**بسم تعالی**



**VHDL\_Report2**

**نویسنده:**

امیرحسین احمدی\_401611328

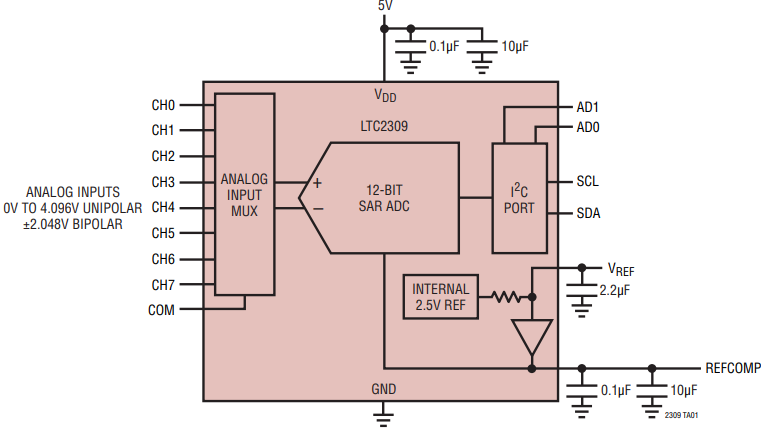
**نام استاد:**

دکتر ستار میرزا کوچکی

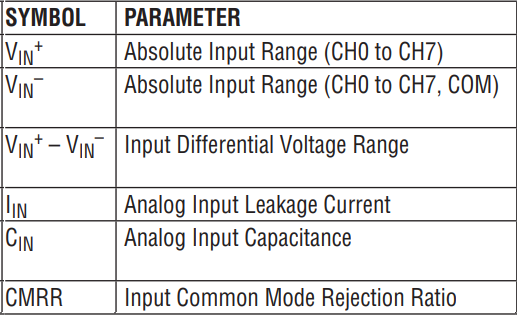
**تاریخ:**

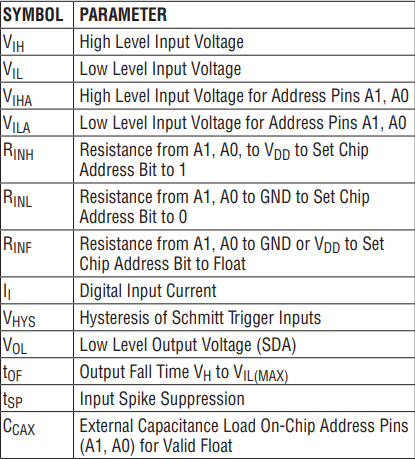
1401/11/04

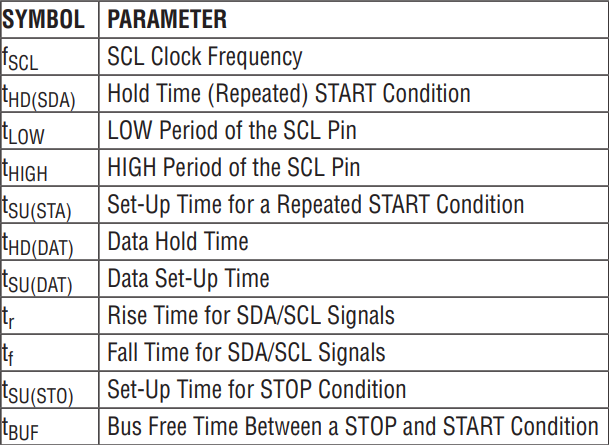
**LTC2309 a 8-channels,12-Bit SAR ADC with I2C Interface:**

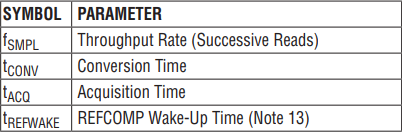
LTC2309 یک ADC توان پایین با نویز کم و هشت کاناله 12بیتی با یک واسط سریال دوسیمه I2C سازگار با 9+1 آدرس برای سنکرون­کردن است. دارای یک مرجع داخلی و یک مدار تمام تفاضلی برای کاهش نویز مشترک(Common) هست. و از کلاک داخلی برای رسیدن به فرکانس 769kHz (1.3us) استفاده می­کند. دارای مالتی­پلکسر(Mux) هشت کاناله می­باشد. همچنین در کنترل پردازش صنعتی، کنترل موتور، مانیتورینگ منبع تغذیه و شتاب­سنج و دسترسی به داده­های ایزوله و از راه دور کاربرد دارد.

بلوک دیاگرام LTC2309

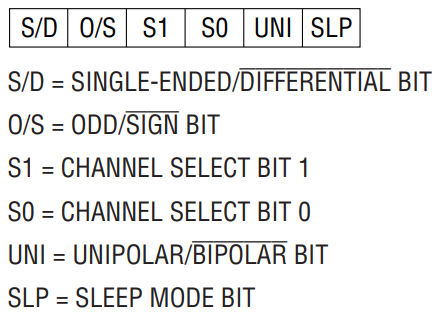
**پین­های ورودی:**

**ورودی و خروجی­های دیجیتالی I2C:**

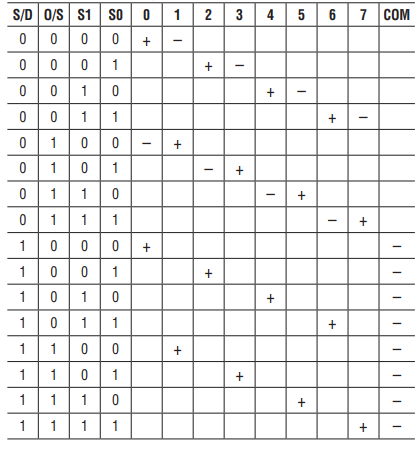
**ویژگی­های زمانی I2C:**

**ویژگی­های زمانی ADC:**

در ارتباط دو سیمه I2C تبدیل­ها با سیگنال­کردن یک شرط توقف(Stop) پس­از این­که بخش با موفقیت برای یک عملیات خواندن/ نوشتن آدرس­دهی شد، آغاز می­شوند. دستگاه تا زمانی که تبدیل به­پایان نرسد، درخواست خارجی را تایید(NACK) نخواهد کرد. وقتی برای حالت خواندن آدرس­دهی شود، دستگاه نتیجه تغییر را تحت کنترل کلاک سریال(SCL) روی خروجی قرار می­دهد که 12 بیت داده خروجی و پس از آن 4 صفر متوالی قرار دارد. با لبه پایین رونده SCL دیتا به­­روز می­شود که به کاربر اجازه می­دهد تا داده­ها را به­طور قابل اطمینانی روی لبه بالارونده SCL قفل کند. عملیات نوشتن ممکن است با استفاده­از یک START تکراری یا یک شرط STOP برای شروع یک تبدیل جدید انجام شود. با انتخاب حالت نوشتن، ADC از طریق Din 6-Bits پروگرم می­شود. ( مالتی­پلکسر از طریق Din  پیکر­بندی می­شود.)

دیتای ورودی SDI در لبه بالارونده کلاک در حین عمل خواندن بارگیری می­شود که S/D بیت اول آن و SLP بیت آخر آن در شش لبه بالارونده بعدی بارگیری می­شوند.

**ورودی آنالوگ Mux:**

ورودی آنالوگ از طریق بیت­های O/S، S1 و S0 از Din پروگرم می­شود. که مطابق جدول زیر است.

**کلاک تبدیل داخلی:**

یک کلاک پیش­فرض کارخانه­ای است که به زمان تبدیل tCONV  به میزان 1.3us تا ماکزیمم زمان 1.8us دست یابیم.

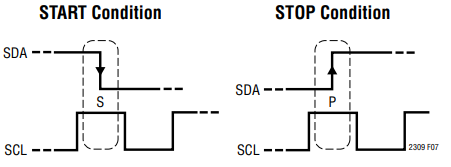
**واسط I2C:**

LTC2309 با واسط I2C ارتباط برقرار می­کند که یک واسط دو سیمه با اتصال Open-Drain است که می­تواند چند دستگاه و چند Master را روی یک باس پشتیبانی کند. دستگاه متصل فقط می­تواند SDA را در حال LOW قرار دهد و هیچوقت HIGH نمی­شود و SDA از طریق مقاومت Pull-Up به تغذیه متصل می­شود.

دیتا از روی باس I2C از نرخ استاندارد 100kBits/sec تا 400kBits/sec در حالت سریع منتقل می­شود.

LTC2309 فقط می­تواند به عنوان Slave آدرس­دهی شود و می­تواند DIN را دریافت کند یا نتیجه آخرین تغییرات را انتقال دهد. SCL به­عنوان ورودی به LTC2309 داده شده و SDA دوطرفه است.

**شرایط Stop/Start:**

حالت Start وقتی اتفاق می­افتد که SDA در لیه پایین­رونده باشد وقتی SCL در حالت HIGH هست و در این حالت باس در وضعیت BUSY قرار می­گیرد. بعد از انتقال داده حالت Stop وقتی اتفاق می­افتد که SDA در لبه بالارونده باشد و SCL مجددا در حالت HIGH باشد و در این حالت باس به وضعیت FREE می­رود. ( وضعیت Stop/Start توسط Master تغییر می­کند.) اگر باس در حال استفاده باشد و وضعیت شروع مکرر (SR) به­جای وضعیت توقف اتفاق بیافتد، باس همچنان در وضعیت BUSY باقی می­ماند. زمان شروع مکرر تبعا برابر وضعیت شروع است و برای خواندن/نوشتن از دیوایس قبل از شروع رخداد جدید است.

**انتقال دیتا:**

بعد از وضعیت Start، باس I2C در حالت Busy بوده و انتقال دیتا بین Master و Slave مذکور انجام می­شود. دیتا از طریق گروه ­های 9-Bits منتقل می­شود که بیت نهم بیت AKC می­باشد و Master خط SDA را در نه کلاک بعدی آزاد می­کند و Slave می­تواند به ACK با قرار دادن SDA در وضعیت LOW پاسخ دهد یا درغیر این صورت به NACK با قرار دادن SDA در وضعیت High-Impedance پاسخ دهد. (مقاومت خارجی خط را در وضعیت HIGH نگه می­دارد.) انتقال دیتا درصورتی اتفاق می­افتد که SCL در وضیعت LOW باشد.

**قالب داده:**

بعد از وضعیت Start، Master يك آدرس 7-Bits به همراه بيت(R/W) ارسال مي­كند (خواندن در وضعيت 1 و نوشتن در وضعيت 0). حال اگر اين آدرس 7بيتي به يكي از 9 پایه انتخابی LTC2309 مطابقت داشته باشد، ADC انتخاب می­شود. اگر ADC حین تغییر آدرس­دهی شود، درخواست R/W را پاسخ نمی­دهد و

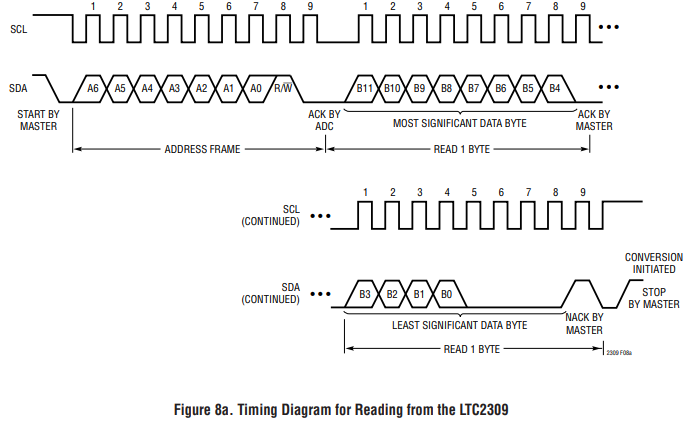
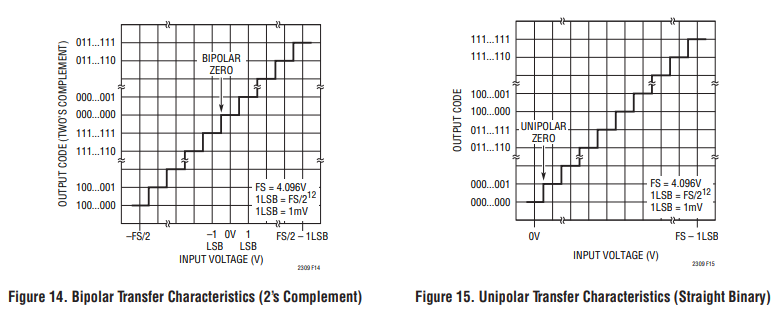
NACK را با قرار­دادن SDA در وضعیت HIGH صادر می­کند. وقتی تبدیل انجام شد، LTC2309 با قرار­دادن SDA در وضعیت LOW، ACK را صادر ­می­کند.

LTC2309 دارای دو رجیستر است؛ رجیستر خروجی 12-Bits شامل آخرین تبدیل است و رجیستر ورودی 6-Bits وضعیت عملکرد ADC و ورودی مالتی­پلکسر را پیکربندی می­کند.

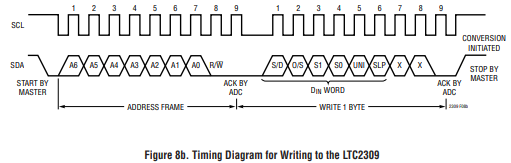
**قالب داده خروجی:**

رجیستر خروجی شامل آخرین تبدیل است و بعد از هر تبدیل، دستگاه به­صورت خودکار به وضعیت Nap یا Sleep Mode بسته­به وضعیت بیت SLP می­رود.

وقتی LTC2309 براي حالت خواندن آدرس­دهي مي­شود، SDA را در وضعيت LOW قرار­داده و به­عنوان فرستنده عمل مي­كند و Master به­عنوان گيرنده مي­تواند تا 2-Bytes از LTC2309 را بخواند. پس از عمل خواندن باید وضعیت Stop صورت بگیرد تا تبدیل جدید شروع شود. درحالی­که تبدیل انجام می­شود، دستگاه عملیات خواندن بعدی را لغو خواهد کرد.

جریان داده خروجی 16-Bits طول داشته و در لبه پایین­رونده SCL جابه­جا می­شود. بیت اول MSB و بیت دوازدهم LSB نتیجه تبدیل خواهد­بود و چهار بیت باقی­مانده صفر هستند؛ در دو حالت دوقطبی و تک­قطبی هم استفاده می­­شود. داده خروجی در وضعیت مکمل 2 برای خواندن دوقطبی و در حالت مستقیم برای خواندن تک­قطبی روی SDA قرار می­گیرد.

**قالب داده ورودی:**

وقتی LTC2309 برای حالت نوشتن آدرس­دهی می­شود، SDA را در وضعیت LOW در زمان پریود LOW قبل از نهمین سیکل قرار­داده و به­عنوان گیرنده عمل می­کند و Master به­عنوان فرستنده 1-Byte را برای پروگرم­کردن ارسال می­کند که 6-Bits آن همان DIN و دو بیت دیگر Don’t care(X) هستند. بیت­های ورودی در لبه بالا­رونده SCL در زمان عمل نوشتن قفل می­شوند

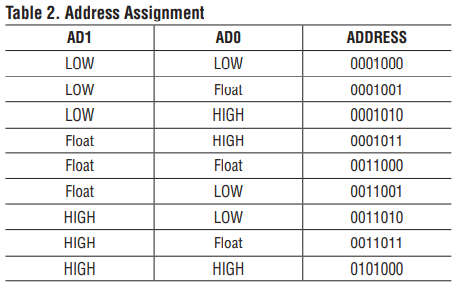
بعد از اعمال، ADC یک سیکل ریست داخلی را شروع می­کند که تمام بیت­های DIN را صفر می­کند. اگر وضعیت پیش‌فرض پیکربندی ADC's مطلوب نباشد، عملیات نوشتن ممکن است انجام شود؛ درغیر­این­صورت ADC باید به­درستی آدرس­دهی شود و به­دنبال آن وضعیت Stop تا تبدیل جدید شروع شود.

**شروع تبدیل جدید:**

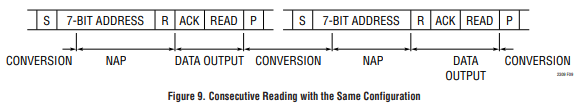
LTC2309 از وضعیت Nap/Sleep موقعی تغییر وضعیت می­دهد که به حالت خواندن یا نوشتن آدرس­دهی شود. یک فرمان توقف ممکن است پس­از اجرای عملیات خواندن/نوشتن برای تحریک تبدیل جدید صادر شود. صدور فرمان توقف بعد از هشتمین کلاک SCL از قالب آدرس و قبل از کامل شدن عمل خواندن/نوشتن می­تواند منجر­به شروع تبدیل جدید شود؛ اما خروجی به­دلیل فقدان زمان کافی معتبر نخواهد بود.

**آدرسLTC2309:**

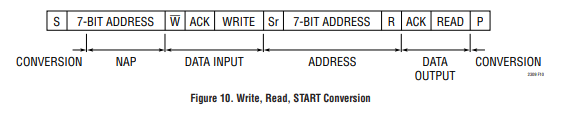
LTC2309 دارای دو پایه آدرس AD0 و AD1 می­باشد که می­تواند به حالت LOW/HIGH یا شناور(Float) محدود شود تا یکی از 9 آدرس ممکن را فعال کند.

علاوه­بر این پیکربندی، LTC2309 یک آدرس کلی (1101011) هم دارد که برای سنکرون­کردن چندین LTC2309s یا بقیه خانواده آن مثل I2C LTC230X SAR ADCs باشد.

**خواندن متوالی:**

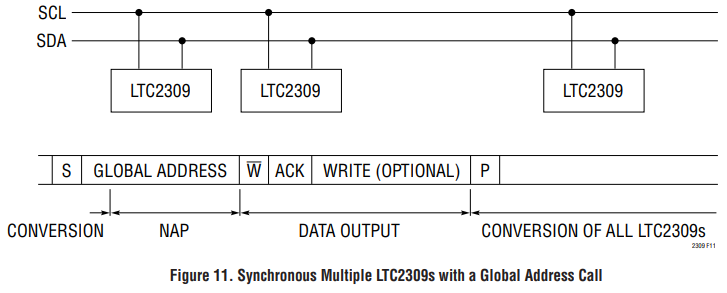
در کاربردهایی که کانال ورودی مشابه هر سیکل نمونه­برداری می­شود، تبدیل می­تواند به­طور حتم اجرا شود و بدون سیکل نوشتن، بخواند. کلمه DIN از مقدار قبلی بدون تغییر باقی می­ماند. اگر از ابتدا مقداری برای DIN تعیین نشده­باشد، مقدار پیش­فرض 0 برای همه بیت­های آن درنظر گرفته­می­شود. در پایان فرآیند خواندن، باید شرایط توقف برقرار شود تا تبدیل بعدی شروع شود. در نتیجه سیکل تبدیل، نتیجه بعدی را می‌توان با استفاده از روش شرح‌داده‌شده خواند. اگر چرخه تبدیل به پایان نرسد و یک آدرس معتبر دستگاه را انتخاب کند، LTC2309 یک سیگنال لغو NACK ارسال می­کند که نشان ­دهد که سیکل تبدیل درحال انجام است.

**خواندن/نوشتن متوالی:**

زمانی­که سیکل تبدیل کامل شد، LTC2309 می­تواند با استفاده­از فرمان شروع مکرر(Sr) نوشته و سپس خوانده­شود. شکل زیر یک سیکل را نشان ­می­دهد که با نوشتن دیتا شروع می­شود، یک شروع مجدد، پس از آن خواندن با فرمان توقف به پایان رسید. پس از آن­که همه 16 بیت خوانده­شد، یک تبدیل می­تواند با صادرکردن یک فرمان توقف شروع شود. تبدیل بعدی با استفاده از داده‌های جدید برنامه‌ریزی‌شده انجام خواهد شد.

**سنکرون­کردن چندین LTC2309با آدرس کلی:**

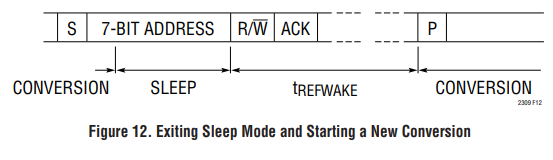
در کاربرد­هایی که چندین LTC2309 یا I2C SAR ADCs از شرکت فن‌آوری خطی در یک باس مشترک I2C استفاده می­شوند، تمام مبدل­ها می­توانند با هم از طریق آدرس کلی(Global address) هم­زمان شوند( البته تمام مبدل­ها باید سیکل تبدیل کامل­شده داشته باشند). Master فرمان شروع را صادر می­کند، که به دنبال آن آدرس کلی 1101011 هست و پس از آن درخواست نوشتن. تمام مبدل ها انتخاب­شده می­شوند؛ سپس Master یک بایت نوشتن همراه با فرمان توقف ارسال می­کند که این کار انتخاب کانال را به روز می‌کند و به طور همزمان تبدیل برای همه ADC ها روی باس آغاز می­شود.

برای همزمان کردن چندین مبدل‌ بدون تغییر کانال‌ها، فرمان توقف باید پس از تایید فرمان نوشتن کلی صادر شود. فرمان خواندن کلی مجاز نیست و مبدل‌ها یک فرمان خواندن کلی را لغو (NACK) می­کنند.

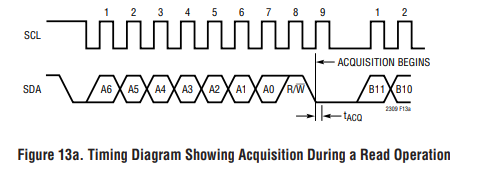
**حالت NAP:**

ADC پس از آن که یک تبدیل کامل شد(tCONV)، به حالت Nap می­روند به­شرطی­که بیت SLP صفر باشد. همچنین میزان جریان تغذیه کاهش می­یابد(210uA).

**حالت Sleep:**

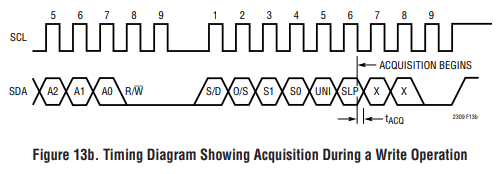
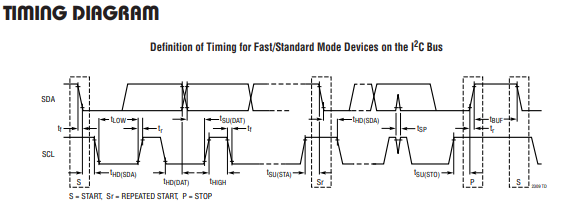
ADC پس از آن که یک تبدیل کامل شد(tCONV)، به حالت Sleep می­روند به­شرطی­که بیت SLP يك باشد. همچنین میزان جریان تغذیه به شرطی که هیچ یک از ورودی‌های دیجیتال سوئیچ نشوند، کاهش می­یابد(7uA). وقتي كه LTC2309 آدرس­دهي شد، ADC از حالت Sleep خارج شده و مدت زمان 200ms(tREFWAKE) طول مي­كشد تا به اصطلاح بيدار شود و قبل از اين زمان تبديلي صورت نمي­گيرد.

**Acquistion:**

LTC2309 شروع به دریافت سیگنال ورودی در موارد مختلف بسته به این که آیا عملیات خواندن یا نوشتن باشد، مي­كند. اگر عمل خواندن اجرا شود، دريافت سيگنال ورودي از لبه بالارونده كلاك نهم به دنبال قالب آدرس شروع مي­شود.

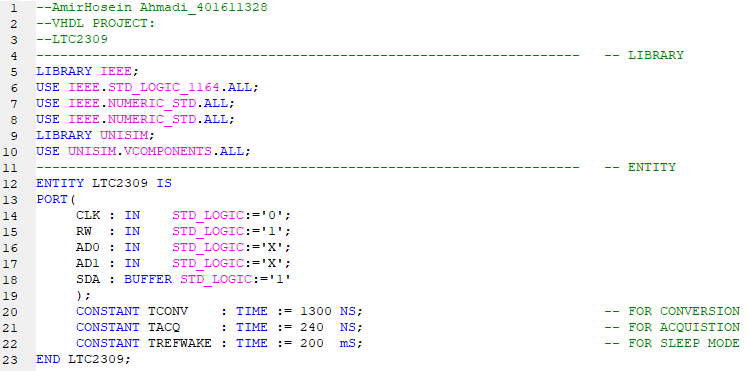
همچنين اگر عمل نوشتن اجرا شود، دريافت سيگنال ورودي از لبه پایین­رونده كلاك ششم پس از آن که کلمه DIN جا­به­جا شد، شروع می­شود.

LTC2309 سیگنال را از کانال ورودی­ای دریافت می­کند که اخیرا از طریق D­­IN ­ برنامه­ریزی­شده است.

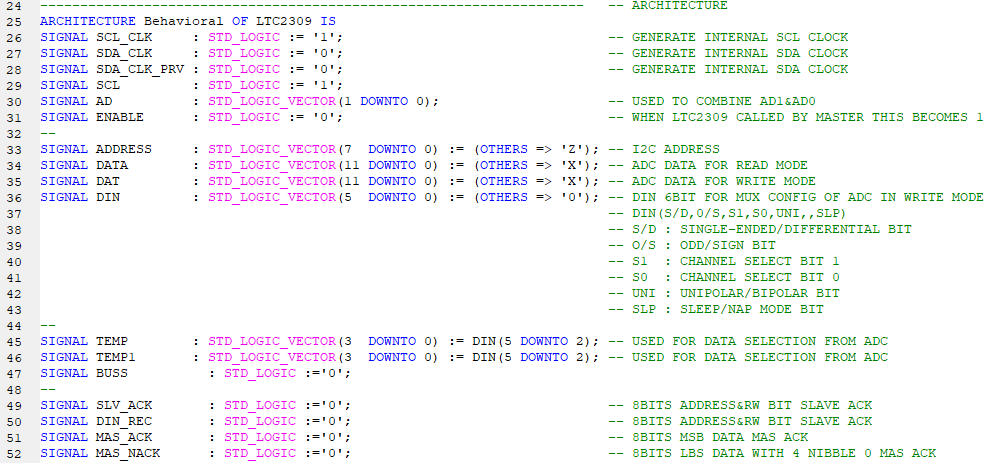
حداقل زمان 240ns نیاز است تا قبل از این که تبدیل شروع شود، سیگنال ورودی دریافت شود.

تا اینجا شکل کلی دیتاشیت LTC2309 بررسی شد. حال باید بخش­­های مختلف را شبیه­سازی کنیم.

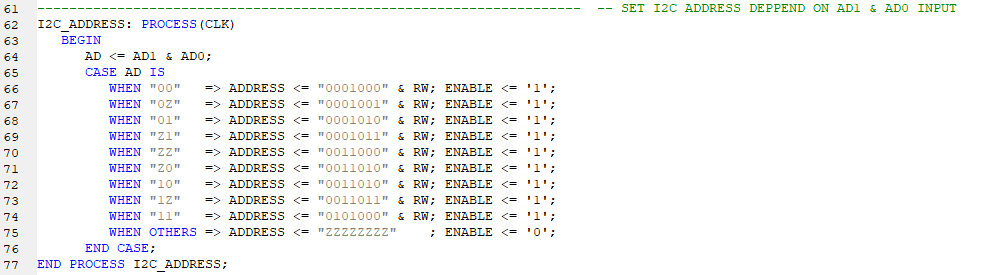
در ادامه شکل کد همراه با توضیحات آورده شده است.

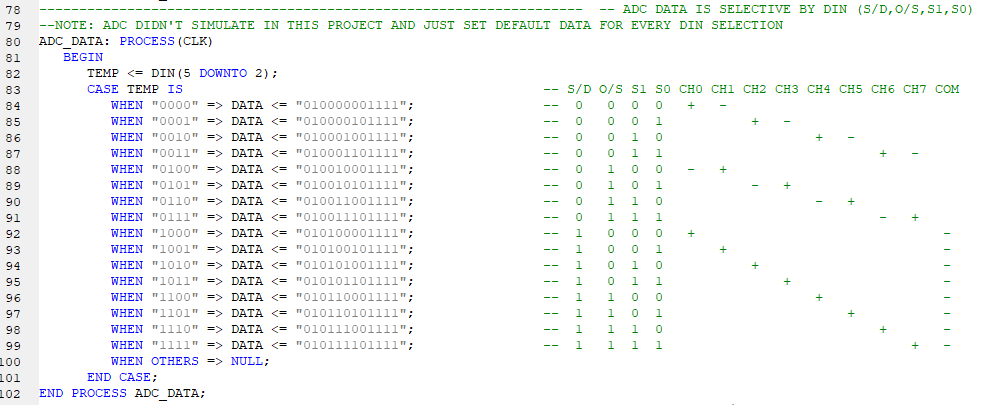
کد LTC2309 بخش دیجیتال:

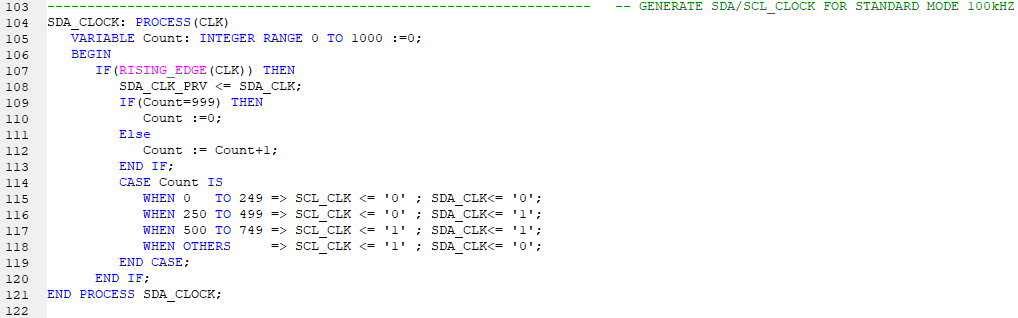
خب در تصویر بالا پین­های ورودی CLk,R/W,AD0,AD1 و SDA تعریف می­شوند. LTC2309 همواره یک SLAVE بوده و باید کلاک از MASTER به آن وارد شود؛ پورت SDA دوطرفه بوده و ما آن را یک BUFFER در نظر می­گیریم؛ وضعیت خواندن/نوشتن با R/W مشخص می­شود و آدرس­دهی از طریق AD0,AD1 انجام می­شود. در ادامه ثابت­های زمانی مد نظر تعریف شده است.

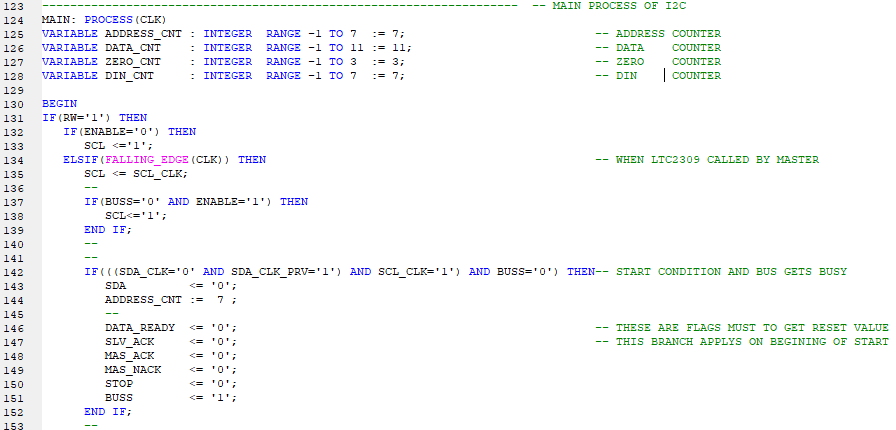


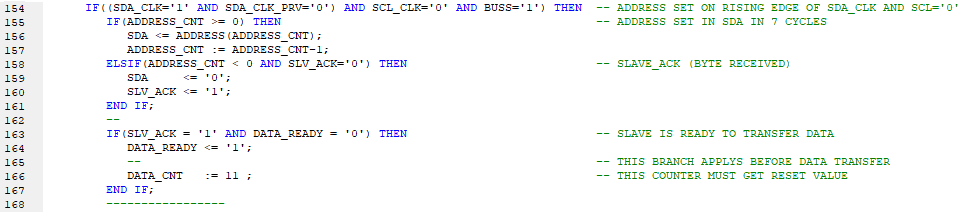
کد زیر بخش مربوط به ARCHITECTURE با سیگنال­های لازمه که توضیحات COMMAND شده است.

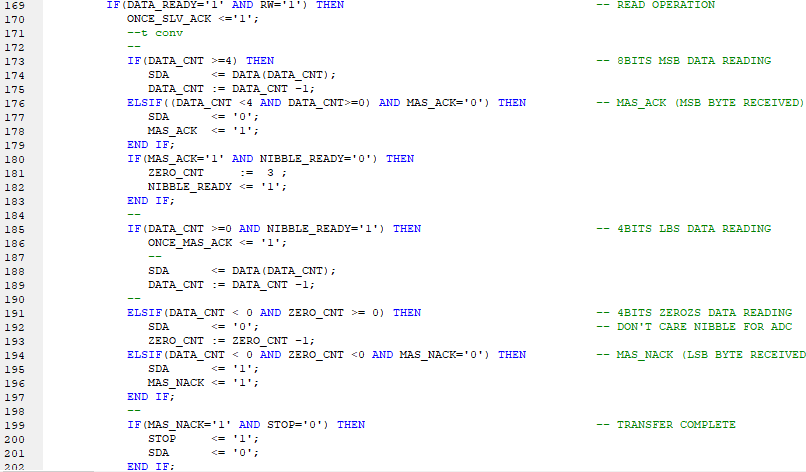
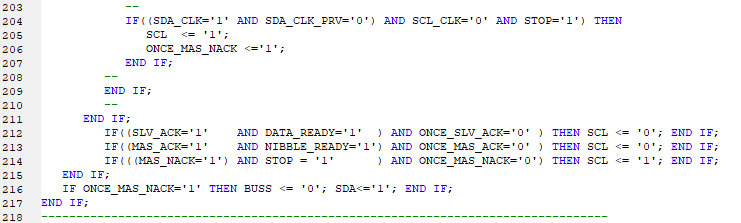
کد زیر برای بدست آوردن آدرس از روی AD0,AD1 می­باشد.(در طول ارتباط فرض می­شود آدرسی که میدهیم حتما یکی از این آدرس­ها است و دیگر شرط تطبیق آدرس­ها را چک نمی­کنیم.(فرض است ولی به­طور کلی باید چک شود.)

بخش زیر برای چک کردن دیتای ADC با توجه به وضعیت چهار پین MSB DIN که در حالت Write ورودی LTC2309 است مشخص می­­شود. ( در اینجا چون نمی­شد ورودی آنالوگ به شبیه­ساز داد، مقادیر پیش­فرض برای هر حالت در نظر گرفته شده است.

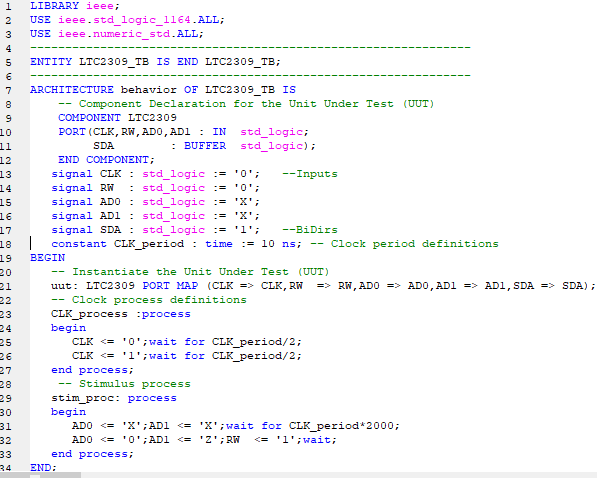
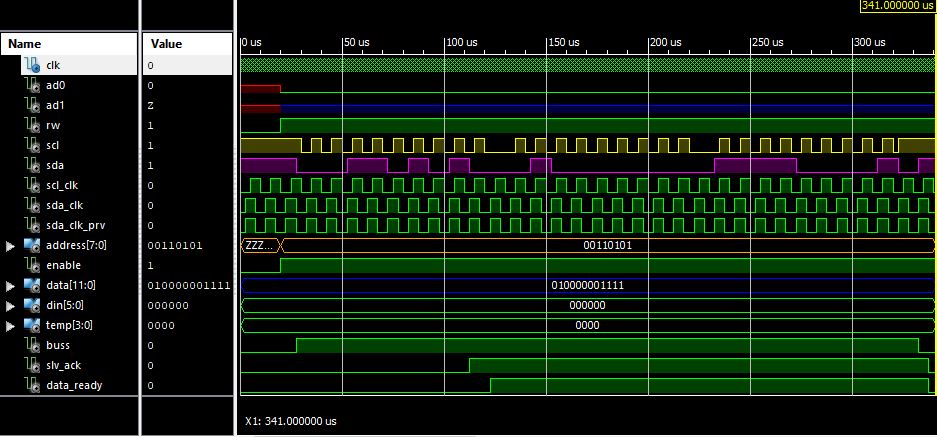
در بخش بعدی فرض بر این است فرکانس کلاک سیستم و کلاک کاری ما متفاوت است؛ لذا از یک کانتر برای تولید SCLو SDA که اختلاف فاز 90 درجه دارند استفاده می­شود. در اینجا کلاک اصلی 100MHz بوده و با یک کانتر 1000تایی آن را به مود استاندارد 100kHz می­رسانیم. در اینجا بخش استاندارد بررسی شده اما می­شد بخش fastmode را هم با یک کانتر دیگر درست کرد.

بخش زیر برای حالت READ نوشته شده است. دراین کد برای آدرس،دیتا ، نیبل صفر و ورودی DIN کانترهای مختلف تعریف شده است. در اینجا با درست بودن آدرس مقدار ENABLE=1 شده و وارد بخش START می­شود، با وارد شدن به بخش START، باس به حالت BUSY می­رود.

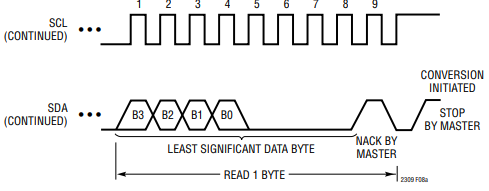
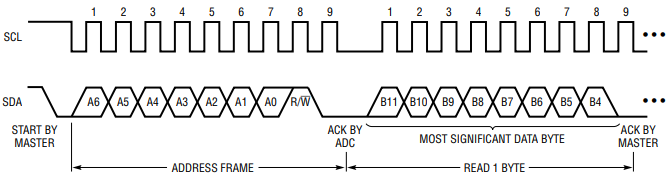
در اینجا آدرس بر روی SDA قرار می­گیرد و فرض بر این است که آدرس­ها در نهایت مساوی است و ارتباط برقرار می­شود، پس از برقراری ارتباط پرچم DATA\_READY فعال شده و آماده خواندن دیتا از LTC می­باشد(12بیت دیتای ADC به همراه نیبل صفر که در دو بایت منتقل می­شود.)

بخش زیر برای دریافت 12 بیت دیتا از ADC به مستر است، در اینجا باید در دوبایت دیتا منتقل شود، لذا برای دریافت 8بیت باارزش از نه کلاک به همراه پرچم MAS\_ACK استفاده شده و در دریافت 8 بیت بعدی، ابتدا چهار بیت کم ارزش منتقل شده و سپس چهار بیت نیبل صفر به همراه پرچم MAS\_NACK منتقل می­شود. در این حالت برای اثر گذاشتن روی SCL، پرچم­های دیگری تعریف شده اند که باعث عوض شدن SCL شود،(هر 9 سیکل باید SCL LOW، شود تا بفهمیم دیتا منقل شده است.)

در ادامه برای بررسی این کد یک تست بنچ تعریف شده که به صورت کد صفحه بعد است.

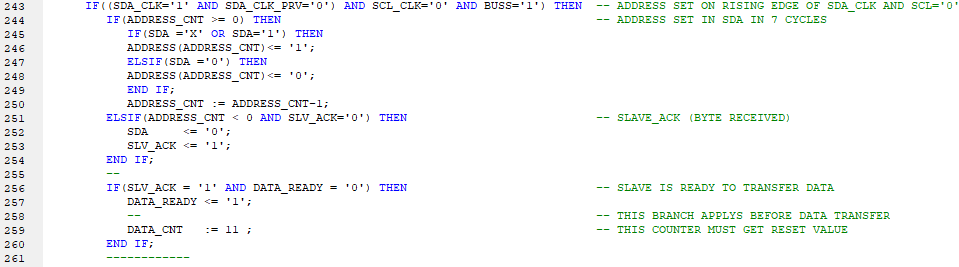
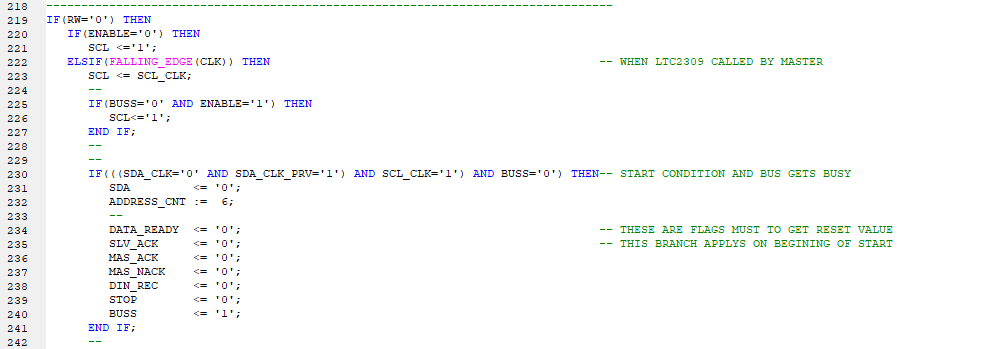
برای این تست بنج خروجی به صورت زیر در می­آید.

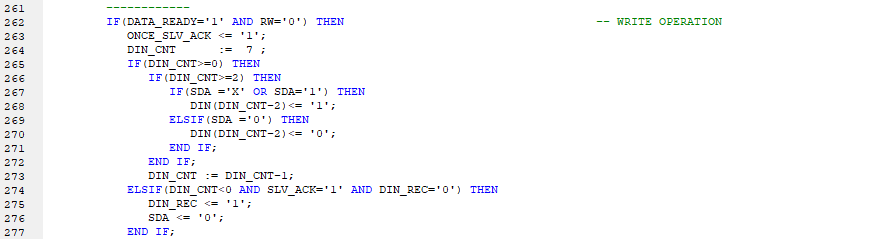
همانطور که مشاهده می­شود، شکل موج زردSCL و شکل موج بنفشSDA شکل­موج دیتاشیت است.

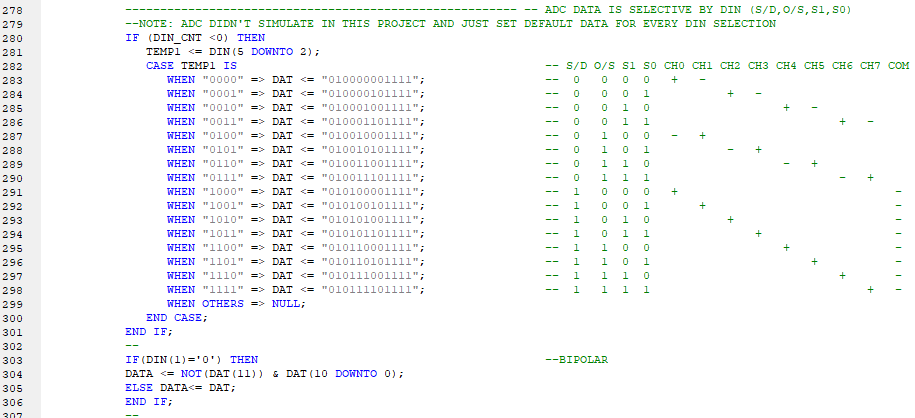
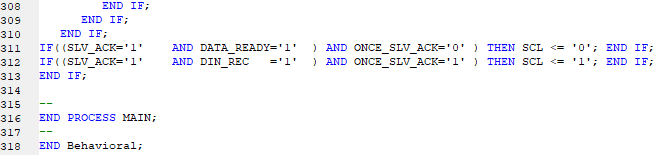


بخش زیر برای حالت WRITE نوشته شده است. بخش اول دریافت آدرس مشابه قسمت خواندن است

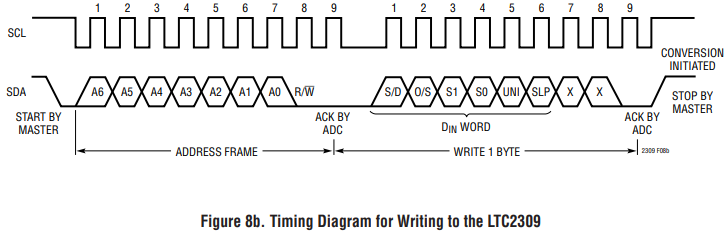
(البته ما فرض کردیم که آدرس درست فراخوانی می­شود وگرنه باید شرط تطابق آدرس فراخوانی با آدرس یکی باشد. تا مستر و اسیلو کانکت شوند.)



در بخش خواندن باید DIN شیش بیتی به همراه دو DON’T CARE در بخش داده روی SDA قرار بگیرد. با اعمال این بیت­ها به DIN می­توان داده ی ADC را تغییر داد. پس از دریافت داده پرچم DIN\_REC فعال می­شود ونشان از این است که داده دریافت شد.

در قسمت بعدی با توجه به اینکه دیتای ADC باید تغییر کند، پس باید یک CASE بنوسیم که بتوان مقدار دیتای LTC را تغییر داد. پس از مشخص شدن مقدار دیتا به کمک چهار بیت MSB باید دو بیت باقی را هم حساب کنیم، اگر داده قرار باشد به صورت BIPOLAR به شرط UNI=0 باشد، باید مقدار 2’COMPLEMENT دیتا را نگه داریم درغیر این صورت دیتا تغییر نمی­کند.

همچنین برای بیت SLP=NAP/SLEEP اگر LTC به حالت استراحت برود تا زمان TREFWAKE  نمی­توان از آن استفاده کرد و باید منتظر ماند تا این زمان تمام شود.

خروجی این بخش بنا به دلایل شبیه­سازی بدست نیامد اما الگوریتم مشابه دیتاشیت می­باشد.