# بنام خدا

نوشتار حاضر توضيح و گزارشي است راجع به ساخت وسيله‌اي كه قادر به ضبط صدا روي رايانه و تبديل آن فرمتي قابل درك براي سيستم عامل است و در واقع مي‌توان آن را به نوعي مدار واسطي با استفاده از اسلات ISA كامپيوتر دانست.

اين پروژه در واقع شامل سه بخش طراحي و ساخت بخش آنالوگ، طراحي و ساخت سخت‌افزار ديجيتال براي ارتباط با كامپيوتر و طراحي و اجراي نرم‌افزاري براي شناساندن به رايانه است.

به عقيده اينجانب مهمترين هدف اين كار را مي‌توان گامي براي درك نحوه ارتباط رايانه با محيط خارج دانست، كاري كه امروزه مي‌تواند يكي از مهمترين اصول و نيازهاي اتوماسيون باشد. زيرا با توجه به اينكه رايانه‌هاي شخصي اكنون وسايلي در دسترس و ارزان قيمت هستند، بهترين و ارزانترين راه اتوماسيون ارتباط دادن وسايل گران ديروزي به رايانه ارزان امروزي است كه با توجه به قابليت

برنامه‌پذيري فوق‌العاده بالا ما را قادر مي‌سازد با هزينه‌اي بسيار كم امكاناتي فوق‌العاده زياد را به وسايل خود اضافه نماييم و علاوه بر اينكه كاري كه سخت‌افزار پيچيده و گران قيمت انجام مي‌داد را نيز رايانه با نرم‌افزار به نحوي بهتر و دقيقتر انجام خواهد داد. به عنوان نمادي از اين روند در اين پروژه بر خلاف كليه كارتهاي صوتي موجود در بازار كه روي برد خود تراشه‌اي به عنوان DSP دارند، با استفاده از يك نرم افزار كار پردازش سيگنال ديجيتال انجام مي‌شود.

توضيحات خود را نخست با توضيح كامل اسلات ISA كه مهمترين بخش پروژه است آغاز مي‌نماييم. سپس توضيحاتي راجع به كارتهاي صوت موجود در بازار و در نهايت راجع به نحوه كار مدار آنالوگ، نحوه ارتباط با رايانه و چگونگي عملكرد نرم‌افزار ادامه مي‌دهيم. در اين قسمت لازم است از زحمات و راهنماييهاي ارزشمند استاد راهنما جناب آقاي دكتر شريفي قدرداني

كرده و براي ايشان آرزوي موفقيت در فعاليتهاي علمي و تحقيقاتيشان را داشته باشم. همچنين از جناب آقاي مهندس وكيلي به عنوان مشاور سختافزاري و جناب آقاي مهندس محمد حسن بستاني به عنوان مشاور نرم افزاري نهايت امتنان را دارم.

# مقدمه اي بر باس ISA

همراه با پيشرفت سيستم هاي كامپيوتري و ظهور CPU هاي قويتر، باسهاي ارتباطي اجزاء كامپيوتري نيز، دچار تغيير و تحول شده اند. باس اولين كامپيوترهاي IBM ، باس XT ي ٨ بيتي بود. با ظهور CPU هاي ١٦ بيتي اين باس جاي خود را به باس AT يا ISA ي ١٦ بيتي با فركانس كاري ٨ مگاهرتز داد. ظهور CPU هاي ٣٢ بيتي و كاربردهاي سريع گرافيكي از يك طرف و مشكلات باس ISA از طرف ديگر، سازندگان كامپيوتر را بر آن داشت كه به فكر ايجاد يك باس جديد و سريع باشند. بدين ترتيب باسهايي نظير IBM Micro Channel و EISA معرفي شدند كه ٣٢ بيتي بودند. اين باسها داراي سرعت بيشتري نسبت به ISA بودند و بسياري از مشكلات آن را برطرف كرده بودند ولي باز داراي مشكلاتي بودند. مثلا IBM Micro Channel با ISA سازگار نبود و EISA داراي سازگاري الكترومغناطيسي خوبي نبود.

براي افزايش سرعت مخصوصا براي كارتهاي گرافيكي يك روش اين است كه به جاي اينكه كارتها از طريق

اسلاتهاي توسعه نظير ISA به كامپيوتر وصل شوند بطور مستقيم به باس محلي كامپيوتر وصل گردند و بدين ترتيب چندين باس محلي بوجود آمد كه از جمله مهمترين آنها ميتوان به باس VESA يا VLBUS اشاره نمود. بوسيله اين باس ميتوان حداكثر ۳ كارت را به باس محلي CPU وصل نمود.

با روي كار آمدن پردازنده پنتيوم و مشكلات موجود در گذرگاههاي قبلي، شركت اينتل به فكر طراحي يك باس استاندارد با سرعت و قدرت بالا افتاد. بدين ترتيب باس PCI معرفي گرديد كه براي دسترسي به اجزاي جانبي با همان سرعت باس محلي طراحي شده است.

باس محلي CPU به دو باس به اسم front side bus و backside bus تقسيم شده است.باس backside يك كانال سريع و مستقيم بين CPU و حافظه كش (مرتبه دوم) را فراهم ميكند.باس frontside از يك طرف حافظه سيستم را از طريق كنترلر حافظه به CPU وصل ميكند و از طرف ديگر باسهاي كامپيوتر نظير PCI ، ISA و ... را به CPU و حافظه سيستم وصل مينمايد.در واقع اين كار باعث گرديده است كه وقتي CPU با حافظه كش كار

مي‌كند، وسايل جانبي ديگر بتوانند به حافظه سيستم دسترسي پيدا كنند.

در اين پروژه سعي شده باس ISA به طور كامل مورد بررسي قرار گيرد كه به ترتيب مطالب فصول ۱و ۲ را تشكيل مي دهند. در اين فصول به طور مفصل مشخصات الكترونيكي اين باسها و نحوه ارتباط آنها با CPU بيان شده . اميد كه اين پروژه بتواند در تفهيم مطالب مذكور مفيد فايده قرار گيرد.

.

# *ISA BUS*

**باس ISA (Industry Standard Arehitecture)**

باس ISA كه برخي به آن باس AT نيز مي‌گويند داراي مشخصات زير مي‌باشد:

١- ١٦ بيت باس ديتا

٢- ٢٤ بيت باس آدرس

٣- ١١ خط وقفه (IRQ2-ERQ7، IRQ14-IRQ15، IRQ10-IRQ12)

٤- ٧ كانال DMA

٥- ماكزيمم فركانس باس برابر ٨/٣٣ مگاهرتز

٦- سيكل‌هاي باس بدون Wait state را حمايت مي‌كند

٧- حمايت از masterهاي alternate

٨- انتقال داده به صورت سنكرون است و Muster هيچ سركشي از Slave به عمل نمي‌آورد. بلكه Master و Slave خود را با كلاك سيستم سنكرون مي‌كنند. ماكزيمم انتقال داده برابر است با:

$$8/33^{MHZ} * \frac{2Bytes}{2Clockcycle} = 8/33 MB/S$$

**محدوديتهاي ISA**

۱- باس ديتاي آن ١٦ بيتي است و نمي‌تواند باس ديتاي ۳۲ و ٦٤ بيتي پردازنده‌هاي پنتيوم را حمايت كند.

۲- باس آدرس آن ٢٤ بيتي است و مي‌تواند ١٦MB حافظه را آدرس كند و قادر نيست باس آدرس ۳۲ بيتي (٤GB) پردازنده‌هاي پنتيوم را حمايت كند.

۳- شيارهاي گسترش باس ISA بزرگ بوده و علاوه بر اينكه جاي زيادي را مي‌گيرد به دليل افزايش اثرات فازي و القايي فركانس باس به ۸/۳۳ مگاهرتز محدود مي‌گردد. يعني CPU كه با فركانسهاي بالا نظير ٥٠ مگاهرتز كار مي‌كند هنگام كار با ISA با نرخ ٥٨/۳۳ مگاهرتز تبادل داده مي‌كند. به علت كم بودن پايه‌هاي زمين اثرات تابش فركانس راديويي و اثرات Crosstalk كاهش نيافته و ISA از نظر اجرايي دچار مشكل مي‌گردد.

٤- چون وقفه‌ها (IRQها) حساس به لبه‌اند، به هر يك فقط يك وسيله مي‌تواند اختصاص پيدا كند. و دو يا چند وسيله نمي‌توانند از يك پايه وقفه مشترك استفاده نمايد. در سيستم‌هاي فركانس بالا، وقفه حساس به لبه، به دليل نويز در ورودي IRQ، امكان فعال شدن غلط وجود دارد.

٥- در كامپيوترهاي قديمي PC/XT ٤ كانال DMA ٨ بيتي وجود داشت كه كانال ٠ براي Refresh حافظه‌هاي DRAM بكار مي‌رود. كانالهاي ٣-١ بعنوان DMA براي انتقال داده بكار مي‌روند.

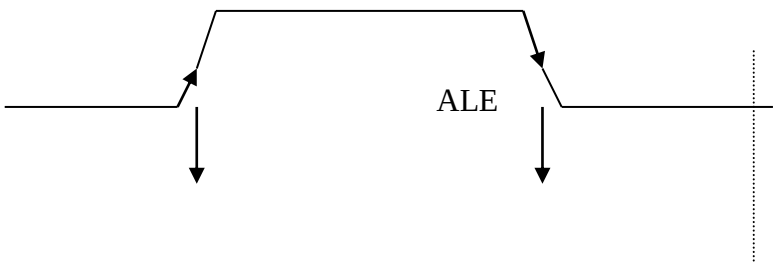
در كامپيوترهاي جديد PC/AT، كانال ٠ وظيفه Refresh حافظه‌هاي DRAM را بر عهده ندارد و بجاي آن يك مدار Refresh اين كار را انجام مي‌دهد. بنابراين كانال ٠ نيز مي‌تواند مانند بقيه كانالها براي انتقال داده استفاده شود. در كامپيوترهاي PC/AT، ٣ كانال DMA، ١٦ بيتي اضافه شده است. پس در مجموع ٧ كانال DAM وجود دارد كه كانالهاي ٥ الي ٣، ٨ بيتي و كانالهاي ٤ الي ٧، ١٦ بيتي هستند. مشكلي كه وجود دارد انستكه كانالهاي DMA ١٦ بيتي تنها قادر به انتقال داده از آدرس‌هاي زوج هستند ولي

DOS داده را از آدرس فرد يا زوج به حافظه RAM منتقل مي‌نمايد و با اين كار سازگار نيست. بنابراين عمليات انتقال بجاي DMA از طريق CPU انجام مي‌گيرد.

**سيگنالهاي گذرگاه ISA :**

**خطوط آدرس A0-A19**

A0-A19 (كه به آن SA0-SA19 نيز مي‌گويند) جهت دستيابي به حافظه و I/Oها مورد استفاده قرار مي‌گيرند. چون سرعت CPU زياد است و ممكن است چپهاي جانبي با اين سرعت كار نكنند و قبل از برداشتن آدرس توسط وسايل جانبي آدرس نامعتبر گردد. بنابراين آدرس را latch مي‌كنيم (مثلاً توسط 74373). اين كار توسط سيگنال ALE انجام مي‌گيرد. تراشه Latch توسط لبه بالا رونده ALE فعال مي‌شود و خطوط آدرس در لبه پايين رونده ALE در داخل Latch قرار مي‌گيرند. اين كار در درون PC انجام مي‌شود و خطوط فوق كه در Slot موجود مي‌باشند Latch شده هستند و در طول سيكل خواندن يا نوشتن ثابت مي‌مانند.



Time to latch Address Lnvalid

Address Valid

**شكل (۱-۱)**

براي وسايل I/O فقط پايه‌هاي A0-A15 استفاده مي‌شود و خطوط وزن بالا براي كار با حافظه مي‌باشند.

**: (Address Latch Enable) ALE**

اين سيگنال براي ايجاد اطلاعات زماني براي latch كردن آدرس بكار مي‌رود. لبه بالارونده اين سيگنال وجود آدرس معتبر را روي پايه‌هاي A0-A19 نشان مي‌دهد. لبه پايين‌رونده، ALE را مي‌توان براي latch كردن آدرس‌هاي دريافتي از ريزپردازنده بكار برد. آدرس روي خطوط آدرس از لبه پايين‌رونده اين سيگنال تا آخر سيكل باس معتبر است.

**: (Address Enable)**

اين سيگنال براي اين منظور بكار مي‌رود كه مشخص نمايد كدام يك از دو وسيله پردازنده يا DMA كنترل باس را در اختيار دارد.

1=AEN : كنترل باس در اختيار DMA است.

0=AEN : كنترل باس در اختيار CPU است.

**CLK :**

فركانس سيستم است كه تمام عمليات خواندن و نوشتن با اين كلاك سنكرون هستند. در كامپيوترهاي PC/XT، CLK=4.7MHZ است. در كامپيوترهاي PC/AT، CLK=6MHZ، است. در PCهاي بعدي اين فركانس بالاتر است. البته مقدار آن توسط بايوس سيستم قابل انتخاب است.

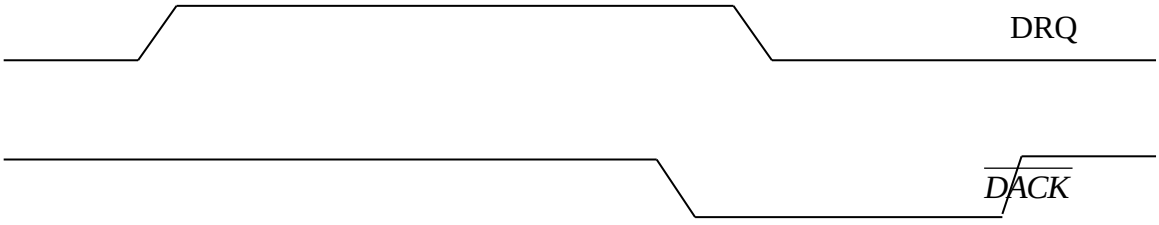
**باس اطلاعات D0-D7 :**

باس اطلاعات دوطرفه ۸ بيتي كه براي ارتباط با I/O، حافظه و ساير المانهاي سيستم مي‌باشد.

***DACK3 ، DACK2 ، DACK1 ، DRQ3 ، DRQ2 ، DRQ1***

***تقاضاي DMA و تصديق DMA :***

اين سيگنالها براي بدست آوردن كانال DMA مورد استفاده قرار ميگيرد. وسايل جانبي كه ميخواهند ازكانال DMA استفاده نمايند، پاية DRQ را 1 ميكنند. اين پايه تا وقتي در حالت 1 ميماند كه DACK فعال شود، (از 1 به 0 تغيير حالت دهد) در مواردي كه بيش از يكي از DRQ1 ، DRQ2 ، DRQ3 فعال شده باشد، DRQ1 بالاترين اولويت و DRQ3 پايينترين اولويت را دارد.




**شكل (١-٢)**

**: (Terminal Count) TC**

TC يك سيگنال خروجي است كه هر وقت شمارنده كانال DMA به پايان شمارش خود مي‌رسد و در واقع سيكل كانال DMA تمام مي‌شود اين پايه 1 مي‌گردد.

**: (I/O Channel Cheek)** $\overline{IOCHCHK}$

سيگنال فوق يك پايه ورودي است كه فعال ٠ مي‌باشد و به مادربورد سيستم خبر بروز خطا در كارت قرار گرفته در اسلات را مي‌دهد. در داخل سيستم پين $\overline{IOCHCHK}$، به پايه $\overline{NMI}$ پردازنده وصل شده تا پردازنده بدون اتلاف وقت به خطاي فوق پاسخ دهد.

**: (I/O Channel Ready)** $\overline{IOCHRDY}$

زمانيكه اين سيگنال توسط يك كارت موجود در اسلات 0 گردد يك تأخير به پردازنده داده خواهد شد. حداكثر زمانيكه اين پايه مي‌تواند 0 گردد توسط سرعت سيستم محدود مي‌گردد. مثلاً در PC/AT با سرعت ٦MHZ اين سيگنال نبايد بيش از $2/5\mu S$، 0 گردد. در واقع اين سيگنال به وسايل سرعت پايين اجازه كار با پردازنده‌هاي سرعت بالا را مي‌دهد.

در شکل زیر خلاصه timing لازم جهت عملکرد درست سیگنال $\overline{IOCHRDY}$ نشان داده شده است. براي يك سيستم PC/XT، اين سيگنال بطور معمول 1 بوده و بايد ۹۰ نانوثانيه قبل از لبة پالس T3 به حالت 1 برگردد.

**$\overline{IOW}$ و $\overline{IOR}$ :**

اين دو سيگنال جهت خواندن و نوشتن از I/O مورد استفاده قرار ميگيرند. فعال شدن $\overline{IOW}$ باعث ميشود تا ابزار I/O اطلاعات را از روي باس ديتا بردارد. فعال شدن $\overline{IOR}$ باعث گذاشتن اطلاعات بر روي باس ديتا توسط وسايل I/O ميگردد.

**IRQ3-IRQ7 و IRQ9 (پايه‌هاي تقاضاي وقفه) :**

وقتي هر كدام از پايه‌هاي فوق از 0 به 1 تغيير ميكنند يك سيگنال وقفه به CPU ارسال ميگردد. اگر همزمان چند پايه فعال گردد، اولويت به IRQ9 و سپس IRQ3 ، IRQ4 و... داده ميشود. ERQ9 بالاترين اولويت و IRQ7 پايين‌ترين اولويت را دارد.

IRQ7 , IRQ6 , ERQ5 .....IRQ3 , IRQ9

بالاترين اولويت  پايين‌ترين اولويت

**OSC :**

سيگنال بعنوان خروجي بوده و فركانس ۱٤/۳۱۸۱۸ مگاهرتز دارد و dluty cycle اين سيگنال ۵۰ درصد مي‌باشد و اين سيگنال با سيگنال CLK سنكرون نمي‌باشد

***REFRESH*** **:**

اين سيگنال خروجي بوده و فعال شدن آن (۰ شدن) نشان مي‌دهد كه يك سيكل تازه‌سازي حافظه‌هاي ديناميكي در حال انجام است.

**RESET DRV (راه انداز پايه ريست) :**

يگ سيگنال خروجي است كه فعال 1 مي‌باشد. و براي ريست كردن ابزارها و كارت‌هاي موجود در اسلات‌ها استفاده مي‌گردد. اين سيگنال توسط مادربورد براي ريست و برنامه‌ريزي كارت‌هاي موجود در اسلات‌ها در زمان روشن شدن سيستم (Power on) و قبل از برنامه‌ريزي توسط BIOS به كار برده مي‌شود.

**$\overline{SMEMW}$ و $\overline{SMEMR}$ (خواندن و نوشتن در حافظه):**

اين دو سيگنال براي خواندن و نوشتن در حافظه مورد استفاده قرار مي‌گيرند. اين دو سيگنال فقط زماني فعال مي‌شوند كه محدوده آدرس در بازه 00000 تا FFFFF (1MB) باشد. اين دو سيگنال بوسيله سيكل‌هاي ٨ بيتي مورد استفاده قرار مي‌گيرند.

**$\overline{0WS}$ (Zero Wait States):**

يك سيگنال ورودي با حالت فعال 0 است كه ميزان تأخير خواندن و نوشتن در I/O يا حافظه را 0 تعيين مي‌كند. مدار مولد تأخير در ماداربورد بطور اتوماتيك يك سيكل تأخير را در سيكل‌هاي خواندن و نوشتن حافظه و I/O قرار مي‌دهد. با اين سيگنال مي‌توان اين تأخير را 0 تعريف نمود. فعال شدن اين سيگنال بايد توسط يك مدار كلكتور باز يا tri-state با جريان ٢٠mA انجام گيرد. باس ISA ١٦ بيتي بدون وجود اين سيگنال در ٣ پالس ساعت كامل مي‌شود و با وجود اين سيگنال در ٢ پالس ساعت كامل خواهد شد.

پايه‌هاي تغذيه $5^{V}+$ و $5^{V}-$ و $12^{V}+$ و $12^{V}-$ و GND و GND

سيگنالهاي قسمت ٣٦ پين، پايين باس ISA بصورت زير مي‌باشند.

**A17-A23 يا (LA17-LA23)(unlatcched Address 17-23):**

همانطور كه مي‌دانيم در قسمت اول اسلات ISA خطوط آدرس A0-A19 وجود دارد بنابراين انتظار مي‌رفت كه فقط خطوط A20-A23 را داشته باشيم. اما يك المثني از سيگنالهاي A17 تا A19 در قسمت اضافه شده ISA قرار داده شده است. سيگنالهاي A0-A19 در اسلاتهاي ISA ٨ بيتي بصورت لچ شده مي‌باشند پروسه لچ به تأخير باس آدرس اضافه مي‌كند. به اين دليل سيگنالهاي LA17-LA23 لچ نشده و مستقيماً به پردازنده وصل مي‌شوند. چون اين سيگنالها لچ نشده‌اند در طول خواندن و نوشتن با ثبات نيستند، به اين دليل بايد بوسيله كارت اضافه شده لچ شوند تا آدرسها با ثبات گردد. اين خطوط آدرس در لبه بالا رونده سيگنال ALE در دسترس بوده و از لبه پايين رونده اين سيگنال مي‌توان بر لچ كردن خطوط آدرس استفاده نمود

**D8-D15 :**

۸ بیت وزن بالای گذرگاه داده ۱۶ بیتی ISA می‌باشند. وسایل I/O می‌توانند با باس ISA بصورت ۸ بیتی یا ۱۶ بیتی ارتباط برقرار نمایند. باس ISA را می‌توان بصورت ۸ بیتی یا ۱۶ بیتی بکار برد. اگر بخواهیم باس ISA را بصورت ۱۶ بیتی بکار ببریم باید پایه ورودی $\overline{IOCS16}$ را ۰ کنیم. اگر این پایه ۱ باشد باس ISA بصورت ۸ بیتی می‌باشد. بسته به اینکه I/O و باس ISA چند بیتی باشد حالات زیر وجود دارد.

۱- باس ISA و I/O هر دو ۸ بیتی باشند : در اینصورت وسیله I/O به خطوط D0-D7 وصل می‌گردد.

۲- باس ISA و I/O هر دو ۱۶ بیتی باشند : در اینصورت وسیله I/O به خطوط D0-D15 وصل می‌گردد.

۳- باس ISA ۱۶بیتی و I/O ۸ بیتی باشد : در اینصورت وسیله I/O به خطوط D0-D7 وصل می‌گردد، در اینصورت هر انتقال داده ۱۶ بیتی در دو سیکل متوالی انجام می‌گیرد که در هر سیکل یك بایت انتقال می‌یابد، و در اینحالت باید پایه $\overline{IOCS16}$ 1 گردد (غیرفعال شود).

**DRQ0 ، DRQ5 ، DRQ6 ، DRQ7 ، DACK0 ، DACK5 ، DACK6 ، DACK7 :**

چهار کانال DMA بوسیله این سیگنالها قابل دستیابي هستند. DRQ0 براي يك كانال DMA ٨ بيتي است. در حاليكه DRQ5-DRQ7 كانالهاي ١٦ بيتي مي‌باشند. در كامپيوترهاي XT كانال DMA0 براي Refresh حافظه‌هاي DRAM استفاده مي‌شد. ولي در كامپيوترهاي AT اين كانال آزاد شده و عمل Refresh حافظه‌هاي DRAM به طريق ديگر انجام مي‌شود.

: $\overline{IOCS16}$

يك سيگنال ورودي فعال است كه به باس ديتا اجازه انتقال ١٦ بيتي را مي‌دهد. در كامپيوترهاي AT براي سازگاري با كامپيوترهاي XT، گذرگاه داده ٨ بيتي است. و در واقع اگر بخواهيم با باس ISA بصورت ١٦ بيتي كار كنيم بايد اين پايه را ٠ كنيم. اين پايه بايد از طريق يك درايور كلكتور باز يا tri-state با جريان‌دهي حداقل ٢٠mA راه‌اندازي شود.

: $\overline{MEMCS16}$

يك سيگنال ورودي فعال ٠ است كه جهت كار كردن حافظه‌ها با پردازنده بصورت ١٦ بيتي بكار برده مي‌شود. اين پايه بايد از طريق يك درايور كلكتور باز يا tri-state با جريان‌دهي حداقل ٢٠mA راه‌اندازي شود.

**:IRQ10-IRQ15**

اين پايه، پايه‌هاي وقفه هستند كه داراي اولويت بالاتري نسبت به وقفه IRQ3 تا IRQ7 مي‌باشند ولي اولويت پايين‌تري نسبت به IRQ9 دارند.

***MASTIER*** **يا #MASTER16 :**

اين سيگنال ورودي فعال صفر است. كه بوسيله كارت ISAي ١٦ بيتي فعال مي‌شود. معمولاً اين سيگنال وقتي فعال مي‌شود كه كارت ISA ي طراحي شده خودش داراي پردازنده باشد و بخواهد كنترل باس سيستم را در اختيار بگيرد. در اينحالت جهت بافرهاي باس معكوس مي‌شود. و MASTER روي كارت ISA، آدرس و اطلاعات مربوط به نوع سيكل باس را فعال مي‌كند. نكته‌اي كه در اينجا وجود دارد آنستكه MASTER خودش نمي‌تواند

مستقيماً به باس سيستم دسترسي پيدا كند بلكه براي اين منظور از كانالهاي DMA استفاده ميكند. به اين دليل قبل از فعال كردن ورودي MASTER، كارت ISA بايد يكي از پايه‌هاي DRQ را فعال كند و پس از دريافتن سيگنال DACK، كارت ISA ميتواند پاية MASTER را فعال كند.

**$\overline{MEMR}$ و $\overline{MEMW}$ :**

اين دو سيگنال براي خواندن و نوشتن در حافظه مورد استفاده قرار ميگيرند. اين دو سيگنال بر خلاف سيگنالهاي $\overline{SMEMR}$ و $\overline{SMEMW}$ كه فقط در محدوده FFFFF-00000 فعال ميشوند، در تمام سيكل‌هاي خواندن و نوشتن فعال هستند.

**$\overline{BHE}$ يا #SBHE (System BUS High Enable) :**

وقتي اين سيگنال فعال ميشود نشان ميدهد كه داده در ۸ بيت وزن بالاي گذرگاه ISA يعني در خطوط D8-D15 در حال انتقال ميباشد. جدول (۱-۱) چگونگي $\overline{BHE}$ و A0 را براي انتقال داده روي خطوط D0-D15 نشان ميدهد.

| $\overline{BHE}$ | A0 | وضعيت گذرگاه داده |
|---|---|---|
| 0 | 0 | انتقال ١٦ بيت روي D0 – D15 |
| 0 | 1 | انتقال يك بايت روي نيمه بالايي گذرگاه داده D8-D15 |
| 1 | 0 | انتقال يك بايت روي نيمه پاييني گذرگاه داده D0-D7 |
| 1 | 1 | رزرو (گذرگاه داده بيكار است) |

**جدول (١-١)**

پردازنده هاي 80286 و 80386SX پايه $\overline{BHE}$ را در طول عمليات انتقال ٨ بيتي با آدرس فرد و در طول عمليات انتقال ١٦ بيتي با آدرس زوج فعال مي‌كنند.

**پايه هاي $5^V$+ و QND**

انواع سيكل‌هاي باس ISA بصورت جدول (١-٢) مي‌باشد :

| خطوط ISAي فعال شده | نوع سيكل باس |
|---|---|
| هيچكدام | Interrupt Acknowledge |

| | |
|---|---|
| I/O Write | $\overline{IOW}$ |
| I/O Read | $\overline{IOR}$ |
| Halt or Shutdown | هيچ كدام |
| Memory Write | $\overline{SMEMW}$ و $\overline{MEMW}$ : |
| Memory Read | $\overline{SMEMR}$ و $\overline{MEMR}$ |

**جدول (۱-۲)**

**نمودارهاي سيكل‌هاي باس ISA :**

**عمليات انتقال ۸ بيتي :**

باس ISA مي‌تواند با وسايل ۸ بيتي به يكي از روشهاي زير مبادله داده انجام دهد:

- سيكل مربوط به وسايل ٨ بيتي استاندارد با ٤ Wait State
- سيكل مربوط به وسايل ٨ بيتي Snortened با ١، ٢ يا ٣ Wait State
- سيكل مربوط به وسايل ٨ بيتي Stretched با بيشتر از ٤ Wait State

در شكل (١-٣) نمودار زماني عمليات انتقال ٨ بيتي با ٤ Wait State نشان داده شده است.

پاية BALE ، 1 ميگردد و آدرس روي خطوط آدرس SA لچ ميگردد. وسيله جانبي Slave بطور قابل اطمينان ميتواند آدرس را در لبة پايينرونده BALE بردارد. خطوط آدرس تا آخر عمليات انتقال معتبر ميمانند. توجه كنيد كه AEN در طول سيكل انتقال 0 ميباشد.

خط فرمان كه عبارتست از $\overline{IOR}$ يا $\overline{IOW}$ براي وسايل I/O و $\overline{SMEMR}$ يا $\overline{SMEMW}$ براي حافظه، صفر ميگردد.

پايه $\overline{0WS}$ در وسط هر سيكل Wait نمونهبرداري ميشود. اگر پايه صفر باشد سيكل انتقال بدون سيكل Wait ديگر خاتمه پيدا ميكند.

پايه $\overline{IOCHRDY}$ در نيمه اول هر سيكل نمونهبرداري مي شود. اگر اين پايه صفر باشد سيكلهاي Wait اضافي

مي‌توان وارد نمود. مقدار پيش‌فرض سيكل‌هاي Wait براي وسايل ٨ بيتي برابر ٤ سيكل مي‌باشد. در برخي از كامپيوترها تعداد اين سيكل‌ها قابل تغيير است.

**شكل(١-٣)**

**عمليات انتقال ١٦ بيتي :**

باس ISA مي‌تواند با وسايل ١٦ بيتي به يكي از روشهاي زير مبادله داده انجام دهد :

- سيكل مربوط به وسايل ١٦ بيتي استاندارد (براي كار با حافظه يا وسايل I/O) با Wait state 1
- سيكل مربوط به وسايل ١٦بيتي Shortened (براي كار با حافظه) با 0 Wait state
- سيكل مربوط به وسايل ١٦ بيتي Stretched با بيشتر از 1 Wait state

در شكل(١-٤) نمودار زماني عمليات انتقال ١٦ بيتي با 1 Waitstate نشان داده شده است.

**شكل (١-٤)**

علامتهاي * نقاطي را نشان مي‌دهند كه سيگنال در آنجا نمونه‌برداري مي‌شود.

[1] بخشي از آدرس روي خطوط A باس كه براي سيكل بعدي استفاده مي‌شوند از نقطه [1] مي‌توانند روي باس قرار گيرند. اين امر به اين خاطر است كه وسايل جانبي بتوانند زودتر آدرس را ديكو كنند. و در واقع يك نوع pipelining آدرس مي‌باشد.

[2] AEN در طول سيكل صفر است كه نشان مي‌دهد يك عمليات انتقال بدون كمك DMA در حال انجام است.

[3] برخي از كنترلرهاي باس، پايه $\overline{IOCS16}$ را مشابه پاية $\overline{MEMCS16}$ نمونه‌برداري مي‌كنند (بجاي نمونه‌برداري در اولين سيكل Wait كه در شكل نشان داده شده است) در اينحالت بايد پايه $\overline{IOCS16}$ بمحض

دیکود شدن آدرس صفر گردد یعني قبل از فعال شدن خطوط فرمان.

[4] در مواردي که کارت توسعه در نمونه اول M16 را فعال نمود این پایه دفعه دوم نمونه برداري مي‌شود.

در عملیات انتقال ١٦ بیتي، قبل از شروع سیکل انتقال یك آدرس معتبر روي خط LA قرار داده مي‌شود. بر خلاف خطوط SA، خطوط LA لچ شده نیستند و در تمام سیکل انتقال معتبر باقي نمي‌مانند. خطوط LA باید در لبه پایین رونده BALE لچ شوند. کارتهاي I/O نیازي ندارند که بر خطوط LA یا BALE نظارت کنند، چون آدرس مربوط به وسایل I/O در محدوده آدرس خطوط SA قرار دارد.

$\overline{BHE}$ بوسیله بورد سیستم فعال مي‌شود و کارت نصب شده روي باس ISA باید به این سیگنال پاسخ مناسب بدهد. این کار را با فعال کردن خطوط $\overline{IOCS16}$ و یا $\overline{MEMCS16}$ انجام مي‌شود. در غیر اینصورت عملیات انتقال به ٢ عملیات انتقال ٨ بیتي تقسیم مي‌شود. بیشتر سیستم‌ها نیازمند آن هستند که قبل از فعال شدن خطوط فرمان خطوط $\overline{IOCS16}$ و یا $\overline{MEMCS16}$ فعال

شوند، اين نيازمند آن است كه پايه‌هاي $\overline{IOCS16}$ و يا $\overline{MEMCS16}$ به محض ديكود شدن آدرس فعال شوند.

مقدار پيشفرض سيكلهاي Wait براي وسايل ١٦ بيتي ١ سيكل مي‌باشد. تعداد اين سيكل‌ها بوسيله پايه‌هاي $\overline{0WS}$ و يا $\overline{IOCHRDY}$ كم يا زياد مي‌شوند. پايه‌هاي $\overline{SMEMR}$ و $\overline{SMEMW}$ بترتيب داراي همان timing پايه‌هاي $\overline{MEMR}$ و $\overline{MEMW}$ مي‌باشند اگر آدرس در محدود 1MB پايين باشد. اگر آدرس در محدوده 1MB پايين نباشد خطوط $\overline{MEMR}$ و $\overline{MEMW}$، ١ مي‌گردند براي يك وسيله ٨ بيتي اين امكان وجود دارد كه از قسمت بالاي باس ديتا استفاده كند. در اين حالت timing مشابه سيكل ١٦بيتي خواهد بود اما آدرس‌هاي فرد روي باس ظاهر مي‌شوند. يعني باس از خطوط ديتاي SD8-SD15 براي انتقال استفاده مي‌كند.

### ***كم و زياد كردن سيكل باس :***

دياگرام زماني شكل(1-٥) سه سيكل I/O مختلف را نشان مي‌دهد.

اولين سيكل يك سيكل خواندن از I/O ١٦ بيتي استاندارد مي‌باشد.

دومين سيكل يك سيكل خواندن از I/O ١٦ بيتي مي‌باشد كه يك Wait State به آن وارد شده است. كانال I/O پايه $\overline{IOCHRCY}$ را صفر مي‌كند كه نشان مي‌دهد آماده نيست تا عمليات انتقال را تكميل كند. اين باعث ايجاد يك سيكل Wait مي‌شود و پايه $\overline{IOCHRDY}$ دوباره نمونه‌برداري مي‌شود. در دفعه دوم كانال I/O عمليات انتقال را تكميل نموده و پايه $\overline{IOCHRDY}$ ١ شده است. سيكل سوم يك سيكل خواندن از I/O ٨ بيتي مي‌باشد. كه داراي يك عدد سيكل Wait مي‌باشد. توجه كنيد كه در حالت معمولي سيكل ٨ بيتي داراي ٤ سيكل Wait است كه در اينجا با صفر كردن پاية $\overline{OWS}$ اين سيكل به ١ عدد كاهش يافته است.

**شکل(۱-۵)**

**زمان سیکل I/O در گذرگاه ISA :**

80286 و پردازنده‌هاي بالاتر هر عملیات انتقال را حداقل در ۲ سیکل BCLK انجام مي‌دهند در این حالت wait state صفر مي‌باشد. سرعت گذرگاه ISA بدون توجه به

سرعت CPU در ۸ MHZ محدود شده است. بنابراين زمان هر سيكل BCLK برابر يك هشتم نانو ثانيه يا ۱۲۵ نانوثانيه مي‌باشد. هر عمليات انتقال ۱۶ بيتي حداقل در ۲ سيكل BCLK انجام مي‌شود يعني هر عمليات انتقال ۱۶بيتي به زمان ns ۲۵۰=۱۲۵*۲ احتياج دارد.

حال فرض كنيد پردازنده سيستم ۸۰۲۸۶ با كلاك ۲۰ مگاهرتز استفاده شود. زمان سيكل ساعت آن $\frac{1}{20} = 50ns$ مي‌باشد بنابراين به $\frac{250}{50}$ يا 5 عدد سيكل لازم دارد تا يك سيكل I/O، ISA گردد. و چون حداقل ۲ سيكل بوسيله CPU تأمين مي‌شود بنابراين به ۳ سيكل Wait State نيازمنديم تا عمليات انتقال را انجام دهيم.

**آدرس وسايل I/O :**

فقط ۱۰ بيت وزن پايين خطوط آدرس براي عمليات I/O، ديكود مي‌شوند. بنابراين محدوده آدرس I/O بين 0 تا 3FF قرار دارد. برخي از اين آدرس‌ها بوسيله كنترلرهاي روي مادربورد نظير كنترلر DMA، كنترلر وقفه و ... اشغال شده‌اند كه در جدول (۱-۳) آورده شده است. همانطور كه ملاحظه مي‌گردد محدوده آدرس

300H-31FH براي وسايل جانبي و كارتهاي توسعه طراحي شده در نظر گرفته شده است.

**جدول(۱-۳)**

**جدول مربوطه همراه با جدول (۱-٤) آمده است**

**: DMA**

باس ISA از دو کنترلر DMA استفاده مي‌کند که بطور کسکود به هم وصل شده‌اند. يکي از آنها Slave بوده و از طريق کانال شماره ٤ به کنترلر master وصل مي‌گردد. بنابراين کنترلر DMA ، Slave از طريق کنترلر DMA ، master کنترل باس را در اختيار مي‌گيرد. در باس ISA کنترلر DMA بطور اولويت ثابت برنامه‌ريزي مي‌گردد (کانال صفر داراي بالاترين اولويت مي‌باشد) بنابراين کانالهاي ٠ الي ٤ از کنترلر Slave داراي بالاترين اولويت مي‌باشند (چون به کانال ٠ کنترلر master وصل شده‌اند) و کانالهاي ٥ الي ٧ در اولويت بعدي قرار دارند (چون به کانالهاي ١ الي ٣ کنترلر master وصل شده‌اند).

کنترلر DMA مي‌تواند براي عمليات خواندن (خواندن از حافظه و نوشتن در وسيله I/O) براي عمليات نوشتن (خواندن از I/O و نوشتن در حافظه) و يا براي عمليات Verify (در کامپيوترهاي قديمي براي refresh حافظه هاي DRAM بوسيله کانال 0 استفاده مي‌شد) برنامه‌ريزي گردد.

قبل از شروع عمليات انتقال، كنترلر DMA بايد برنامه‌ريزي گردد. اين كار با نوشتن آدرس شروع، تعداد بايتهاي انتقالي و جهت انتقال در كنترلر DMA انجام مي‌گيرد. بعد از اينكه كنترلر DMA برنامه‌ريزي گرديد، وسيله جانبي مي‌تواند پايه DRQ مناسب را فعال كند.

**مد انتقال منفرد :**

كنترلر DMA براي عمليات انتقال برنامه‌ريزي مي‌گرد. وسيله جانبي با فعال كردن پايه DRQ مناسب تقاضاي انتقال مي‌نمايد. كنترلر DMA در پاسخ پايه AEN را 1 مي‌كند و پاية DACK را فعال مي‌سازد. خطوط فرمان مربوط به حافظه و وسايل I/O (خطوط خواندن يا نوشتن) نيز فعال مي‌شود. وقتي وسيله جانبي سيگنال DACK را ديد پاية DRQ را غيرفعال مي‌سازد، كنترلر DMA، آدرس حافظه را روي باس ISA قرار مي‌دهد (در همان لحظه كه خطوط فرمان فعال شده‌اند) وسيله جانبي بسته به نوع عمليات انتقال از حافظه مي‌خواند يا در آن مي‌نويسد. شمارنده انتقال كاهش مي‌يابد و آدرس ممكن است كاهش يا افزايش يابد. حالا

پایهٔ DACK غیرفعال میگردد و CPU دوباره کنترل باس را در اختیار میگیرد تا زمانیکه وسیله جانبي دوباره براي عملیات انتقال حاضر گردد. وسیله جانبي پروسه بالا را تکرار میکند یعني پایه DRQ را فعال میکند و منتظر DACK میماند. این پروسه به اندازه شماره انتقال، تکرار میشود. بعد از تکمیل عملیات انتقال، کنترلر DMA از طریق سیگنال TC (Terminal Count) به CPU خبر میدهد که عملیات انتقال خاتمه یافته است.

**مد انتقال بلوکي :**

کنترلر DMA براي عملیات انتقال برنامهریزي میگردد. وسیله جانبي با فعال کردن پایه DRQ مناسب تقاضاي انتقال مینماید، مادربورد در پاسخ پایهٔ AEN را 1 و پایه DACK را 0 میکند. این نشان میدهد که وسیله جانبي حالا بعنوان BUS MASTER عمل میکند. در پاسخ به DACK وسیله جانبي پایه DRQ را غیرفعال میکند. کنترلر DMA آدرس لازم براي عملیات انتقال را روي باس آدرس قرار میدهد. هر دو خطوط فرمان مربوط به حافظه و وسایل I/O فعال میگردد. پایه AEN

وسايل I/O را از پاسخ دادن به خطوط فرمان I/O باز مي‌دارد، زيرا اگر چه خطوط فرمان I/O فعال هستند ولي يك آدرس حافظه روي باس آدرس وجود دارد. عمليات انتقال انجام مي‌شود (خواندن از يا نوشتن در حافظه) شمارنده انتقال كاهش مي‌يابد و آدرس ممكن است كاهش يا افزايش يابد. اين سيكل به تعداد برابر با شمارنده انتقال تكرار مي‌شود. وقتي عمليات انتقال تمام شد كنترلر DMA از طريق سيگنال TC به CPU خبر مي‌دهد كه عمليات انتقال خاتمه يافته است.

**مد انتقال Demand :**

كنترلر DMA براي عمليات انتقال برنامه‌ريزي مي‌شود. وسيله جانبي پاية DRQ مناسب را فعال مي‌كند و مادربورد با فعال كردن AEN (1) و DACK (0) پاسخ مي‌دهد. اين نشان مي‌دهد كه وسيله جانبي حالا بعنوان BUS MASTER عمل مي‌كند. بر خلاف عمليات انتقال از نوع منفرد و بلوكي، وسيله جانبي پايه DRQ را غيرفعال نمي‌كند. وسيله جانبي به همان روش بلوكي ديتا را منتقل مي‌كند. كنترلر DMA تا

زمانيكه پايه DRQ فعال باشد سيكل‌هاي DMA را توليد مي‌كند. وقتيكه وسيله جانبي از ادامه عمليات انتقال ناتوان گردد (بعنوان مثال ديتاي آماده نداشته باشد) پايه DRQ را غيرفعال مي‌كند و CPU دوباره كنترل باس را در اختيار مي‌گيرد. كنترل باس وقتي در اختيار وسيله I/O قرا رمي‌گيرد كه DRQ دوباره فعال شود. اين پروسه تا زماني ادامه مي‌يابد كه به شمارنده انتقال برسيم و در اينصورت سيگنال TC به CPU پايان عمليات انتقال را خبر مي‌دهد.

در كامپيوترهاي PC/XT از كنترلر DMA به شماره 8237 ساخت اينتل استفاده مي‌شده است. اين كنترلر داراي ٤ كانال DMA مستقل از هم و قابل برنامه‌ريزي مي‌باشد. هر كانال قبل از استفاده بايد بطور جداگانه مقداردهي اوليه شود. مقداردهي اوليه يك كانال عبارتست از نوشتن موارد زير در هر كانال :

١- آدرس اولين بايت بلوك داده كه بايد منتقل گردد (كه به آن آدرس مبنا نيز مي‌گويند).

٢- تعداد بايتهاي مورد انتقال

اگر چه چهار كانال فوق بايد بطور جداگانه براي آدرس مبنا و تعداد بايتهاي انتقالي برنامه‌ريزي گردند ولي همه كانالها تنها داراي مجموعه ثبات كنترل/ وضعيت مي‌باشد كه در جداول (١-٤) و (١-٥) كل ثباتها براي كنترلر Slave و master نشان داده شده‌اند. براي دسترسي به هر كدام از ثباتها از طريق آدرس آفست نوشته شده در كنار آنها اقدام مي‌كنيم. بدين ترتيب كه خود كنترلر DMA داراي يك آدرس $\overline{CS}$ مي‌باشد كه اين آدرس بعلاوه آدرس آفست نوشته شده براي دسترسي به ثباتهاي مورد نظر بكار برده مي‌شود.

جدول (١-٤)

**جدول (۱-۵)**

همانطور که ملاحظه مي‌شود براي هر کانال رجيسترهاي آدرس مبنا و تعداد کلمات انتقالي وجود دارد علاوه بر اين مي‌توان به رجيسترهاي زير اشاره نمود :

**رجيستر کنترل :**

اين رجيستر ۸ بيتي بوده و در آدرس آفست 8h قرار دارد و فقط بوسيله CPU قابل دسترسي است. بيتهاي آن به شرح زير است :

D0 : امکان عمليات انتقال از حافظه به حافظه را فراهم مي‌کند.

توضيح : 8237 قادر است داده را از وسيله جانبي به حافظه، از حافظه به وسيله جانبي و يا از حافظه به حافظه منتقل کند. مثالي از عمليات انتقال از حافظه به حافظه، عمليات انتقال از ROM به RAM مي‌باشد. بدين ترتيب که چون زمان به دسترسي به ROM طولاني است CPU مي‌تواند ROM را در بخشي از

RAM كپي كند و با سرعت بيشتر به آن دسترسي پيدا كند. براي اين منظور از دو كانال DMA 0 و 1 استفاده مي‌شود. كانال 0 براي منبع و كانال 1 براي مقصد استفاده مي‌گردد. كانال 0 بايت را از منبع به يك ثبات موقت در داخل 8237 مي‌ريزد و كانال 1 بايت را از آنجا به مقصد منتقل مي‌كند. اين بر خلاف انتقال‌هاي I/O به حافظه يا حافظه به I/0 مي‌باشد كه در آن داده به گذرگاه داده خوانده مي‌شود و از آنجا بدون ذخيره شدن در هر محل موقت، به مقصد منتقل مي‌گردد.

D1 : تنها به هنگام فعال شدن عمليات انتقال از حافظه به حافظه فعال مي‌شود و مي‌تواند افزايش يا كاهش آدرس حافظه كانال 0 را از كار بيندازد تا يك مقدار ثابت در بلوكي از حافظه نوشته شود.

D2 : براي فعال يا غيرفعال كردن كنترلر DMA مي‌باشد.

D3 : مي‌توان زمان سيكل حافظه را بوسيله اين بيت انتخاب نمود (معمولي يا فشرده) در حالتي معمولي زمان سيكل حافظه حداقل ٤ پالس ساعت مي‌باشد. در

حالت فشرده زمان سيكل حافظه حداقل ۲ پالس ساعت مي‌باشد.

D4 : براي انتخاب اولويت كانال DMA بصورت ثابت يا چرخشي بكار مي‌رود. در حالت ثابت كانال 0 داراي بالاترين اولويت و كانال ۳ داراي پايين‌ترين اولويت مي‌باشد.

D5 : طولاني شدن سيگنال نوشتن براي وسايل كند را فراهم كند.

D6 : سطح فعاليت پاية DRQ را انتخاب مي‌كند.

D7 : سطح فعاليت پايه DACK را انتخاب مي‌كند.

**رجيستر وضعيت :**

اين رجيستر ۸ بيتي بوده و در همان آدرس آفست 8h قرار دارد و فقط بوسيله CPU قابل دسترسي مي‌باشد. اين رجيستر فقط خواندني مي‌باشد. و اگر CPU در آدرس 8h بنويسد در رجيستر كنترل نوشته مي‌شود.

بيتهاي D0-D3 فعال شدن سيگنال TC براي كانالهاي ۰ الي ۳ يعني خاتمه عمليات انتقال كانالهاي ۰ الي ۳ را نشان مي‌دهند.

بيتهاي D4-D7 كانالهاي DMA را كه داراي تقاضاي DMA در حال انتظار مي‌باشند، نشان مي‌دهد.

**رجيستر مد :**

اين رجيستر ۸ بيتي بوده و در آدرس آفست Bh قرار دارد و فقط بوسيله CPU قابل دسترسي است. بيتهاي آن به شرح زير مي‌باشند :

D0 و D1 : براي انتخاب كانال DMA مي‌باشند.

D2 و D3 : نوع انتقال را مشخص مي‌كنند. انواع عمليات انتقال به شرح زير مي‌باشد.

عمليات انتقال نوشتن :

كانال DMA با فعال كردن خطوط $\overline{IOR}$ و $\overline{MEMW}$ داده را از يك وسيله I/O مي‌خواند و در حافظه مي‌نويسد.

عمليات انتقال خواندن :

كانال DMA با فعال كردن خطوط $\overline{IOW}$ و $\overline{MEMR}$ داده را از حافظه مي‌خواند و در I/O مي‌نويسد.

عمليات انتقال Verify :

شبیه عملیات انتقال نوشتن یا خواندن می‌باشد با این تفاوت که هیچ سیگنال کنترلي خواندن یا نوشتن تولید نمي‌شود. این عملیات بوسیله کانال شماره 0 در کامپیوترهاي PC/XT براي Refresh حافظه‌هاي DRAM بکار مي‌رفت.

D4 : اگر این بیت 1 باشد، رجیستر آدرس پایه و رجیستر شمارنده تعداد کلمات انتقالي، در پایان انتقال داده بوسیله DMA، با مقادیر اولیه خود بار مي‌شوند. (هنگامي رجیستر شمارنده کلمات انتقالي به صفر مي‌رسد)

D5 : براي افزایش یا کاهش آدرس مي‌باشد.

D6 و D7 : طریقه استفاده از 8237 را مشخص مي‌سازند، انتخاب‌ها عبارتند از :

١- مد درخواست ٢- مد بلوکي ٣- مد تکي

٤- مد متوالي

مد متوالي براي متوالي کردن چندین کنترلر 8237 بکار مي‌رود.

**رجیستر ماسك تکي (پوشش) :**

اين رجيستر فقط بوسيله CPU قابل دسترسي بوده و در آدرس Ah قرار دارد. از ۸ بيت اين رجيستر فقط ۳ بيت آن قابل استفاده است.

D0 و D1 : براي انتخاب كانال DMA مي‌باشند.

D2 : اگر اين بيت فعال گردد كانال DMA مربوطه mask مي‌گردد. يعني حتي اگر پايه DRQ آن كانال فعال گردد كنترلر DMA به درخواست آن كانال پاسخ نمي‌دهد.

اين رجيستر در هر بار فقط يك كانال را mask مي‌كند. اگر بخواهيم بيش از يك كانال را mask كنيم بايد از رجيستر ماسك كلي كه در آدرس Fh قرار دارد و مشابه رجيستر ماسك تكي مي‌باشد استفاده كنيم. با رجيستر ماسك كلي مي‌توان چندين كانال DMA را همزمان mask كرد.

### رجيستر *DMA Master* :

اين رجيستر در آدرس Dh قرار دارد. فقط بوسيله CPU قابل دسترسي مي‌باشد. بايت نوشته شده در اين رجيستر اهميتي ندارد و اگر در آن بنويسيم باعث ريست كنترلر مي‌شود.

**رجيستر پاك كردن رجيستر mask :**

اين رجيستر در آدرس Eh قرار دارد فقط بوسيله CPU قابل دسترسي بوده و بايت نوشته شده در آن اهميتي ندارد و اگر در آن بنويسيم بيتهاي mask مربوط به كانالهاي DMA پاك ميشوند.

**رجيستر صفحه DMA :**

هر كانال DMA ميتواند هر محل حافظه را كه در محدوده ١٦MB بين 00000h-FFFFFh قرار داد، آدرس كند. اينكار با استفاده از رجيستر آدرس ١٦ بيتي كه براي هر كانال DMA مستقلاً وجود دارد، و نيز رجيستر صفحه ٨ بيتي كه براي هر كانال DMA مستقلاً وجود دارد انجام ميگيرد. در واقع دو جفت رجيستر با هم يك آدرس ٢٤ بيتي را فراهم ميكنند. در واقع يك آدرس ٢٤ بيتي بوسيله رجيسترهاي زير توليد ميشود:

A16 A15 A0

A23

| رجيستر آدرس كانال DMA | رجيستر صفحه كانال DMA |
|---|---|

**شكل(١-٦)**

همانطور كه ملاحظه ميگردد هركانال DMA ميتواند بلوكهاي داده را كه حداكثر اندازه آنها ٦٤KB ميباشد منتقل نمايد.

**نكته :** براي نوشتن آدرس ١٦ بيتي در داخل رجيستر آدرس كانال DMA، دو عمليات نوشتن متوالي ٨ بيتي انجام ميشود.

البته توضيحات فوق براي كانالهاي ٠ الي ٣ كه عمليات انتقال را ٨ بيتي انجام ميدهند صادق است.

براي كانالهاي ٥ الي ٧ كه عمليات انتقال را ١٦ بيتي و براي آدرسهاي زوج انجام ميدهند يك آدرس ٢٤ بيتي بصورت زير ساخته ميشود.

A17 A16 A1 A0

A23

| | | |
|---|---|---|
| 0 | رجيستر آدرس كانال DMA | رجيستر صفحه كانال DMA |

**شكل(١-٧)**

همانطور كه ملاحظه مي‌شود در اينحالت كانال DMA مي‌تواند بلوكهاي داده را كه حداكثر اندازه آنها ١٢٨KB مي‌باشد منتقل كند.

رجيسترهاي صفحه، مربوط به كانالهاي DMA در آدرسهاي نشان داده شده در جدول(١-٦) قرار دارند.

| آدرس | ثبات صفحه |
|---|---|
| 87h | كانال DMA0 |

| | |
|---|---|
| کانال DMA1 | 83h |
| کانال DMA2 | 81h |
| کانال DMA3 | 82h |
| کانال DMA5 | 8Bh |
| کانال DMA6 | 89h |
| کانال DMA7 | 8Ah |

**جدول (۱-٦)**

## مقدمه اي بر كارت صوت

كارت صوت يكي از عناصر سخت افزاريست كه در كامپيوتر براي پخش و ضبط صدا به كار گرفته مي‌شود. قبل از مطرح شدن كارتهاي صدا، كامپيوترهاي شخصي براي پخش صدا، صرفاً قادر به استفاده از يك بلندگوي داخلي بودند كه از برد اصلي توان خود را مي‌گرفت. در اواخر سال ۱۹۸۰ استفاده از كارت صدا در كامپيوتر شروع و همزمان با آن تحولات گسترده اي در زمينه كامپيوترهاي چند رسانه اي ايجاد گرديد. در سال ۱۹۸۹ شركت Creative كارت صداي خود را به نام Creative labs Sound Blaster Card عرضه نمود. در ادامه شركتهاي متعدد ديگري از قبيل Zotix Opti نيز توليدات خود را عرضه كردند.

## مباني كارت صوت:

كارتهاي صوتي امروزي معمولاً داراي بخشهاي متفاوت زيرند:

۱ ـ يك مبدل آنالوگ به ديجيتال (ADC) براي صوت ورودي به كامپيوتر

۲ ـ يك مبدل ديجيتال با آنالوگ (DAC) براي صوت خروجي از كامپيوتر

۳ ـ يك پردازنده سيگنال‌هاي ديجيتال (DSP) كه مسئول انجام اغلب عمليات (محاسبات) مورد نظر است.

٤ ـ حافظه ROM و يا Flash براي ذخيره سازي داده

٥ ـ يك اينترفيس دستگاههاي موزيكال ديجيتالي (MIDI) براي اتصال دستگاههاي موزيك خارجي

٦ ـ كانكتورهاي لازم براي اتصال به ميكروفون و يا بلندگو

۷ ـ يك پورت خاص بازي براي اتصال Joystick

اغلب كارتهاي صوت امروزي از نوع PCI بوده و در يكي از اسلات‌هاي آزاد برد اصلي (Main beard) نصب مي‌گردند. تا چندي پيش اغلب كارتهاي صوت از نوع ISA بودند. امروزه اكثر كامپيوترهاي جديد كارت صوت را به صورت يك تراشه و بر روي برد اصلي دارند. در اين نوع كامپيوترهاي جديد اسلاتي بر روي برد اصلي استفاده نشده است. و بدين ترتيب مي توان گفت كه يك اسلات صرفه‌جويي شده است. Sound Blaster Pro به عنوان يك استاندارد در دنياي كارتهاي صوت مطرح

است. در زير يك نمونه از اين كارتها را ملاحظه مي‌كنيد.

اغلب توليدكنندگان كارت صوت از مجموعه تراشه‌هاي مشابه استفاده مي‌كنند. پس از طراحي تراشه‌هاي فوق توسط شركت مربوط توليدكنندگان كارت صوت. امكانات و قابليت‌هاي دلخواه خود را به آنها اضافه مي‌نمايند.

كارت صوت را مي‌توان به دستگاههاي زير متصل كرد:

هدفون

بلندگو (Speaker)

يك منبع ورودي آنالوگ نظير: ميكروفون ، راديو، ضبط صوت و CP Player.

يك منبع ورودي ديجيتال نظير CD-ROM

يك منبع آنالوگ خروجي نظير ضبط صوت

يك منبع ديجيتال خروجي مثل CD-WR

**عملیات کارت صدا:**

یك کارت صدا قادر به انجام ٤ عملیات خاص در رابطه با صداست:

١ـ پخش موزیك‌هاي از قبل ضبط شده (از CD، فایل‌هاي صوتي نظیر MP3 و یا Ware) بازي یا DVD

٢ـ ضبط صدا با حالت متفاوت

٣ـ ترکیب نمودن صداها

٤ـ پردازش صوت‌هاي موجود

عملیات دریافت براي کارتهاي صوت از طریق بخش ADC و عملیات ارسال صوت از طریق بخش DAC انجام مي‌گیرد.

پردازش‌هاي لازم بر روي صوت یا توسط بخش DSP انجام مي‌گیرد و یا از CPU خود کامپیوتر براي پردازش صوت استفاده مي‌شود. بدین شکل که صوت پس از دیجیتال شدن توسط عملیات نرم افزاري پردازش مي‌شود.

**تولید صوت:**

فرض کنید قصد داشته باشیم که از طریق میکروفون صداي خود را به کامپیوتر انتقال دهیم. در این حالت کارت صدا یك فایل صوتي با فرمت war را ایجاد کرده و داده‌هاي ارسالي توسط میکروفون در آن ذخیره گردند. فرآیند فوق شامل مراحل زیر است:

۱ـ کارت صوت از طریق کانکتور میکروفون سیگنال‌هاي پیوسته و آنالوگي را دریافت مي‌دارد.

۲ـ از طریق نرم‌افزار مربوطه نوع دستگاه ورودي براي ضبط صدا را مشخص مي‌نمائیم.

۳ـ سیگنال آنالوگ ارسالي توسط میکروفون بلافاصله توسط تراشه مبدل آنالوگ به دیجیتال (ADC) تبدیل و یك فایل حاوي صفر و یك تولید مي‌گردد.

٤ـ خروجي تولید شده توسط ADC در اختیار تراشه DSP براي انجام پردازش‌هاي لازم گذاشته مي‌شود. DSP توسط مجموعه دستوراتي که در تراشه دیگر است برنامه‌ریزي براي انجام عملیات خاص مي‌گردد. یکي از عملیاتي که DSP انجام مي‌دهد فشرده‌سازي داده‌هاي دیجیتال به منظور ذخیره‌سازي است.

۵ـ خروجي DSP با توجه به نوع اتصالات كارت صدا در اختيار گذرگاه داده كامپيوتر قرار مي‌گيرد.

٦ـ داده‌هاي ديجيتال توسط پردازنده اصلي كامپيوتر پردازش و ادامه براي ذخيره‌سازي در اختيار كنترل كننده هارد ديسك گذاشته مي‌شود. كنترل كننده هارد ديسك اطلاعات را بر روي هارد و بعنوان يك فايل ضبط شده صوتي ذخيره خواهد كرد.

**شنيدن صوت:**

مراحل شنيدن صوت به شرح زير است:

١ـ داده‌هاي ديجيتال از هارددیسك خوانده شده و در اختيار CPU قرار مي‌گيرند.

٢ـ پردازنده اصلي داده‌ها را براي DSP موجود بر روي كارت صدا ارسال مي‌دارد.

٣ـ DSP داده‌هاي ديجيتال را از حالت فشرده خارج مي‌نمايد.

٤ـ داده‌هاي ديجيتال غير فشرده شده توسط DSP بلافاصله توسط مبدل ديجيتال به آنالوگ (DAC)

پردازش و يك سيگنال آنالوگ ايجاد مي‌گردد. سيگنال‌هاي فوق از طريق هدفون و يا بلندگو قابل شنيدن خواهد بود.

### *اصول طراحي كارت صوت ساخته شده در اين پروژه*

شايد به نوعي بهتر باشد نام اين وسيله كه در پروژه ساخته شده را دستگاه ضبط، ذخيره و پخش صدا از روي كامپيوتر، بناميم. چرا كه در حال حاضر اين دستگاه مي‌تواند صدا يا داده آنالوگ ما را به صورت فايلي در كامپيوتر ذخيره كرده و بعد از آن نيز مي‌تواند همان فايل ذخيره شده را به صورت صدا پخش نمايد.

ولي آنچه موجب شد كه من به جرات بتوانم اين وسيله را كارت صوت بنامم اين است كه اين دستگاه مي‌تواند يك كارت صوت كامل باشد. امروزه در بسياري از پايان‌نامه‌ها ، پروژه‌ها يا حتي مقالات سبك بر اين است كه علاوه بر بيان ايده اصلي نگارنده، ايده‌هايي بالقوه نيز براي ادامه كار ارائه مي‌شود و ايده من نيز همين است كه اين پروژه جاي آن را

دارد که فرد یا افراد دیگري براي تکمیل آن پروژه ها تعریف کنند.

ایده اصلي این کارت صوت آن است که طبقه ورودي یا طبقه ضبط صدا، سیگنال آنالوگ صوت را نهایتاً تبدیل به داده دیجیتال روي پورت ISA کرده و این وظیفة درایور این دستگاه است که با استفاده از امکانات CPU داده دیجیتال را پردازش کند. وظیفه طبقه خروجي نیز این است که داده دیجیتال روي پورت را به آنالوگ تبدیل کرده و نهایتاً آن را قابل پخش براي بلندگو کند. و اما روند کار:

ابتدا در طبقه ورودي میکروفون صداي ما را دریافت میکند. میکروفون استفاده شده در اینجا میکروفون خازني است که با یك مقاومت Pull-up به t5 متصل میشود.

سیگنال دریافتي از میکروفون توسط یك تقویت کننده عملیاتي تقویت میشود، که در اینجا از آپ امپ LF351

براي این تقویت استفاده میشود. در مرحله بعد موج تقویت شده وارد یك فیلتر دیجیتال بسیار کارا به نام MF10 میشود که قادر است هم با فرکانس Clock

و هم با مقاوتهاي خارجي باند عبور را تعيين كند. از مدي از فيلتر استفاده كرديم كه يك فيلتر پائين گذر است كه تا 36KHz را از خود عبور مي‌دهد. كلاك اين فيلتر را به وسيله يك تراشه كريستال اسيلاتور 1.8 MHz (١/٨ مگاهرتز) ايجاد مي‌كنيم. پس از آن مجدداً مجبوريم خروجي فيلتر را نيز تقويت كنيم كه باز هم از يك آپ امپ LF351 ديگر استفاده مي‌كنيم. اكنون به نظر مي‌رسد كه موج ما براي ورود به AD آماده باشد. اما يك شكل ديگر و آن اينكه موج دريافتي ما بين ٢- تا ٣ ولت است در صورتي كه براي ورودي A to D به موج ٠ تا ٥ ولت نياز داريم، براي حل اين شكل از يك آپ اسپ LM358 به عنوان يك جمع كننده استفاده شده و موج با ٢ ولت جمع مي‌شود، اكنون وقت آن است كه موج را وارد A to D ، ADC استفاده شده يكي از رايج‌ترين ADCهاي بازار به نام ADC 0804 است كه يك A to D ٨ بيتي است.

از آنجائيكه فيلترها فركانس موج را در 3.6 KHz قطع مي‌كرد ما نياز داريم تا از هر سيكل حداقل ٣ سمپل بگيريم، بنابراين بايد كلاك ADC را ١٠ كيلو هرتز تنظيم كنيم. ADC مورد نظر خود تامين كننده

كلاك دارد ولي ترجيحاً ما از كلاك خارجي استفاده كرديم كه توسط IC555 تامين مي‌شود. تا اينجا داده ديجيتال ما آماده است و ما پورت ISA را براي انتقال اين داده به كامپيوتر به كار مي‌گيريم. چون A to D ما ۸ بيتي است و باس ISA نيز ۸ بيت داده دارد، ۸ بيت خروجي ADC را به صورت hand Shaking به ۸ بيت داده ISA متصل مي‌كنيم.

اكنون نوبت به نرم‌افزار مي‌رسد تا داده روي پورت را خوانده. آن را در حافظه به صورت يك فايل ذخيره كند.

كار ديگر نرم‌افزار گذاشتن مجدد داده روي پورت هنگام پخش صدا است. در هنگام پخش صدا داده ۸ بيتي روي ۸ بيت ورودي DAC و (ديجيتال به آنالوگ) قرار مي‌گيرد و اين DAC كه در واقع با ADC ما همخوان است در خروجي موجب آنالوگ مي‌دهد كه ما اين موج را وارد يك LF351 كه در اين جا فقط نقش بافري دارد مي‌كنيم. سپس موج خروجي از LF351 وارد فيلتر MF10 مي‌شود. فيلتر MF10 ما نيز در اينجا همانند طبقه ورودي فركانسهاي بالاي 3-6 KHz را قطع نموده و موج خروجي به ما مي‌دهد. اينجا نيز كلاك فيلتر ما همان

۱/۸ مگاهرتز است که توسط کریستال اسیلاتور تولید می‌شود.

نگفته نماند که کلاك DAC نیز همان ۱۰ کیلوهرتز است که توسط IC555 تولید می‌شود. خروجي فیلتر نهایتاً وارد یك تقویت کننده کم نویز به نام LM380 شده و خروجي این تقویت کننده عملیاتي آماده ورود Speaker یا هدفون است.

## شرح کار قسمتهاي مختلف مدار:

### ۱ ـ میکروفون

انواع مختلفي از میکروفون در بازار موجود است که به طور کلي شامل میکروفونهاي اکتیو یا فعال و میکروفونهاي غیر فعال هستند، این مدار قابلیت کار با هر دو نوع میکروفن را دارد یك میکروفون فعال در خروجي خود ولتاژي حدود ۱۰ تا ۲۰ میلي‌ولت می‌دهد ه باید در مراحلي تقویت شود ولي میکروفونهاي غیر فعال احتیاج به بایاس یا به عبارتي نیاز به درایو کردن دارند. شکل زیر نحوه درایو یك میکروفون غیر فعال را نشان می‌دهد.

در يك ميكروفون غير فعال با تفسير مقاومت Pull – Up ميتوان ولتاژ خروجي را تا حدي تغيير داد كه با قرار دادن اين مقاومت خروجي تقريباً هان خروجي ميكروفون اكتيو است. همانطور كه گفته شد علاوه بر اين يك جك ميكروفون اكتيو نيز در مدار قرار دارد.

## تقويت كننده عملياتي:

در اين قسمت براي تقويت خروجي ميكروفون از يك آپ امپ LF351 استفاده شده است. اين آپ امپ يك تقويتكننده ارزان با سرعت بالا و ورودي JFET است.

اين آپ اسپ جريان مصرفي بسيار كمي دارد در حاليكه Slew rate آن بسيار سريع و گين پهناي باند مناسبي نيز دارد.

LF351 پين‌هاي منطبق با استاندارد LM741 را دارا است. و از يك مدار تنظيم آفست مشابه با آن استفاده مي‌كند. اين ويژگي اين قطعه به طراحان كمك مي‌كند كه به راحتي مدارهاي طراحي شده با LM741 را توسط اين قطعه Upgrade كنند.

در مدار ما اين آپ امپ وظيفه تقويت سيگنال ميكروفون را دارد و بدين شكل استفاده مي‌شود:

مقاومت ۱۰۰ اهم متصل به پايه ۳ براي حذف آفست DC است. اين مدار در واقع ولتاژ خروجي ميكروفون را به حدود يك ولت مي‌رساند. نكته قابل توجه در استفاده از اين مدار سهولت استفاده و در دسترس بودن آپ امپ مورد نظر است و جوابگوي از آن كار پيچيده‌اي نيست. پهناي باند بهره آن حدود ٤ مگاهرتز است و امپدانس خروجي آن بسيار بالا و در حدود $10^{12}$ اهم است. اكنون سيگنال خروجي از تقويت كننده آماده فيلتر شدن است.

**فيلتراسيون:**

اين بخش يكي از مهمترين قسمتهاي مدار است كه واقعاً نقشي تعيين كننده در انتخاب كليه فركانسها

دارد. IC استفاده شده در این قسمت یك فیلتر فوق‌العاده توانا به نام MF10 است كه فیلتري همه منظوره با قابلیتهاي بالاست. این IC شامل دو فیلتر كاملاً مستقل است. هر فیلتر جداگانه با كلاك خارجي و مقاومت‌هاي خارجي قابل تنظیم است.

این فیلترها قابل تبدیل به انواع بالاگذر، پائین‌گذر، میان‌گذر و ناچ فیلتر را دارد.

این فیلترها هر كدام درجه دو بوده و قابلیت تبدیل به انواع كلاسیك مانند باترورث و چبي‌شف و غیره دارد.

در این پروژه ما احتیاج به یك فیلتر پائین‌گذر داریم.

مدهاي متفاوتي وجود دارد كه مي‌توانیم از این فیلتر استفاده كنیم كه در datasheet آن در ضمیمه قایل مشاهده است در این ما از مد 6b استفاده كردیم كه یك فیلتر درجه تك قطبي است كه پائین‌گذر است و فركانس‌هاي تا 3.6KHz را عبور مي‌دهد.

در این مد محاسبه فركانس عبور به سه عامل بستگي دارد:

۱ـ فرکانس کلاك ورودي

۲ـ تغذیه با باس مدار

۳ـ مقاومتهاي $R_3, R_2$

هنگامي که منبع تغذیه‌ها ۵ ولت باشد فرکانس عبور از این فرمول محاسبه مي‌شود. $f_c = \frac{R_2}{R_3}\frac{f_{clk}}{50}$

در اینجا با توجه به اینکه $R_3 = 33k$ , $R_2 = 33k$ است. بنابراین نسبت $\frac{R_2}{R_3}$ برابر ۱۰ است و $f_c$ با استفاده از فرمول برابر 3.6 کیلوهرتز مي‌شود.

اگر تغذیه ما ۱۰ ولت باشد فرمول محاسبه بسامد عبور بدین شکل تغییر مي‌کند: $f_c = \frac{R_2}{R_3}\frac{f_{clk}}{100}$

در ابتداي این بحث گفتیم که این قسمت از مدار نقش تعیین کننده در نحوه کار قسمتهاي دیگر دارد. یکي از مصادیق این صحبت اینجاست که از آنجائیکه فرکانس عبور را 3.6 کیلوهرتز در نظر گرفتیم فرکانس Sampling مبدل آنالوگ به دیجیتال را ۱۰ کیلوهرتز در نظر مي‌گیریم که در هر سیکل حداقل ۳ سمپل از سیگنال صوت گرفته شود که انتقال سیگنال صوت به نحو احسن انجام گیرد.

ورودي کلاك این فیلتر را یك چیپ کریستال اسیلاتور تولید مي‌کند که خود با وصل شده به منبع تغذیه

قادر به توليد فركانس مربعي است. اين چيپ يك چيپ ٤ پايه است كه پايه‌بندي آن به صورت زير است:

همانطور كه ملاحظه كرديم تنها از يك فيلتر اين IC براي فيلتراسيون سيگنال صوت استفاده كرديم براي جلوگيري از نفوذ پايه‌هاي آزاد اين فيلتر را توسط مقاومت زمين كرديم و نتيجه بسيار مطلوبتري قابل مشاهده بود.

اكنون سيگنال خروجي يك سيگنال فيلتر شده با دامنه‌اي از ١ ولت است. در اينجا مجدداً از يك تقويت كننده عملياتي LF351 مشابه آنچه در ابتدا استفاده شده بهره مي‌بريم تا دامنه سيگنال را به حدود ٥ ولت پيك تا پيك برسانيم. مبدل آنالوگ به ديجيتال : APCDtlo

## مبدل آنالوگ به ديجيتال

مبدل آنالوگ به ديجيتال استفاده شده در اين مدار ADCD804 است كه مبدلي با كارايي بسيار عالي است. اين مبدل با ولتاژ تغذيه 5 ولت كار مي‌كند و فركانس نمونه‌برداري آن تا ۱۰ كيلوهرتز قابل افزايش است يعني قادر است در هر صد ميكروثانيه يك نمونه‌برداري تند كه با توجه به اينكه فيلترها حداكثر سيگنالهاي صوتي با فركانس ۳/٦ كيلوهرتز را عبور مي‌دهد بنابراين در بدترين حالت مبدل آنالوگ به ديجيتال، قادر است در هر سيكل سه نمونه از سيگنال صوت بردارد.

اين مبدل يك مبدل ۸ بيتي است و بنابراين پيك تو پيك ۰ تا ٥ وات را به ۲۸ قسمت تبديل مي‌كند بنابراين براي يك سيگنال صوتي مبدل مناسبي است. اين مبدل علاوه بر ۸ بيت ديتاي ديجيتال پايه‌هاي $\overline{CS}, \overline{RD}, \overline{WR}, \overline{Clkin}, \overline{ClkR}, \overline{INT}, \overline{V^{+}cc}, \overline{V_{rf/2}}$ است، در اين جا ما از حالت free running اين مبدل استفاده مي‌كنيم.

پايه‌هاي ۳ و ۵ كد $\overline{INT}, \overline{RD}$ هستند را به هم بسته و با يك كليد بين آنها و زمين در واقع $\overline{Enable}$ مدار را مي‌سازيم زيرا اين در پايه Active low هستند.

اين مبدل هم مي‌تواند از كلاك خارجي استفاده كند و هم خود داراي كلاك ژنراتور است كه فركانس كلاك دروني آن قابل تنظيم به وسيله خازن و مقاومتي است كه بين پايه‌هاي ٤ و ۱۹ يعني $\overline{CLKIN}, \overline{CLKR}$ به كار مي‌رود و رابطه آن بدين ترتيب است: $F=\frac{1}{1.1RC}$ كه ما با استفاده از خازن ۱۵۰ پيكوفارا در مقاومت ۱۰ كيلوام فركانسي حدود ۱۰ كيلوهرتز ايجاد مي‌كنيم.

نكته‌اي كه در اينجا قابل ذكر است اين است كه تا قبل از رسيدن به ورودي مبدل ما سيگنالي متقارن يعني از حدود ۲- تا ۲+ داشتيم ولي مبدل ما تنها قادر است بين صفر تا ۵ را تبديل كند. براي اين موضوع قبل از ADC ما از يك level shiffer استفاده كرديم.

**ـ آپ امپ LM741 به عنوان level shiffer:**

همانطور كه گفته شد براي ورودي مبدل آنالوگ به ديجيتال نياز به سيگنالي بين صفر تا ۵ داشتيم در

حالیکه سیگنال خروجي از فیلتر بین ۲- تا ۲+ بود بنابراین ما یك شیفت DC به وسیله یك آپ امپ LM741 که به صورت زیر در مدار قرار گرفت داریم.

در اینجا اندکي هم بهره ولتاژ نیاز داشتیم که همانطور که مشاهده مي‌شود در فیدبك به جاي مقاومت یك کیلواهم از مقاومت ۱۰ کیلو اهم استفاده شد.

**رگولاتور 7805:**

در مدار، تقویت کننده‌ها با ولتاژ متقارن ۱۲± ولت کار مي‌کنند در حالیکه ADC نیاز به ۵+ ولت دارد و همانطور که دیدیم فیلتر و level shiffer نیز نیاز به ۵± دارند بنابراین از یك رگولاتور به نام

7805 استفاده شد که این رگولاتور یك IC سه پایه است.

این IC قادر است تا دامنه ۱۸ ولت را در ورودي بگیرد و خروجي ٥ ولت دقیق بدهد.

**مباحث نرم افزاري:**

پروژه انجام شده را به نوعي ميتوان يك مدار واسط خواند كه قادر است ارتباطي بين رايانه و محيط بيرون ايجاد كند. امروزه يكي از مفيدترين شاخه هاي الكترونيك مدارهاي واسط است. زيرا كامپيوتر وسيله‌اي ارزان قيمت است كه قابليت انجام خدمات بسيار پيچيده كه قبلاً توسط سخت‌افزارهاي گران قيمت انجام ميشد را به كمك نرم افزار و مدار واسط دارد.

براي برقرار كردن ارتباط اين مدار با كامپيوتر نخست برنامه‌اي تحت Dos به وسيله زبان برنامه نويسي C++ نوشتيم. اين برنامه وظيفه خواندن ديتا از روي باس ديتاي اسلات ISA دارد.

نحوه كار اين برنامه بدين شكل است كه آدرس خاص باس ديتاي اسلات ISA را از ما گرفته و آن را روي فايلي كه خود ايجاد كرده‌ايم ميريزد ولي كار بدين سادگي‌ها هم نيست.

همانطور كه از بخشهاي گذشته به خاطر داريم، مبدل آنالوگ به ديجيتال ما با فركانس ۱۰

كيلوهرتز كار مي‌كرد يعني در واقع هر صد ميكرو ثانيه داده هاي لچ شده روي باس ديتاي مبدل تغيير مي‌كند. بنابراين مجبوريم برنامه خود را طوري تنظيم كنيم كه هر صد ميكرو ثانيه از روي باس نمونه‌برداري كند. مشكل اصلي اينجاست كه تقسيم‌بندي‌هاي زماني در زبان برنامه نويسي در رنج ميلي‌ثانيه است.

براي بردن آن در رنج ميكرو ثانيه يك راه اين است كه از Interrupt بايا‌س كامپيوتر استفاده كنيم. بايا‌س كامپيوتر تراشه‌اي است كه عمده وظيفه آن هنگام راه‌اندازي كامپيوتر است و پس از انها با وجود توانايي بسيار زياد كار خاصي انجام نمي دهد.

پس يك راه نوسان دادن به دستور خواندن با فركانس ۱۰ كيلو هرتز استفاده از بايا‌س كامپيوتر است.

دومين راه استفاده از يك سري دستورات زبان ماشين است كه در C ++ قابل اجراست يكي از اين دستورات، دستور Sleep است كه قادر است تا

میکروثانیه وقفه ایجاد کند. نحوه نوشتن این دستور بدین شکل است:

Public Static Vaid Sleep (lpng milis , int naves) این دستور در مواردي قادر است در حد نانو ثانیه نیز وقفه ایجاد کند. نهایتاً این برنامه دیتاي ما را وارد فایلي به نام Voice.TXT مي کند که آن را در درایو C ذخیره میکند.

برنامه ما برنامه ایست که این فایل نمونهبرداري شده را تبدیل به فایل Wav کند.

این نرم افزار توسط زبان برنامه نویس Visual Basic نوشته شده که فایل Volce.TXT ما را گرفته و آن را تبدیل به فایل Wave میکند.

اصول کار این برنامه بدین صورت است که از آنجاییکه ما فرکانس Sampling کار خود را میدانیم و میدانیم که دادههاي دیجیتال با چه سرعتي در فایل نمونه ذخیره شده است میتوانیم این عدد را براي کامپیوتر تعریف کنیم و علاوه بر آن این برنامه Header File خاص فرمت Wav را نیز براي این فایل تعریف میکند. نهایتاً خروجي ما یك فایل Wav قابل شناسایي براي کامپیوتر است.

علاوه بر اين دو برنامه، برنامه ديگري نيز توسط ويژوال بيسيك نوشته شده كه وظيفه جمع‌بندي و قابل استفاده كردن اين دو برنامه را دارد.

اين برنامه علاوه بر داشتن يك اينترفيس زيبا مي‌تواند تك تك برنامه‌هاي قبل را احضار كرده و در نتيجه به رايانه دستور Record يا decode را بدهد.

همكنون پس از decode كردن فايل ما قابل اجراست و امكان تغيير به فرمتهايي نظير real audio , mP3 و ... را نيز دارد.