده کرد که هر کدام از آنها مزایا و معایب خاص خود را دارند.

مزایای میکروکنترلرها

۱ - برنامه‌پذیری و قابلیت برنامه‌نویسی با زبان‌هایی سطح بالا

۲ - آسانی استفاده و ساخت

۳ - رایج بودن

مزایای FPGA

۱ - قابلیت کارکرد با سرعت‌های بالا

۲ - ورودی و خروجی‌ها تعداد بالا

۳ - پردازش موازی

معایب میکروکنترلرها

۱ - سرعت کم

۲ - سرعت کم ارتباط با سریال

۳ - کمبود تعداد پایه‌ها

معایب FPGA

۱ -سختی کار کردن و ساخت سخت افزار

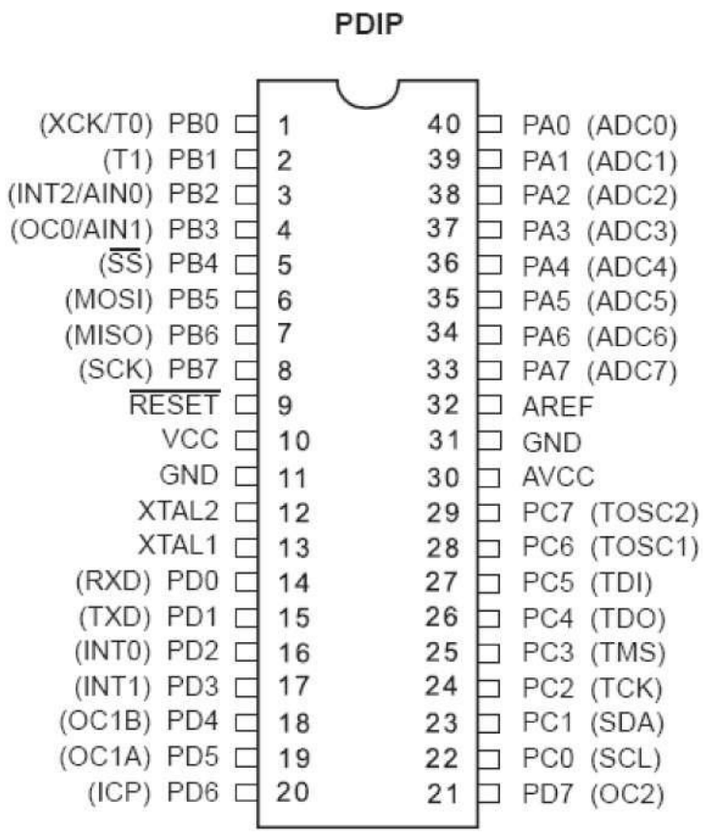
۲ -رایج نبودن و گران بودن

۳ -مشکلات پیاده‌سازی الگوریتم

بنابراین با توجه به مزایا و معایب بالا به دلیل دانشجویی بودن کاربرد و ساخت استفاده از میکرو به صرفه‌تر می‌باشد . خانواده‌های زیادی برای میکرو وجود دارد که سری AVR به دلیل راحتی کارکرد و زبان برنامه‌نویسی C و امکانات مناسب برای انجام این بخش در نظر گرفته شده است.

AVR : میکروکنترلهای این سری دارای امکانات مناسبی می‌باشند که در این پروژه به صرفه می‌باشد .

ATMEGA32:



شکل ۴ : Pin Out Atmega32

این میکرو کنترلرهای ۸ بیتی دارای توان مصرفی پایینی است . و در معماری آنها از ساختار پیشرفته RISC (کم کردن دستورالعملهای میکروکنترلها) بهره گرفته شده است به عبارت دیگر این میکروکنترلها دارای ۱۳۱ دستورالعمل هستند که اغلب آنها در یک پالس ساعت اجرا می‌شوند . اجرا شدن دستورالعلمها در یک سیکل باعث افزایش سرعت این میکروکنترلها گردیده است .

این میکرو دارای ۳۲ رجیستر همه منظوره ۸ بیتی است و قابلیت اجرای حداکثر ۱۶ میلیون دستورالعمل در ثانیه دارد این قابلیت یکی دیگر از دلایل افزایش سرعت این میکرو می‌باشند .

ATMEGA32: دارای 32 KB حافظه Flash با قابلیت خواندن و نوشتن تا ۱۰ هزار مرتبه و ۵۱۲ بایت حافظه EEPROM با قابلیت خواندن و نوشتن تا صد هزار بار و 2KB حافظه داخلی SRAM می‌باشد.

تذکر: حافظه داخلی SRAM می‌تواند به عنوان RAM استفاده شود، بنابراین در حالت نمونه برداری می‌توان به عنوان فضای ذخیره سازی نمونه استفاده شود .

برای برنامه ریزی میکروکنترلرهای AVR می‌توان از استاندارد JTAG استفاده نمود این استاندارد برای برنامه‌ریزی FLASH و EERPOM فیوزبیت‌ها و LOCK BIT از طریق رابط JTAG به کار می‌رود .

تجهیزات جانبی :

۱ -دارای دو شمارنده ۸ بیتی و یک شمارنده ۱۶ بیتی است فرکانس کار این شمارنده‌ها به صورت مجزا تنظیم می‌شود این شمارنده‌ها دارای واحدمقایسه می‌باشند که برای ایجاد موج PWM می‌توان استفاده کرد .

۲ -مبدل ADC با ۸ کانال ۱۰ بیتی

۳ -رابط سریال TWI که اتصال دو یا چندین میکرو را توسط دوباس دیتا و پالس فراهم می‌کند

۴ - قابلیت ارتباط سریال USART از دیگر مشخصات این میکروکنترلرها می‌باشد. توضیح اینکه ارتباط با استفاده از پورت سریال USART به دو صورت سنکرون و آسنکرون صورت می‌گیرد در حالت سنکرون از یک پالس ساعت برای همزمانی استفاده می‌شود و در حالت آسنکرون میکروکنترلها ورود و خروج اطلاعات را کنترل کرده و برنامه‌ریزی در این حالت ساده‌تر است .

۵ - دارای رابط سریال SPI است که در دو مد Master /slave به کار گرفته می‌شود .

۶ - شمارنده watchdog با اسیلاتور جداگانه که برای جلوگیری از هنگ کردن میکروکنترلر به کار می‌رود . در صورتی که تنظیمات لازم برای فعال شدن این شمارنده انجام شده باشد . با شروع به کار میکروکنترلر در این شمارنده شروع به کار می کند برنامه‌نویس با توجه به مدت زمان اجرای دستورالعمل ها در زمان مشخصی قبل از رسیدن شمارنده به انتهای سیکل کاری خود با استفاده از دستور WDR شمارنده را Reset می‌کند حال اگر میکروکنترلر به دلایلی از کار اف ئده باشد . دستور WDR اجرا نشده شمارنده Reset نمی‌شود در نتیجه watchdog تا انتهای سیکل کاری خود شمارش می‌کند و میکرو را Reset نموده خود از ابتدا شروع به شمارش می‌کند .

در این طراحی خروجی‌های ICهای NOT را به پورت Cوصل میکنیم در نتیجه پورت Cبه عنوان ورودی وظیفه دریافت اطلاعات را از سیگنال‌ها به عهده دارد.

یعنی خروجی ICهای NOTبعد از تبدیل سیگنال ورودی از TTL/ CMOS (بسته به انتخاب) به CMOS به این ورودی وصل می‌شود .

PORTC: P [22..29]

وظیفه کنترل Ctrl (pull up یا pull down) توسط PROTD.7 انجام می‌شود بنابراین این پایه یک پایه خروجی می‌باشد که بسته به حالت تشخیص Z دارای مقادیر زیر می‌باشد .

Z عدم تشخیص ➔ PORTD .7 = 1

در یک سیکل قبل از دریافت PORTD.7=1 ➔ تشخیص Z

در سیکل بعدی PORTD .7 = 0

دو پایه موردنیاز دیگر برای پایه‌های وقفه می‌باشد که از پایه‌های INT0 و INT1 برای این کار استفاده شده است این پایه‌ها برای حالتی به کار می‌رود که بخواهیم از کلاک یا تریگر خارجی استفاده کنیم به کابرده می‌شود یعنی بسته به مودکاری (لبه بالا رونده یا پایین رونده که در برنامه تنظیم می‌شود) این دو پایه برای کلاک یا تریگر خارجی استفاده می‌شود .

مدار منبع تغذیه

چون تغذیه تمام IC ها 5V می‌باشد و جریان مصرفی مدار پایین می‌باشد و به دلیل کم مصرف بودن اجزای مدار این قسمت از مدار خیلی ساده می‌باشد .

در این مدار بعد از یک سو سازی و گذشتن از فیلتر خازنی با استفاده از رگ‌ولاتور ۷۸۰۵ سطح ولتاژ به ۵ می‌رسد در طراحی قسمت تغذیه همچنین امکان استفاده از خروجی USBکه دارای قدرت ۵۰۰ میلی آمپر است . می‌توان استفاده کرد یعنی با استفاده از یک جا م پر می‌توان قسمت تغذیه را از خروجی رگولاتور یا از خروجی USB گرفت بنابراین سخت‌افزار ما می‌تواند بدون منبع استفاده شود یعنی با قابلیت استفاده از خروجی تغذیه USB:

نحوه کارکرد مدار :

با توجه به توضیحاتی که در مورد قسمت‌های مختلف و نحوه ارتباط آنها داده شده در این قسمت نحوه عملکرد کلی مدار و برنامه آن توضیح داده می‌شد (سخت‌افزار موردنظر دارای ۲ مدکاری می‌باشد) .

۱- Real time که در این مد نمونه‌برداری به صورت مستقیم می‌باشد یعنی بعد از هر سیکل نمونه‌برداری داده‌ها به کامپیوتر ارسال می‌شوند . فرکانس کاری در این حالت به سرعت ارسال محدود می‌شود .

۲- مد storage در این حالت کاری ارسال داده‌ها بعد از نمونه‌گیری انجام می‌شود یعنی بعد از ذخیره مقدار مشخصی sample در حافظه داده ها ارسال می‌شوند فرکانس کاری در این حالت به سرعت اجرای دستورات میکرو محدود می‌شود و طبعاً دارای سرعت بیشتری می‌باشد .

برای ارتباط میکرو از قسمت سریال آن استفاده شده است در این ارتباط از حداکثر سرعت ارتباط سریال بهره گرفته شده است تا در حالت Real time حداکثر سرعت sample برداری را داشته باشیم برای این منظور برای کلاک میکرو کریستالهایی که مخصوص سریال می‌باشد استفاده کردیم تا حداقل مقدار خطا را داشته باشد . کریستال استفاده شده 11.0592 MHZ می‌باشد بنابراین می‌توان حداکثر باودریت را داشته باشد . یعنی مقدار با وود ریت ارسال و دریافت برابر 115200b/s می‌باشد

توضیحات « سریال »

تابع وقفه receive مربوط به USART وظیفه دریافت اطلاعات و تبدیل بیت کنترلی ارسالی از کامپیوتر به متغیرهای کنترلی موجود را دارد .

بیت کنترلی

Bit0 : انتخاب مد کاری Storage , Real time

BIT0

0 Real time

1 Storage

Bit1: برای انتخاب آشکارسازی z یا عدم آشکار سازی z می‌باشد.

BIT 1

عدم آشکار سازی ۰

آشکارسازی ۱

Bit (۲ و ۳) انتخاب منبع کلاک و تریگر :

منبع کلاک	Bit 3 2
Internal	۰ ۰
Rising Edge	۰ ۱
Falling Edge	۱ ۰
Any change	۱ ۱

جدول ۲: انتخاب منبع کلاک و مد کاری

Bit [4..7]

انتخاب فرکانس کاری

100HZ	۰ ۰ ۰
1 KHZ	۰ ۰ ۱
10 KHZ	۰ ۱ ۰
100 KHZ	۰ ۱ ۱
1 MHZ	۱ ۰ ۰

جدول ۳ : انتخاب فرکانس کاری

بنابراین بعد از تبدیل شدن بیت کنترلی به متغیرهای کنترلی برنامه با استفاده از شرط‌ها وظیفه کنترل سخت‌افزار را به عهده دارد.

Real time با منبع کلاک داخلی:

در این حالت مدار با حداکثر فرکانس اطلاعات را به سریال ارسال می‌کند بنابراین سرعت نمونه‌برداری به سرعت ارسال محدود می‌شود یعنی اگر هر کدام از فریم‌های سریال را در نظر بگیریم . ۸ بیت داده + ۱ بیت stop + ۱ بیت start + ۱بیت Parity =۱۱ بیت

115200/11=10KB/Sec

در این حالت برای کم کردن دستورات و چک نکردن شرط‌ها در حلقه ارسال و نمونه‌گیری برای حالت z و بدون z دو حلقه در نظر بگیریم . در نتیجه برنامه با توجه به بیت کنترلی حالت Z را تشخیص داده و حلقه موردنظر را اجرا می‌کند .

```
if(z==1)
{
        while(modechange)
        {
          PORTD.7=1;
          putchar(PINC);
          PORTD.7=0;
          putchar(PINC);
        }
}
else if(z==0)
{
         PORTD.7=1;
        while(modechange)
        {
           putchar(PINC);
        }
}
```

اکسترنال کلاک در حالت Real time

در این حالت در صورت وقوع تغییرات با لبه کلاک داده‌ها و عرض پالس تغییرات ارسال می‌شود که فرکانس کاری به عنوان Resolution استفاده می‌شود یعنی مرجع شمارش ما همان فرکانس کاری می‌باشد.

بنابراین ما نیاز به یک Timer برای شمارش داریم

Timer ها در AVR

Timer / counter0 یک شمارنده ۸ بیتی تک کاناله می‌باشد که دارای مدهای کاری نرمال و CTC می‌باشد و در صورت رسیدن عدد شمارش به یک مقدار خاص شمارنده Reset می‌شود برای کار کردن با این تایمر ما نیاز به مد CTC داریم در این مدکاری در صورت رسیدن مقدار شمارنده یعنی TCNT0 به مقدار OCR0 شمارنده Reset شده و وقفه Compare Match فعال می‌شود بنابراین ما با تنظیم فرکانس کاری تایمر و مقدار OCR0 می‌توانیم مقدار فرکانس موردنظر را داشته باشیم که این تنظیم ات نیز با استفاده از رجستر TCCR0 انجام می‌شود.

7	6	5	4	3	2	1	0	
FOC0	WGM00	COM01	COM00	WGM01	CS02	CS01	CS00	TCCR0
W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
0	0	0	0	0	0	0	0	

شکل ۵ : رجیستر کنترلی تایمر صفر

Bit 7: foc0 = FORCE ONTPUT COMPAR

بیت FOC0 در مدهای نرمال و CTC به کار می‌رود و در مدهای PWM باید صفر باشد با نوشتن یک در این بیت یک سیگنال Match به مولد شکل موج اعمال می‌شود در این حالت خروجی OC0 نیز با توجه به تنظیمات بیتهای COM01 , COM00 تغییر می‌کند بیت FOCO مقدار شمارنده را تغییر نمی‌دهد و منجر به فعال شدن وقفه‌ای نمی‌شود این بیت همیشه صفر خوانده می‌شود .

BIT 6: WGM 00, 01 : WAVE FORM Generation Mode

ترتیب شمارش شمارنده را کنترل می‌کند منبع تعیین کننده ماکزیمم شمارنده (مدکاری توسط این بیت تنظیم می‌شود) .

WGM1	WGMO	
0	0	NORML
0	1	PWM PHASE CORRECT
1	0	CTC
1	1	FAST PWM

جدول ۴

نحوه رفتار پایه OCO را تعیین می‌کند اگر حداقل یکی از بیت‌ها یک شود پورت میکروکنترلر سروظیفه واحد مقایسه را به عهده می‌گیرد در نتیجه پورت B نیز می‌باشد به صورت خروجی تعریف شود .

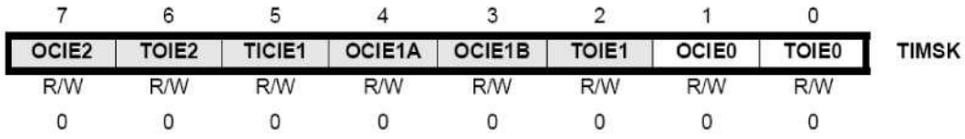
BIT [2..0] : CLOCK SELECT

مطابق جدول مقابل پالس ساعت شمارنده بیتهای CS[2..0] تعیین می‌شود در صورت استفاده از پایه TO پالس خارجی برای شمارنده به عنوان کلاک استفاده می‌شود.

CS02	CS01	CS00	Description
0	0	0	No clock source (Timer/Counter stopped).
0	0	1	clkI/O/(No prescaling)
0	1	0	clkI/O/8 (From prescaler)
0	1	1	clkI/O/64 (From prescaler)
1	0	0	clkI/O/256 (From prescaler)
1	0	1	clkI/O/1024 (From prescaler)
1	1	0	External clock source on T0 pin. Clock on falling edge.
1	1	1	External clock source on T0 pin. Clock on rising edge

جدول ۵

چون از وقفه مقایسه در مد CTC می‌خواهیم استفاده کنیم این بیت را نیز در بایت کنترلی TIMSK یک می‌کنیم



شکل ۶

OCIE0=1

در نتیجه وقفه مقایسه در سر زمانهای تنظیم شده انجام می‌شود بنابراین می‌توان مقدار متغییر شمارش CNT را در این تابع افزایش دهیم .

در صورت زیاد بودن عرض پلس ورودی ممکن است CNT به مقدار ماکزیمم خود برسد بنابراین این مقدار هر بار باید چک شود تا در صورت رسیدن به ماکزیمم دیتا ارسال شود تا از دست رفتن مقدار عرض پالس جلوگیری کنیم .

همانطور که گفته شد در حالت External clock می‌بایست وقفه‌ها نیز فعال شوند تا با تغییرات ورود های نمونه برداری را متوجه شویم .

وقفه در AVR :

AVR دارای ۳ وقفه خارجی می‌باشد که طرز کارکرد آنها نیز با توجه به رجیسترهای کنترلی تعیین می‌شود

شکل ۷ : ثبات عمومی کنترل وقفه

با یک شدن هر یک از ۳ بیت بالایی می‌توان وقفه موردنظر را فعال کرد .

بنابراین برای فعال کردن وقفه اولا می‌بایست که در GICR وقفه موردنظر فعال شود و همچنین مدکاری آن در MCUCR تنظیم شود.

7	6	5	4	3	2	1	0	
SE	SM2	SM1	SM0	ISC11	ISC10	ISC01	ISC00	MCUCR
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
0	0	0	0	0	0	0	0	

شکل۸ : ثبات کنترلی مود کاری وقفه

ISC01	ISC00	Description
0	0	در حالت صفر بودن پایه INT0وقفه اتفاق میافتد
0	1	هر تغییری در پایه INT0 باعث ایجاد وقفه می شود
1	0	لبه پایین رونده تولید وقفه میکند
1	1	لبه بالا رونده باعث تولید وقفه می شود

جدول ۶

در تابع سرویس دهی وقفه صفر می‌بایست داده ورودی جدید ارسال و بعد از آن زمان بین پالس‌ها که همان مقدار CNT می‌باشد ارسال شود و این مقدار صفر شود .

مد Storage

چنانچه در مد Real time دیدیم سرعت نمونه‌برداری در این مد وابسته به سرعت ارسال می‌باشد و چون سرعت ارسال پایینی داریم راه‌حل این کار ذخیره نمونه‌های موردنظر و بعد از آن ارسال آنها می‌باشد .

همانطور که می دانید در AVR می‌توان دستورات را با کدهای C نوشت ولی چون این دستورات تبدیل به کد اسمبلی می‌شوند و نرم‌افزار موردنظر به صورت بهینه این تغییر را اعمال نمی‌کرد و ممکن بود حجم کد موردنظر زیاد شود و بالطبع زمان اجرا بالا رود و در نتیجه سرعت نمونه‌برداری پایین بیاید بنابراین

برای این قسمت نیاز به برنامه‌نویسی اسمبلی داریم که کنترل بیشتری به اجرا داشته باشیم .که در این حالت بعد از 1024 نمونه‌گیری داده‌ها ارسال می‌گردد .

مد storage با حداکثر سرعت

برای این قسمت کد جداگانه‌ای نوشته شده است تا از تایمر استفاده نکنیم و با حداکثر سرعت نمونه‌برداری کنیم در این حالت بسته به z دو نوع نمونه‌برداری انجام می‌شود در نتیجه شرط مربوط به z در هر دوره نمونه‌برداری چک نمی‌شود.

در سرعت پایین تر از حداکثر , یک تایمر (تایمر ۲) با توجه به توضیحات داده شده فعال می‌شود و سرزمان‌های مشخص نمونه‌برداری انجام می‌دهد . و بعد از 1024 نمونه با set کردن یک flag در تابع اصلی برنامه (Main) داده‌ها ارسال می‌شوند.

مد storage با External Clock

این حالت کاملاً شبیه به Real External می‌باشد با این تفاوت که در این حالت داده‌ها به جای ارسال ذخیره می‌شوند بنابراین سرعت بالاتری دارد در این حالت نیز بعد از اتمام نمونه‌برداری یک flag ست می‌شود که تابع اصلی اتمام نمونه‌برداری را متوجه شده ارسال داده ها آغاز می‌شود.

در تمام این حالات یک بیت کنترلی برای تغییر مد در نظر گرفته شده که با دریافت کلمه کنترلی این بیت صفر می‌شود بنابراین میکرو از اجرای دستورات حالت فعلی بیرون آمده و با توجه به کلمه کنترلی جدید و شرایط دستورات جدید را اجرا می کند .

فصل دوم

USB و مدارات مربوط به آن

(USB) باس عمومی سریال یک واسط سریال با سرعت بالا و قابل انعطاف می باشد که به عنوان یکی از بهترین و راحت ترین روش های اتصال دستگاه های جانبی به کامپیوتر است.

اولین بار در سال ۱۹۹۵ توسط چند شرکت معتبر در زمینه الکترونیک و کامپیوتر از جمله USB Philips, Compaq , Hewlett < Packard invent , NEC , Intel , Microsoft بوجود آمد ، در حال حاضر این استاندارد توسط چهار شرکت Compaq, Intel, NEC, Microsoft توسعه یافته و پشتیبانی می شود.

هدف اصلی در توسعه و گسترش USB در ابتدا رسیدگی به چندین دستگاه با حداکثر سرعت ممکن بود.که در این میان مسائلی چون ، ارزان قیمت بودن ، تطابق با معماری کامپیوتر، بکارگیری راحت و قابل انعطاف بودن و همچنین افزودن تعداد زیادی از دستگاه های جانبی مدنظر قرار گرفت.

Interface	Format	Max. Devices	Max. Length (ft)	Max. Speed (bps)	Typical Use
USB	Async. Serial	127	16(up to 5)	1.5M,12M,480 M	Mouse, Keyboard
RS232	Async. Serial	2	50-100	20k (115k)	Mouse, Modem
RS485	Async. Serial	32(up to 256)	4000	10M	Data acqu. Control sys
IrDA	Async. Serial Infrared	2	6	115k	Printers, Handheld computers
Microwire	Sync. Serial	8	10	2M	µController communic.
SPI	Sync. Serial	8	10	2.1M	µController communic.
I2C	Sync. Serial	40	18	3.4M	µController communic.
IEEE-1394 (Fire Wire)	Serial	64	15	3.2G (in IEEE-1394b)	Video, mass storage
IEEE-488 (GPIB)	Parallel	15	60	8M	Instrument-ation
Ethernet	Serial	1024	1600	10M/100M/1G	Network PC
MIDI	Serial current loop	2	50	31.5k	Music, show control
Parallel Printer	Parallel	2(8 by Daisy-ch.)	10-30	8M	Printer, Scanner

جدول ۷ : جدول مقایسه پورت ها

در جدول بالا پورت های موجود نشان داده شده و با هم مقایسه شدند

خصوصیاتUSB:

۱. اتصال خارجی و استفاده آسان توسط کاربر.

۲.تشخیص دستگاه ها به صورت خودکار و اجرایی توابع راه انداز و همچنین پیکربندی آنها

۳. قابلیت اتصال ۱۲۷ دستگاه به کامپیوتر

۴.مناسب برای ارتباط با سرعت بالا معادل چند کیلو بیت تا چند مگا بیت در ثانیه

۵.پشتیبانی از دو نوع روش اتصال همزمان و غیر همزمان

۶. Hot plugقابل شناخت بدون نیاز به راه اندازی مجدد کامپیوتر

۷. دارای جریان ۵۰۰ میلی آمپر مناسب برای تغذیه

NRZI Decoding.۸

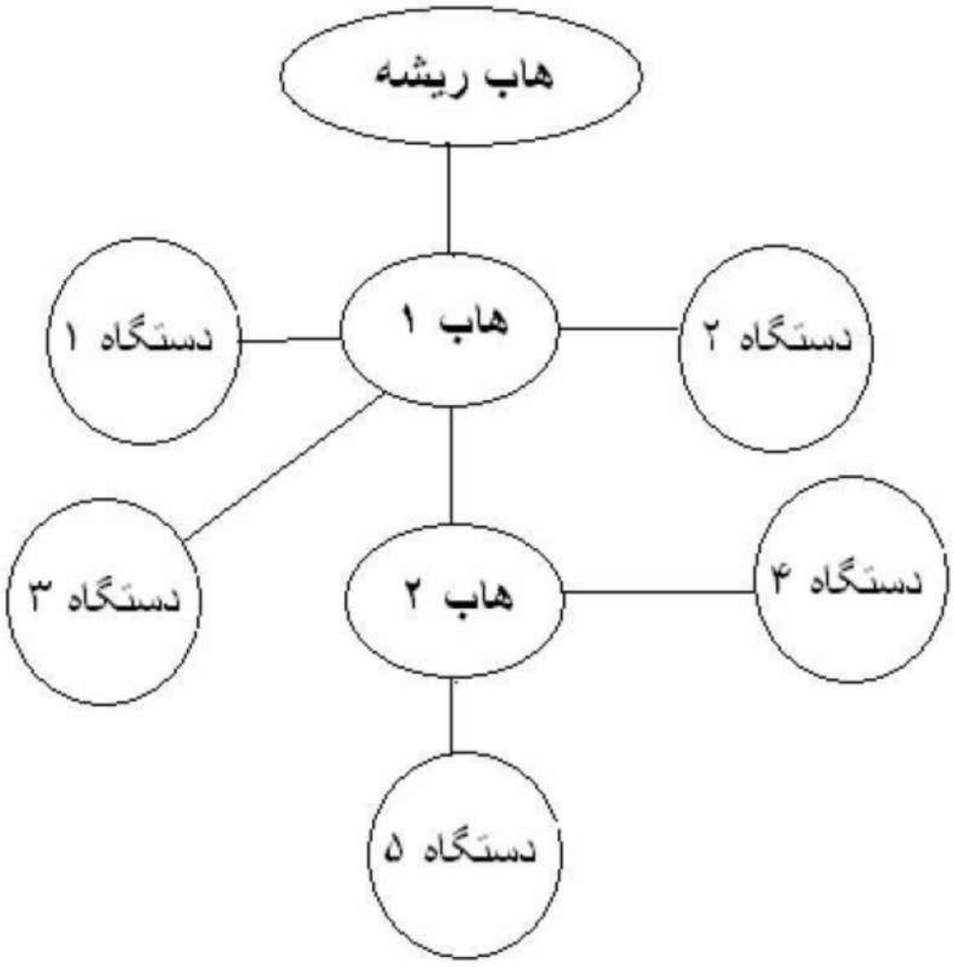
.....۹

، بطور کلی دارای ۲ نسخه Usb1.1 , Usb 2 میباشد . Usb 1.1می تواند اطلاعات را تا ۱مگابیت در ثانیه برای دستگاه های سرعت پایین و حداکثر ۱۲ مگابیت در ثانیه برای دستگاه های سرعت بالا پشتیبانی نماید ، تولید بعدی USBنسخه ۲.۰ بود که سرعت ارسال آن به ۴۸۵ مگابیت در ثانیه می رسد که سرعت آن به ۴۵ برابر سرعت نسخه ۱.۱ میرسد ولی ساختار آن کاملا منطبق با آن است.ارتباط با پورت usb از طریق یک کابل انجام می شود که از طریق آن تمام تجهیزات و دستگاههای جانبی به یکدیگر متصل می شوند این کابل امکان تبادل اطلاعات بین کامپیوتر و دستگاههای جانبی را میسر می سازد و تمامی دستگاه های جانبی از پهنای باند پورت به طور مشترک استفاده می کنند

اتصالات USB

نحوه مدل ارتباطی بین دستگاه های جانبی و میزبان در سیستم USB می باشد ارتباط فیزیکی به صورت ردیفهای ۴ ستاره ای است در مرکز هر ستاره یک هاب (HUB) وجود دارد . که خود هاب نیز دارای چند پورت است ..هنگامیکه چندین هاب پشت سر هم متصل شود می تواند چندین شاخه بوجود آورد که به هرکدام می توان یک وسیله جانبی متصل نمود .

اولین هاب موجود درسیستم USB در داخل میزبان کامپیوتر می باشد که به آن هاب ریشه می گویند .حداکثر طول قابل ارتباطی بین دو هاب را وسیله جانبی می تواند ۵ متر باشد یعنی برای اتصال ۵ هاب به هم می تواند یک دستگاه که در فاصله ۲۵ متری قرار دارد را راه اندازی نمود.



شکل ۹

دستگاه هایUSB

دستگاه های USB با توجه به شکل می تواند یکی از موارد زیر باشد

۱. هاب ها که توسعه دهنده اتصالاتUSB هستند

۲. دستگاه های جانبی که حداکثر می توانند ۱۲۷ وسیله باشد مانند پرینترها – موس – کی برد

از وظایف دستگاه های جانبی می توان به موارد زیر اشاره کرد

- تشخیص ارتباط (انتخاب شدن از طرف میزبان)
- پاسخ به درخواست های استاندارد میزبان
- بررسی خطاها
- مدیریت انرژی

میزبان USB

در سیستمUSB تنها یک میزبان وجود دارد. میزبان یک کامپیوتر است که شامل دو بخش است

۱.کنترل کننده میزبان

۲.هاب ریشه

میزبان به کمک این دو بخش و سیستم عامل می تواند با وسیله جانبی ارتباط برقرار کند .

از وظایف میزبان می توان به طور خلاصه به

۱. تشخیص دستگاه

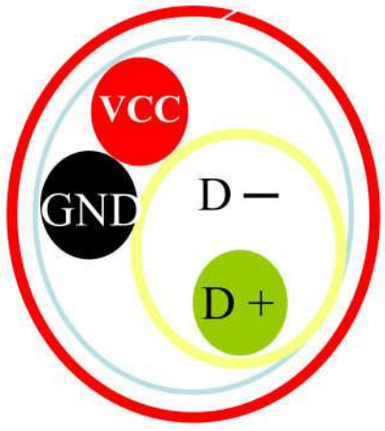
۲.نظارت بر انتقال اطلاعات

۳. بررسی خطاها

۴. تامین انرژی برای وسیله جانبی

اتصالات کابلUSB

درسیستم USBکابل ارتباطی دارای۴ سیم می باشد .



شکل ۱۰

www.elec4u.ir

۱.VCC منبع تغذیه ۵ ولت

۲.GND مرجع زمین برای- D ، +D، VCC

۳.+D، - D جفت سیم تفاضلی برای انتقال داده ها

میزبان چگونه دستگاه را می شناسد

قبل از اینکه برنامه کاربردی با دستگاه ارتباط برقرار کند ، میزبان باید دستگاه را بشناسد و راه انداز آن رابارگذاری نماید . سرشماری ، اولین تبادل اطلاعاتی است که انجام می شود.این پروسه شامل نسبت دادن یک آدرس به دستگاه ، خواندن ساختار داده ها از دستگاه ، نسبت دادن و بارگذاری کردن راه انداز دستگاه وانتخاب یک پیکربندی با استفاده از داده های بازیابی شده می باشد.بعد از طی این مرحله دستگاه پیکربندی شده و آماده انتقال داده با استفاده از اندپوینت های تعریف شده در پیکربندی خواهد بود.

پردازش

یکی از وظایف هاب ، تشخیص اتصال یا جداشدن دستگاه هاست.هر هاب یک مسیر ارتباطی وقفه ای ورودی برای گزارش این گونه رخدادها به میزبان دارد.در هنگام بالا آمدن سیستم میزبان هاب ریشه رابرای تشخیص دستگاه هایی که به آن متصل هستند جستجو می کند.این جستجو شامل هاب های خارجی ،دستگاه هایی که به آنها وصل هستند نیز می شود .پس از بالا آمدن نیز میزبان همواره آماده است که اتصال دستگاه های جدید یا جدا شدن بعضی دستگاه ها را تشخیص دهد .پس از تشخیص دستگاه جدید ، میزبان یک سری نیازمندی ها را به هاب می فرستد که باعث می شود هاب مسبر ارتباطی بین میزبان و دستگاه را برقرار کند .سپس میزبان ، توسط فرستادن واسته های کنترلی به اندپونیت پیش فرض صفر تلاش می کند تا دستگاه را سرشماری کند. همه دستگاه های استاندارد USB

باید از انتقال کنترلی ، خواسته های استاندارد و اندپونیت صفر پشتیبانی کنند.برای یک سرشماری موفق دستگاه باید با بازگرداندن اطلاعات خواسته شده و انجام عملیات مناسب به هر خواسته پاسخ دهد.از دید کاربران مرحله سرشماری باید پنهان وخودکار باشد. البته معمولا در نخستین بار استفاده ازدستگاه احتیاج است که کاربر از دیسکی که حاوی فایل INF و راه انداز دستگاه است استفاده کند.وقتی سرشماری کامل شد .ویندوز آن دستگاه جدید را به پنجره مدیریت دستگاه در کنترل پانل اضافه می کند .برای مشاهده پنجره مدیریت دستگاه در ویندوز ۹۸ مسیر زیر را طی کنید:

Start menu > Settings > Control Panel > System > Device Manager

در ویندوز ۲۰۰۰ همین مسیر را دنبال کنید با این تفاوت که فقط پس از کلیک کردن گزینه System سپس Hardware را کلیک کرده Device Manager را فشار دهید.

اگر کاربر وسیله جانبی را جداکند ، ویندوز به صورت خودکار دستگاه را از این پنجره حذف می کند .در یک وسیله جانبی عمومی ، کد برنامه دستگاه باید حاوی اطلاعاتی که میزبان در خواست ها همه مرحله (ASIC)می کند و پاسخهای مناسب به خواسته ها باشد .بعضی از تراشه های ویژه برنامه ی سرشماری را ندارند .در سمت میزبان ، تحت ویندوز احتیاجی به نوشتن کد برای سرشماری نیست، به این دلیل که ویندوز به صورت خودکار این عمل را انجام می دهد .ویندوز به دنبال فایل های متنی خاصی که INF نامیده می شوند.می گردد که وظیفه ی آنها معرفی راه انداز دستگاه می باشد

.

مد های کاری:

الف.transfer control

۱.Control Mode

ب.data transfer

۲.Bulk Mode

۳.Interrupt Mode

۴.Isochronous

Control Mode

مد کنترلی: برای تنظیمات پورت مورد استفاده قرار میگیرد. در اولین بار اتصال usb به کامپیوتر تمام دستگاهها دارای End Point 0 (مسیر انتقال) میباشند که بعد از اتصال وکامپیوتر End Point جدید را به دستگاه ارسال میکندکه بعد از ان کار تمام ارتباطات از طریق Endpoint جدید انجام می شودو همچنین در این زمان دستگاه VID , PID سرعت کاری خود را مشخص میکند که تمام این کارها در مد کنترلی انجام میگیردبعد از تکمیل تنظیمات دستگاه در مد کاری دادهای قرار میگیردو تنها زمانی که نیاز به تنظیمات جدید باشد وارد مد کنترلی می شود بنابراین برای انجام تنظیمات تمام دستگاهها نیاز به پشتیبانی از این مد را دارند این مد در تمام نسخه های USB وجود دارد شایان ذکر است که تمام دستگاه ها در شروع اتصال میبایشت دارای EndPoint0 باشند

Bulk Mode

مد توده ای : این برای انتقال داده های حجیم استفاده می شود و زمان چندان اهمیتی ندارد از جمله کاربرد ان میتوان به Mass Storage و HDD usb اشاره کرد این مد تنها در high speed , full speed قرار دارد و در این مد کنترل و تصحیح خطا نیز انجام می شود.

Interrupt Mode

مد وقفه ای : از این مد برا انتقال داده هایی با حجم پایین استفاده می شود از کاربردهای ان میتوان به Mouseو Keyboard اشاره کرد در تمام نسخه های USB وجود دارد و کنترل و تصحیح خطا نیز انجام می شود این مد ساده ترین مد برای برنامه نویسی و پیاده سازی سخت افزاری میباشد

Isochronous Mode

مد همزمان: در این مد انتقال داده ها به صورت Real Time میباشد و سرعت انتقال داده ها همیشه ثابت است و طول بلیت های ارسالی هر فریم نیز ثابت میباشد این مود برای ارسال صوت و تصویر مناسب است مانند Web Cam تشخیص و تصحیح خطا ندارد در سرعت های Full Speed و high Speed کار میکند

قبل از این زمانی که سخت افزارها به کامپیوتر متصل میشدند برای اینکه توسط کامپیوتر شناسایی شوند میبایست به همراه خود برنامه و Driver مربوطه را عرضه میکردند ولی طراحان usb امکان ویژه ای را به نام PID , VID به آن اضافه کردند که این قابلیت را به کام÷وتر میداد که قطعات متصل شده را شناسایی کند بدیل صورت که هر قطعه ساخته شده کد شناسایی خود را در ویندوز ثبت میکرد و در هنگام اتصال این کد را به کامپیوتر میفرسد که با استفاده از این کد قابل شناسایی است به این کد ها VID , PID گفته می شود

VID Vendor ID: کد شناسایی که برای هر کمپانی اختصاص دارد و با ان شرکت سازنده مشخص می شود

مثال

Atmel : 03eb

HP : 03f0

PID Product Id: کد شناسایی قطعات تولید شده ان کمپانی میباشد یعنی هر کدام از قطعات تولید شده یک شرکت دارای یک کد شناسایی می باشند

مثال

Atmel USB Mass Storage: 2002

Laser Jet 1000 HP : 0517

برنامه نویسی کامپیوتر :

Virtual Com: برای این نوغع برنامه نویسی نیاز به نصب یک درایور در کامپیوتر است که usb را تبدیل به یک پورت مجازی سریال میکند و ارتباط ما با USB به مانند پورت سریال خواهد بود در نتیجه یک نرم افزار وظیفه تبدیل داده های سریال به داده های USB را دارد . ویژگی این برنامه نویسی ساده بدن آن میباشد به طور مثال در Visual Studio به سادگی میتوان از کامپوننت Mscom استفاده کرد و حتی به صورت تحت dos این کار را انجام داد مشکل عمده این روش کمی سرعت ان میباشد که محدود به حداکثر سرعت پورت سریال یعنی ۹۲۱ کیلو بیت بر ثانیه است.

USB Driver: در این نوع برنامه نویسی به طور مستقیم با usb کار میکنیم و از توابع API ویندوز برای این کار استفاده می شود که پیچیده میباشد و برای راحتی کار یک سری DLL وجود دارد که توابعی را برای کار اماده میکند و کاربرد ان ساده است مزیت این روش استفاده از حداکثر سرعت usb می باشد.

پیاده سازی سخت افزارUSB

برای پیاده سازی USB چندین روش وجود دارد

۱.استفاده از میکرو کنترلر های عمومی

۲.استفاده از FPGA

۳.استفاده از میکروکنترلرهایی با قابلیت USB

۴.استفاده از IC های آماده برای این کار

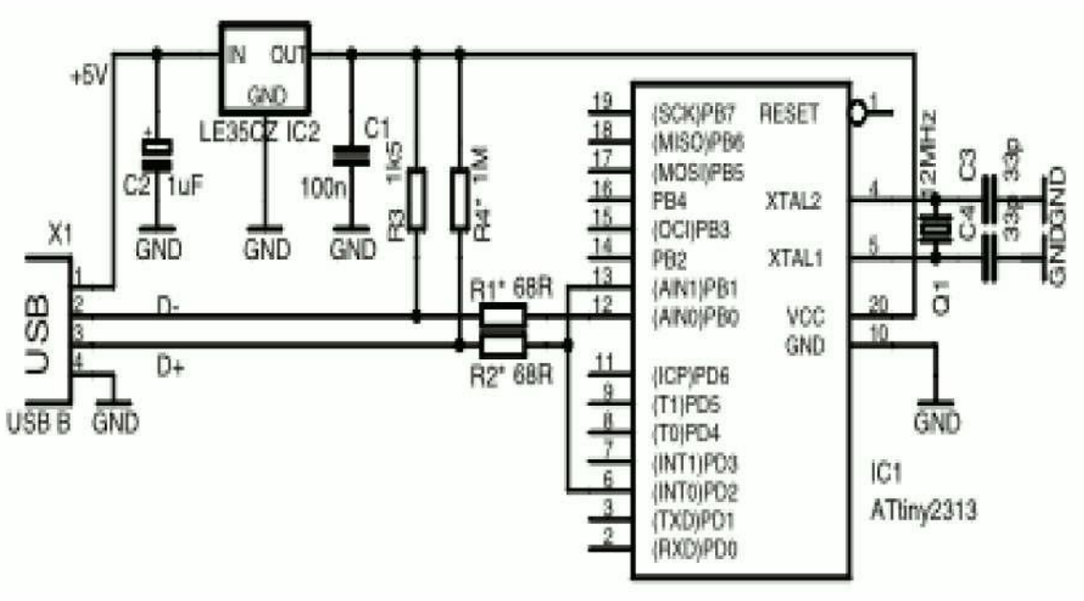
میکرو کنترلرهای عمومی

این میکرو کنترلر ها برای استفاده های عمومی میباشند و پروتکل USB را پشتیبانی نمیکنند بنابراین در صورت استفاده از این میکرو کنترلرها میبایست تمام کارهای زیر با برنامه نویسی و به صورت دستی انجام بشود

- کد گشایی و کد گذاری داده های ارسالی و دریافتی
- تنظیمات
  - داشتن Zero End Point و مد کنترلی
  - ارسال VID , PID
  - ارسال پکت ها (Packets)

در نتیجه کار کردن با این روش خیلی پیچیده و مشکل خواهد بود

در شکل زیر یک نوع ارتباط ساده با میکرو ATTINY 2313 دیده می شود



شکل ۱۱ : طریقه اتصال USB به ATTINY 2313

FPGA:

راه دیگر استفاده از FPGA ها میباشد که کلیه تنظیمات گفته شد شبیه به تنظیمات میکرو کنترلرهای عمومی میباشد با این تفاوت که:

- سخت افزا خاصی برای ورودی خروجی و وقفه خارجی ... ندارند
- سرعت کاری خیلی بالاتری دارند
- دارای پردازش موازی میباشند
- ....

USB Microcontrollers:

این میکرو ها قابلیت سخت افزاری USB را دارند و کار کردن در انها به مراتب ساده تر از ۲ نوع قبلی میباشد ولی باز هم کنترل کار USB با این واحد انجام می شود .اکثر خانواده های میکرو دارای چنین میکرو هایی هستند که در جداول زیر انواع انها با قابلیت هایشان آورده شده است

:۸۰۵۱

Device	Speed	Endpoints
AT89C51SND1	Full	2
AT89C5130	Full	7
AT89C5132	Full	4

جدول ۸

:PIC

Device	Speed	Endpoints
16C745	Full	16
16C765	Full	16
18F2x50	Full	16
18F4x50	Full	16

جدول ۹

:AVR

Device	Speed	Endpoints
AT90SUSB1286	Low-Full	7
AT90SUSB1287	Low-Full	7
AT90SUSB646	Low-Full	7
AT90SUSB647	Low-Full	7

جدول ۱۰

:AVR

Device	Speed	Endpoints	Hub port
AT43USB325	Low-Full	4	4
AT43USB326	Low-Full	3	2
AT43USB351	Low-Full	5	—
AT43USB353	Low-Full	4	2
AT43USB355	Low-Full	4	2

جدول ۱۱

:USB CHIP

این قطعات به عنوان یک راه انداز و مبدل USB به پرو تکل دیگر مانن RS232 یا پارالل.... استفاده می شوند که شرکت های زیادی دارای چنین قطعاتی میباشند از جمله:

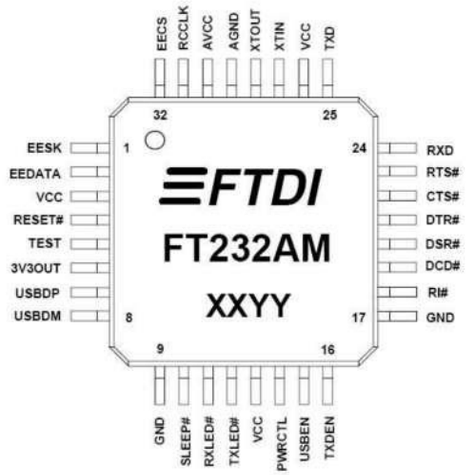
- Philips
  - PDIUSB
- FTDI Chips
  - FT232
  - FT245
  - FT2232
- Cypress Semiconductor

- Intel
- NEC Electronics
- PLX Technology
- ST Microelectronics
- Texas Instruments
- Toshiba
- ……

هر کدام از قطعات بالا دارای مزایا و معایب خاص خود هستند

USB Chipها ساده ترین کارایی را دارن زیرا نیاز به هیچ گونه تنظیمات ندارند و Usb Microcontrollerها نیز نیازی به تنظیمات ندارند با این تفاوت که سرعت پایین تری دارند و اکثر مود اینتراپت را پشتیبانی میکنند بنابراین USB Chipها مناسبترین قطعات برای کار میباشند در بین شرکت های سازنده این قطعات خانواده FTDI کارایی و راحتی بیشتری دارند و مشکلات نرم افزاری به مراتب در انها کمتر است.

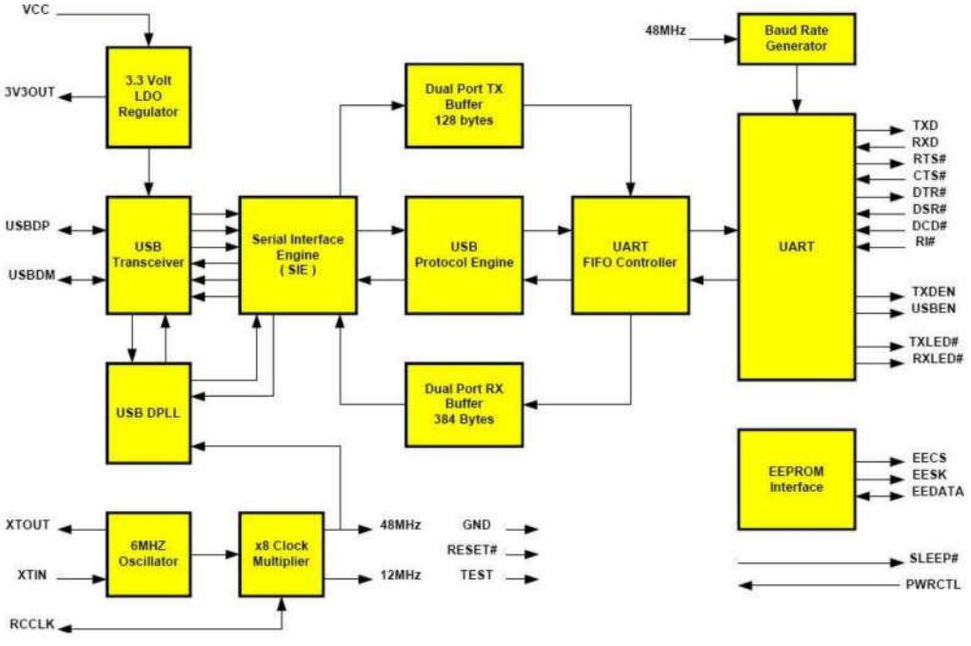
FTDI FT232:



شکل ۱۲ : Pin Out FT232

مشخصات:

- Single Chip Interface: بدون نیاز به قطعه خارجی به تنهایی برای راه اندازی USB کافی است
- پشتیبانی باود ریت (Baud Rate)بین ۳۰۰ تا ۹۲۰ کیلو برای اتصال به RS232
- حداکثر باود ریت برای RS422/RS485 مقدار ۲۰۰۰ میباشد
- ۳۸۴ بایت بافر برای دریافت و ۱۲۸ بایت بافر برای ارسال که در انتقال داده های حجیم مورد اهمیت میباشد
- دارای سیگنالهای hand shaking
- رگولاتور ولتاژ ۳.۳ ولت داخلی بدون نیاز به رگولاتور خارجی
- هماهنگ با پروتکل usb نسخه ۱.۱
- دارای VIDو PID وشماره سریال ذخیره شده در EEPROM خارجی قابل تغییر
- virtual com port driver for windows



شکل ۱۳: مدار داخلی FT232

3.3Volt LDO Regulator:
وظیفه تولید ولتاژ ۳.۳ ولت برای واحد USB Transceiver و بافر های خروجی را بر عهده دارد و برای کار نیاز یه یک خازن Decupling خارجی برای اتصال به پایه 3v3 out دارد

USB Transceiver:
وظیفه ایجاد یک رابط فیزیکی به کابل USB را دارد و هماهنگ با usb 1.1 full speed است.
USB DPLL : وظیفه تولید کلاک از روی ورودی NRZIبرای ماژول SIEرا دارد

6MHz Oscillator:برای تولید کلاک ۶MHZ برای واحد x8 میباشد که فرکانس این قسمت از یک اسیلاتور خارجی تامین می شود

X8 Clock Multiplier: با استفاده از یک ضرب کننده از کلاک 6MHZ تولید کلاک های ۱۲و ۴۸ را برای واحد های داخلی انجام میدهد

serial Interface Engine (SIE) : تبدیل کننده سریال به پارالل و پارالل به سریال و تولید CRC ها بر عهده این قسمت میباشد

USB Protocol Engine: این قسمت به صورت سطح پایین برای کنترل usb تولید Endpoint را بر عهده دارد

Dual Port TX Buffer (128 bytes): یک حافظه Dual Port میباشد که داده هایی که از usb میایند در ان ذخیره شده و از طرف دیگر به وسیله واحد UART از ان خارج میشوند.

Dual Port RX Buffer (384 bytes): یک حافظه Dual port میباشد که داده های ورودی از Uart در آن ذخیره شده و با درخواست واحد SIE از آن خارج میشوند.

UART FIFO Controller : این واحد وظیفه کنترل دو حافظه TX , RX که در بالا اشاره شد را بر عهده دارد .

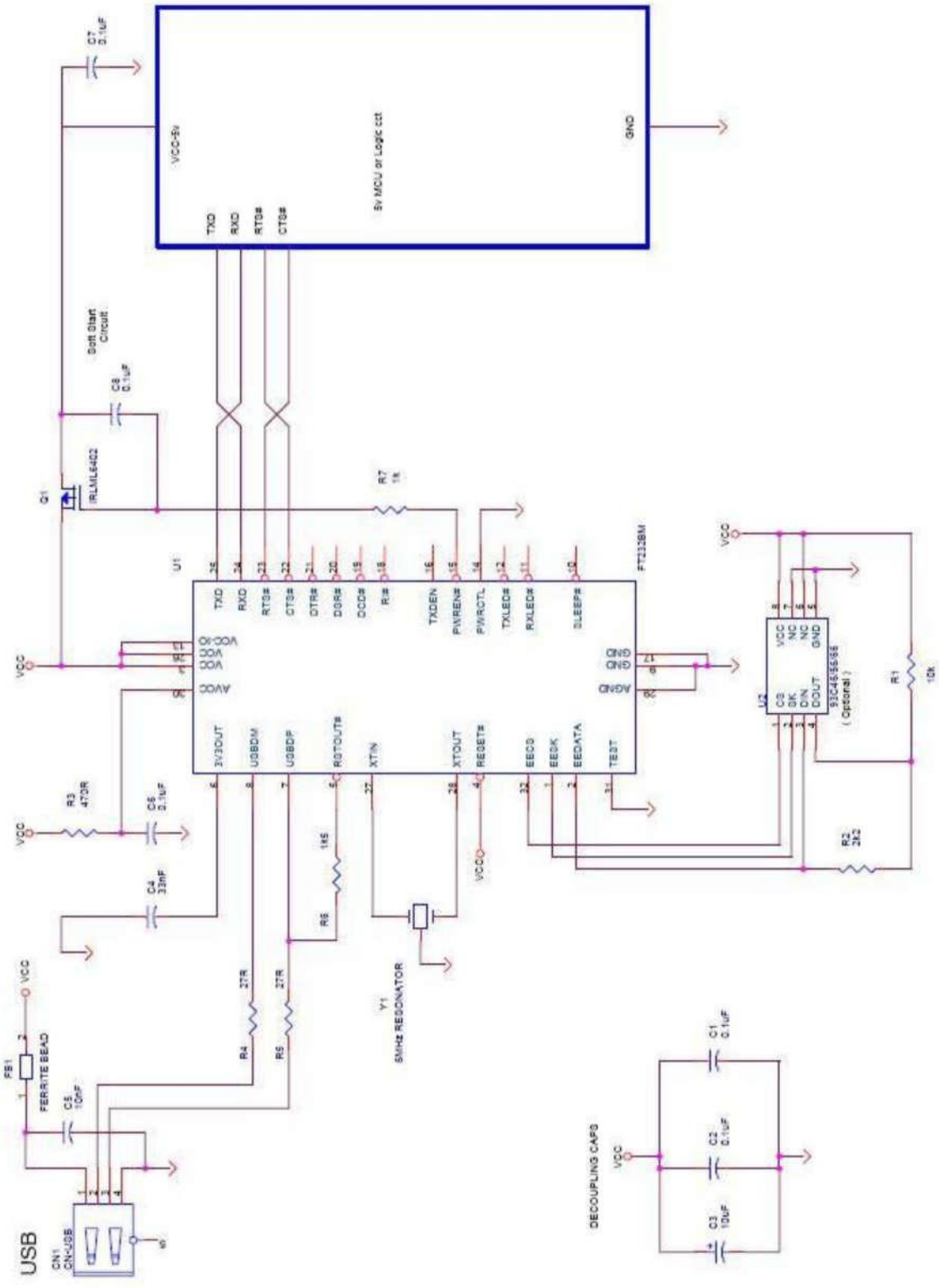
UART: وظیفه این بخش تولید اده های سریال ۸/۷ بیتی از خروجی پارالل و تبدیل داده های ورودی به پارالل را بر عهده دارد در کل این واحد وظیفه تولید داده با پروتکل سریال (Rs232/Rs485) را دارد و همچنین این بخش قابلیت تولید سیگنالهای Handshaking را نیز دارد .

Baud Rate Generator: این بخش با استفاده از یک تقسیم کننده ۱۴ بیتی وظیفه تولید باوود ریت های مختلف از کلاک 48MHZ را بر عهده دارد.

EEPORM: این Ic(FT232) قابلیت کار کردن بدون EEPROM خارجی دارد ولی میتوان از یک EEPROM خارجی مثل C46۹۳ برای ذخیره سازی PID , VID , SERIAL Number استفاده کرد وظیفه این بخش کنترل EEPROM میباشد.

شماره پایه	نام پایه	جهت	توضیحات
7	USBDP	I/O	+D که میبایست با یک مقاومت1.5 کیلو اهم به ولتاژ 3.3 وصل شود
8	USBDM	I/O	D-
6	3V3OUT	OUT	خروجی 3.3 ولت رگولاتور داخلی
27	XTIN	IN	ورودی کریستال 6MHZ
28	XTOUT	OUT	خروجی واحد تولید کننده 6MHZ
31	RCCLK	I/O	RC timer – used to guarantee clock stability on exiting sleep mode
4	RESET#	IN	پایه ریست مدار
32	EECS	I/O	پایه chip Select برای EEPROM خارجی
1	EESK	I/O	پایه CLOCK برای EEPROM خارجی
2	EEDATA	I/O	DATA I/O برای EEPROM خارجی
5	TEST	IN	Puts device in i.c. test mode – must be tied to GND
25	TXD	OUT	UART – Transmit Data Output
24	RXD	IN	UART – Receive Data Input
23	RTS#	OUT	UART – Request To Send Control Output
22	CTS#	IN	UART – Clear To Send Control Input
21	DTR#	OUT	UART – Data Terminal Ready Control Output
20	DSR#	IN	UART – Data Set Ready Control Input
19	DCD#	IN	UART – Data Carrier Detect Control Input
18	RI#	IN	UART – Ring Indicator Control Input
16	TXDEN	OUT	UART – Enable Transmit Data for RS485
15	USBEN	OUT	نشان دهنده فعال بودن usb که بعد از انجام تنظیمات 1 می شود
14	PWRCTL	IN	پایه مشخص کنند نوع تغذیه0 تغذیه توسط usb و1 برای تغذیه خارجی
12	TXLED#	O.C.	پایه درایو دیود نوری که در موقع ارسال داده از usb مقدار 0 مگیرد
11	RXLED#	O.C.	پایه درایو دیود نوری که در موقع دریافت داده توسطusb مقدار 0 مگیرد
10	SLEEP#	OUT	یک پاییه خروجی که در موقع بیکاری usb مقدار 0 به خود میگیرد
3,13,26	VCC	PWR	پایه تغذیه 4.4 ولت تا5.25 ولت
9.17	GND	PWR	پایه اتصال به زمین
30	AVCC	PWR	Device - Analog Power Supply for the internal x8 clock multiplier
29	AGND	PWR	Device - Analog Ground Supply for the internal x8 clock multiplier

جدول ۱۲ شرح کار پایه های FT232



شکل ۱۳ : مدار راه اندازی FT232

فصل سوم

نرم افزار کامپیوتری

یکی از اصلی ترین قسمت های لاجیک آنالایز بخش نمایش و ذخیره سازی آن میباشد که در این پروژه این قسمت توسط کامپیوتر انجام می شود. بنابراین در کامپیوتر به یک برنامه که وظیفه آن ارتباط با سخت افزار و دریافت اطلاعات و نمایش آنها است نیاز داریم.زبان های برنامه نویسی زیاده برای این کار وجود دارند که شامل ۲ دسته بزرگ می شوند.

۱.زبان های تحت DOS

- C++
- Pascal
- Basic

که در بین این زبان ها C++ یکی از قدرتمندترین آنها میباشد ولی ضعف این زبانها راحت نبودن راحت کاربری میباشد

۲.زبانهای ویژوال

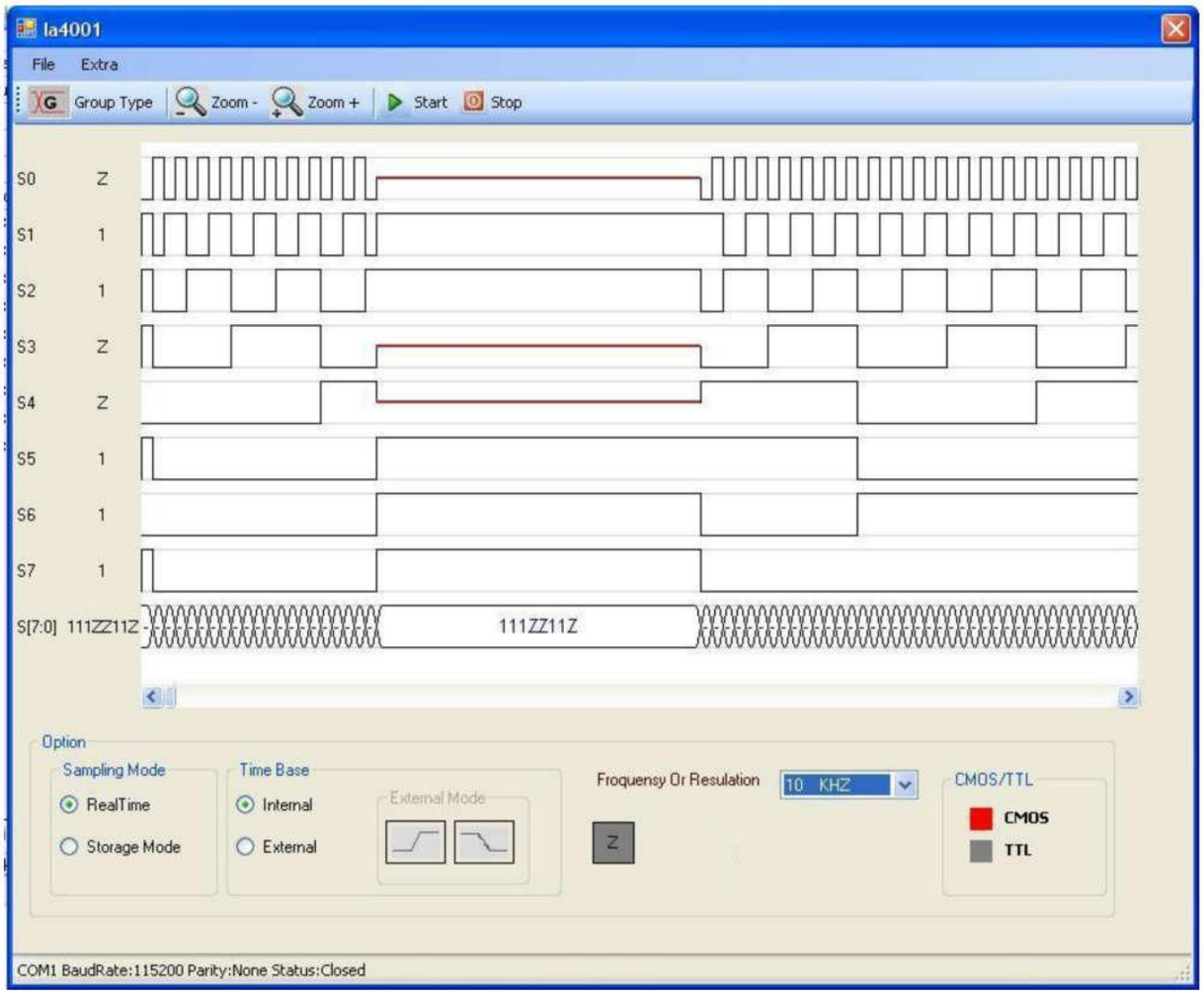
- Visual Basic
- Visual C++
- Visual C#.Net

این زبانها دارای رابط کاربری خیلی خوب میباشند و قسمت طرلحی انها قابلیت ایجاد پنجره ها و محیط کابر پسند تری به ما میدهد.

در بین این زبانها امروزه Visual C#.net از محبوبیت و قدرت زیادی برخوردار است که دارای Syntax زبان C میباشد وکارکردن با آن به مراتب ساده تر از Visual C++ میباشد.بنابراین از این زبان برای نوشتن نرم افزار استفاده شده است.

قابلیت های برنامه

- قابلیت کارکرد در ۲ مد کاری Real Time , Storage
- دارای Zoom In ,Zoom Out
- امکان ذخیره سازی نمونه ها
- توانایی انتخاب پورت کاری
- نمایش حالت انتخاب CMOS /TTL
- انتخاب و یا عدم انتخاب تشخیص Z
- Signal Grouping برای دسته بندی سیگنالها و نمایش در انواع BCD, Binary , HEX
- قابلیت Trace



شکل ۱۴: شمای کلی برنامه

در شکل ۱۴ پنجره اصلی برنامه دیده می شود که دارای تنظیمات کلی می باشد

Sample Mode : این بخش برای انتخاب حالت نمونه برداری می باشد که قابلیت انتخاب مد کاری نمونه برداری را به ما میدهد.

Time Base: برای انتخاب مرجع کلاک به کار میرود و همچنین در مد External میتوان Falling Edge, Rising Edge را انتخاب کرد.

CMOS /TTL: با ۲ نشان گر حالت انتخاب شده بیرونی را به ما نمایش میدهد

Frequency: یک لیست میباشد که در آن فرکانسهای کاری برای انتخاب وجود دارد

Z : این کلید برای انتخاب حالت تشخیص و عدم تشخیص Z به کار میرود

لازم به ذکر است که راهنمای برنامه هم برای ان وجود دارد که قسمت های مختلف را توضیح میدهد

بعد از انتخاب تنظیمات موقع شروع به کار بایت کنترلی توسط برنامه تولید و به سخت افزار ارسال می شودو بعد از نسبت به نوع تنظیمات برنامه منتظر دریافت داده ها می شود و ان هارا نمایش میدهد

فصل چهارم

طرح های پیشنهادی

یکی از مشکلات طرح کنونی سرعت پایین آن بود که عوامل مختلفی باعِث این عیب میشوند

۱.کمی سرعت رابط USB استفاده شده در طراحی

در طراحی این مدار چنانچه توضیح داده شد از FT232 استفاده شده است که ۲ مشکل عمده دارد

I. سرعت پایین: این IC دارای سرعت کاری نسبتا پایین میباشد . سرعت کاری این IC در حدود 1Mb/sec میباشد

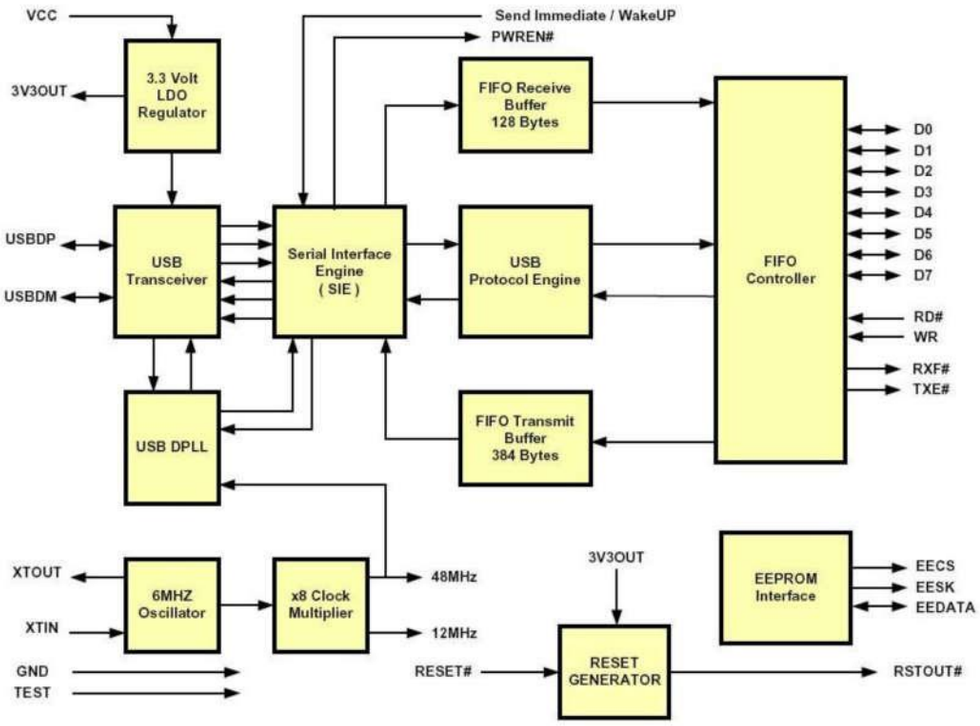
II. خروجی به صورت سریال

از اشکالات این IC انی است که خروجی ان به صورت سریال میباشد و بنابراین به خاطر محدودیت های پروتکل سریا نمیتوان به باوود ریت های بالا دست یافت .یک مشکل مهم دیگر در خروجی سریال سختی کار با آن میباشد. چون زمانیکه ما بخواهیم به جای میکرو از FPGA استفاده کنیم نیاز به پیاده سازی پروتکل سریال خواهیم داشت که مشکلات زیادی دارد ولی کار کردن با پاراللل به مراتب آسان تر از سریال میباشد

برای رفع محدودیت ها و مشکلات بالا میتوان از FT245 استفاده کرد این قطعه ساخهمان شزکت میباشد با این تفاوت که محدودیت های قطعه بالا را ندارد یعنی هم سرعت کاری بالاتر و هم خروجی پاراللل را دارا میباشد. بنابراین کارکرد با آن به مراتب ساده تر میباشد

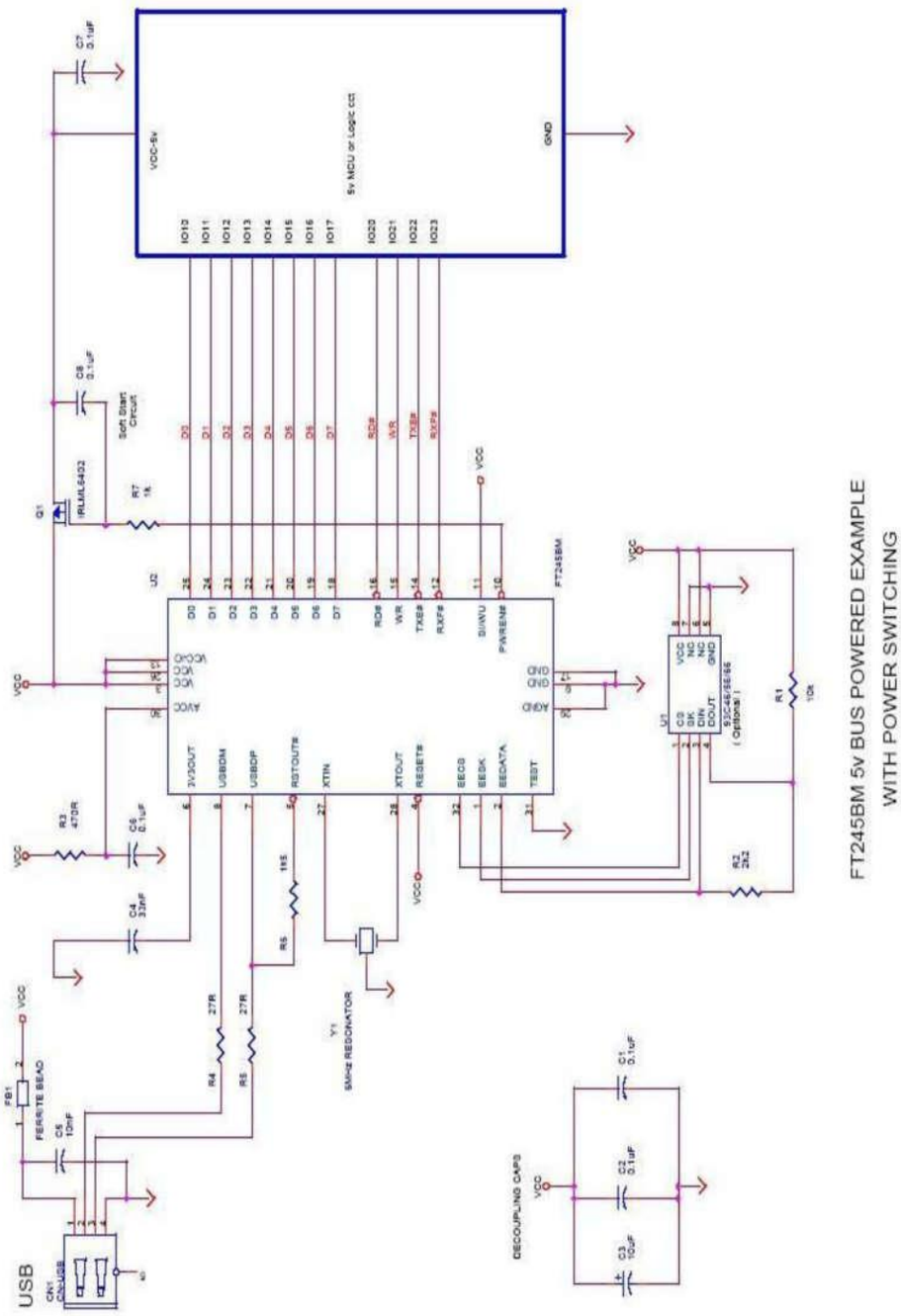
مشخصات کلی FT245

- این قطعه به تنهایی برای راه اندازی رابط USB استفاده می شود و نیازی به قطعات جانبی ندارد و خطوط داده آن به صورت ۲ طرفه عمل میکند یعنی هم قابلیت ارسال و هم دریافت داده را از یک باس داده دارد
- سرعت انتقال داده ها تا 1Mbyte/Sec
- ۳۸۴ بایت بافر FIFO برا ارسال و ۱۲۸ بایت بافر FIFO برای دریافت داده که در حجم بالای داده کمک زیادی میکند
- خطوط داده به صورت پارالل میباشد



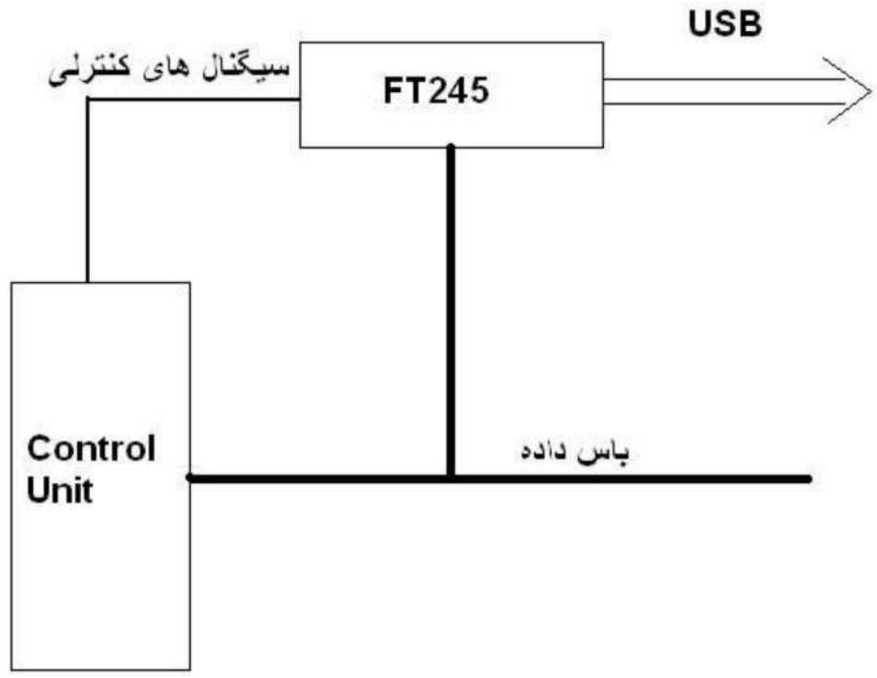
شکل ۱۵ : مدار داخلی FT245

واحد USB Transceiver وظیفه پکت بندی و کد گذاری و کدگشایی داده های ارسالی و دریافتی از خط USB را بر عهده دارد..واحد FIFO Controller وظیفه تبدیل بین سریال و پارالل را با ذخیره موقت بر عهده دارد.. ارسال داده هی کنترلی به پورت USB از طریق Zero End Point که شامل ارسال VIDو PID وتنظیم سرعت انتقال و دریافت شماره END Point ارسال و دریافت توسط واحد USB Protocol Engine انجام میپذیرد.



شکل ۱۶ : مدار راه اندازی FT245

مدار شکل قبلی مدار راه اندازی FT245 میباشد که طریقه اتصال آن با یک واحد کنترلی نشان داده شده است و راه اندازی ان ساده میباشد و تنها واحدی که باید به ان اضافه شود یک EEPRom میباشد که در ان مشخصات ماژول USB در ان ذخیره می شود.



شکل ۱۷ : کاربرد FT245 در حالت دسترسی مستقیم

از مزایای استفاده از FT245 میتوان به کاربرد ان در مد Real Time اشاره کرد به دلیل وجود باس دیتا ۸ بیتی میتوان سیگنالهای ورودی(نمونه برداری) را به صورت مستقیم به ان وصل کرد و تنها کنترل ان با استفاده از میکرو انجام بپذیرد در نتیجه در این حالت داده ها بون گذر از میکرو فرستاده میشوند در نتیجه زمانی که که داده ها از میکرو میگذرند صرفه جویی می شود بنابراین میتوانیم از حداکثر سرعت انتقال استفاده کنیم.

۲.کمی سرعت واحد کنترلی

از عوامل دیگر در کاهش سرعت نمونه برداری سرعت پایین ATMEGA32 میباشد که برای رفع این مشکل میتوان به جای آن از FPGA یا میکرو با سرعت بالا تر استفاده کرد.استفاده از FPGA مزایای زیادی برای افزایش سرعت دارد زیرا FPGA ها در داخل خود از gate logic استفاده میکنند بنابراین تنها زمان صرف شده برای تاخیر gate ها میباشد که خیلی سریعتر از میکرو میباشد در ثانی میتوان از روشهای پالسی برای افزایش سرعت استفاده کرد به طور مثال اگر یک FPGA با کریستال 25MHZ کار کند میتوان با نمونه گیری در ۲ لبه پالس سرعت نمونه برداری را به دو برابر افزایش داد و یا با ایجاد تاخی در کلاک به صورت موازی از چند لبه در یک پریود برای نمونه برداری استفاده کرد.

همانطور که قبلا توضیح داده شد برای ارسال داده ها با FT245 میتوانستیم به صورت مستقیم عمل کنیم یعنی داده ها بدون گذر از واحد کنترلی ارسال بشوندو واحد کنترلی تنها وظیفه کنترل سیگنالهای FT245 را به عهده داشته باشد ولی در مد storage نیز با قرار دادن ram خارجی میتوان بدون گذر اطلاعات از میکرو یا FPGA مستقیما داده ها به RAM اعمال شوند در نتیجه بدون اتلاف وقت در گذر از میکرو مستقیما در Ram ذخیره میشوند و میکرو تنها وظیفه کنترل پایه R/W رم و پایه های آدرس را به عهده میگیرد و حتی در این روش میتوان با گذاشت یک سخت افزار خارجی به عنوان شمارنده آدرس ram را هم مستقیمااعمال کرد در این حالت زمان خیلی کمی از دست میرود و عاملی که در سرعت نمونه برداری تاثیر دارد زمان دسترسی (Access Time) حافظه میباشد شایان ذکر است که روش کاری در این حالت شبیه به مد کاری DMA میباشد با این تفاوت که سیگنالهای Handshaking وجود ندارد بنابراین در برنامه نویسی این قسمت حتما باید به زمان دسترسی Ram توجه کنیم تا داده ها از بین نروند.

روشهای نرم افزاری:

چنانچه توضیح داده شد روشهای بالا برای افزایش سرعت سخت افزاری بودند و این تا حدی به ما کمک میکند بنابراین میتوان با الگوریتم های خوب سرعت را بالا برد.

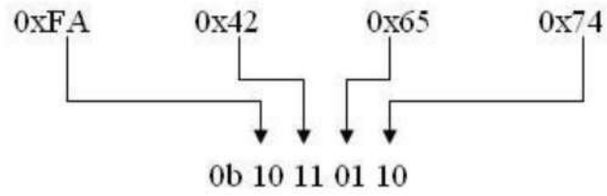
از راه های دیگر، فشرده سازی اطلاعات در هنگام ارسال میباشد .

استفاده بهینه از پهنای باند : با فرض اینکه دستگاه ما n بیتی کار میکند ولی ما برای نمایش نیاز به تعداد بیتهای کمتری داشته باشیم میتوان از این روش استفاده کرد زیرا در هر ارسال تعداد بیتهای اضافی وجود دارد که لزومی به ارسال ندارند بنابرین میتوان با ترکیب چند نمونه ذر چند سیکل کاری نمونه ها فرستاده شوند مثلا با فرض ۸ بیتی بودن دستگاه اگر ما نیاز به نمایش ۴ سیگنال داشته باشیم در هر ارسال ۴ بیت اضافی ارشال میشوند بنابراین میتوان با قرار دادن ۴ بیت نمونه برداری اول در ۴ بیت کم ارزش و ۴ بیت سیکل بعدی در ۴ بیت با ارزش در هر ۲ سیکل داده ها ارسال شوند بنابراین سرعت بالا میرود یعنی از حداکثر پهنای باند استفاده یمشود زیرا چنانچه در فصل های قبلی گفته شد سرعت کار به سرعت ارسال (در مد Real Time) بستگی دارد.

یک روش دیگر در حالتی است که در یک دوره خاص نرخ تغییرات خیلی کم میباشد بنابراین ما نیاز به ارسال مدائم داده ها نداریم و میتوانیم در زمان رخ دادن تغییر داده را به همراه مدت زمان تغییرات ارسال کنیم ودر این مدت توسط یک شمارنده میزان عرض بین تغییرات شمارش می شود ولی باید دقت کنیم که اگر بتوانیم فرکانس شمارش را نزدیک به فرکانس تغییرات کنیم عدد های شمارش شده خیلی کوچک خواهند بود بنابراین اعداد ۲ یا ۳ بیتی خواهیم داشت که میتوانعرض تغییرات چند نونه برداری را باهم فرستاد به طور مثال اگر داشته باشیم

داده ها : 0x74	0x65	0x42	0xFA
تغییرات:0x02	0x01	0x03	0x02

همانطور که میبینید برای ارسال این داده ها نیاز به ارسال ۸ بایت میباشد یعنی تعداد ۰ زیادی که لزومی برای ارسال ندارن فرستاده می شود ولی اگر از روش مذکور استفاده کنیم داریم .



با توجه به شکل بالا مشاهده میکنیم که برای ارسال نیاز به ۵ بایت داریم یعنی سرعت ارسال و حجم ذخیره سازی کاهش می یابد. یعنی با طرح الگوریتمی همیشه فرکانس شمارش را نزدیک فرکانس تغییرات نگه داریم.

علی خاکپراقی