دانشگاه صنعتی امیرکبیر

(پلی تکنیک تهران)

گزارش پروژه اول آزمایشگاه ریزپردازنده

سهیل داودی ۹۵۲۳۰۴۱

محمد جواد رنجبر ۹۵۲۳۰۴۸

در ابتدا با استفاده از دیتاشیت میکرو , GPIO ها به باس APB1 متصل شده اند.

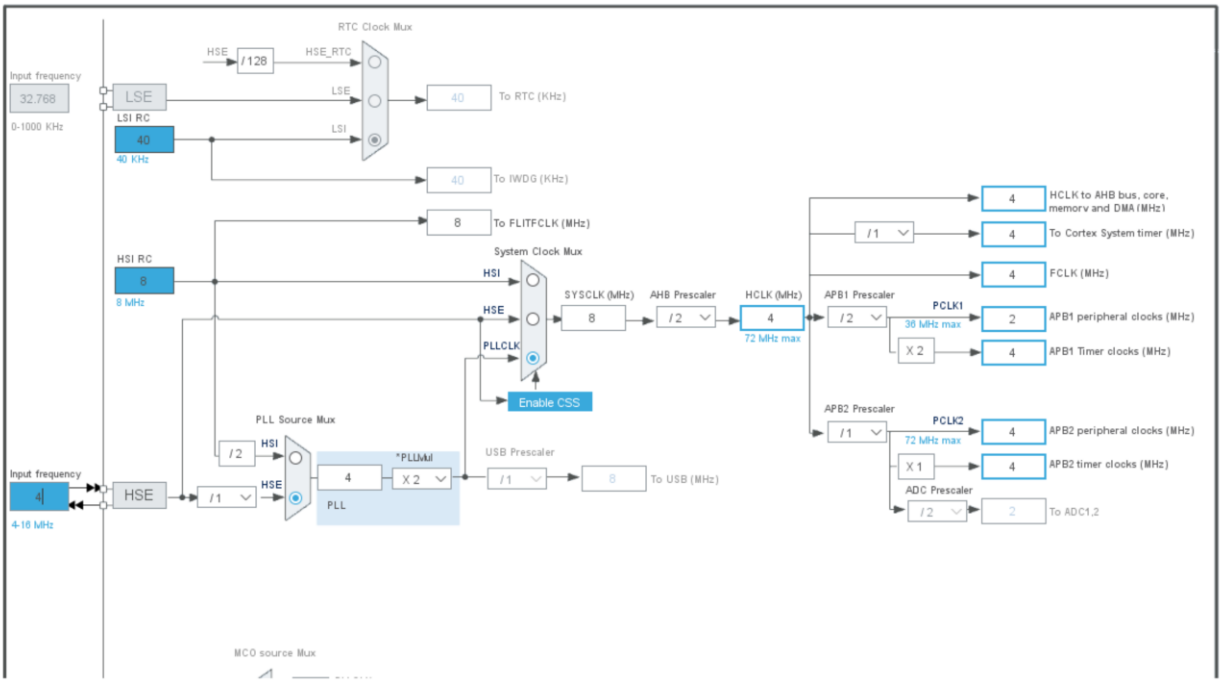
سوال۱ :اگر مقدارکلاک سیستم، کلاک باس اصلی وکالک GPIO را به ترتیب به 4MHz و 2MHz و 1MHz تغییر دهیم، چه تغییری در عملکرد سیستم ایجاد میشود؟

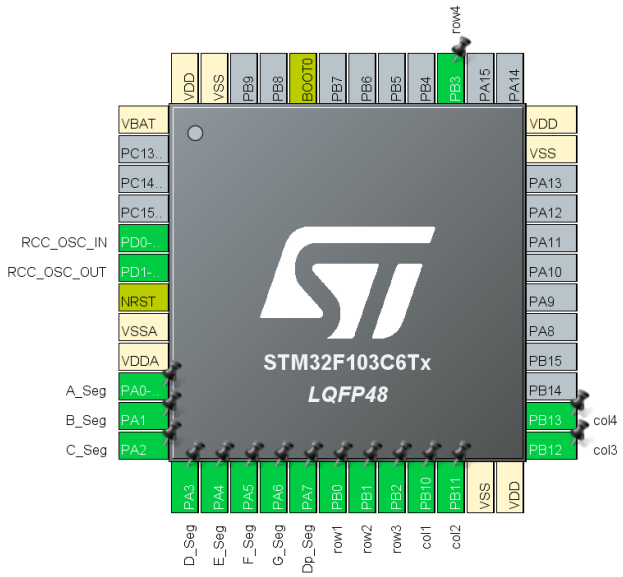
با کم شدن کلاک سیستم ، دقت تایمر ها کم می شود و همینطور سرعت سوییچینگ کمتری در gpio خواهیم داشت .

در اینجا بدلیل اینکه با استفاده از اینترآپت پروژه را پیاده سازی کردیم , پردازنده به طور دایم در حال شیفت دادن سطر ها و خواندن مقادیر ستون ها نیست و با کم کردن کلاک تغییر محسوسی اتفاق نیفتاد.

از طرفی چون فرکانس کلید زنی دست انسان خیلی کمتر از فرکانس پردازنده حتی در این حالت است , اتفاق محسوسی نمی افتد.

برای رسیدن به این فرکانس ها نیزکریستال خارجی را باید به 4MHz تغییر دهیم.





: Void shift (int i)

برای اسکن کردن کیپد ابتدا تابع void shift(int i) را تعریف می کنیم که در این تابع, یک صفر را بین پایه های سطر ها شیفت می دهیم. به عنوان مثال اگر i برابر ۲ باشد, باید سطر ها به صورت ۱۰۱۱ باشند. در ابتدای این تابع یک ارایه ۴ تایی تعریف می کنیم که حالت های مختلف سطر ها در ان بصورت هگز نوشته شده است. حال خروجی هر سطر را با and کردن یک عدد در این ارایه به دست می اوریم و با تست کردن متوجه شدیم که هر عددی به جز ۰۰۰۰ به ۱ کست می شود و برای جلوگیری از وارنینگ,این مقدار را به gpio_pinstate کست می کنیم. برای مثال اگر I = 1 باشد, سطر اول با and کردن ۰۱۱۱ با ۱۰۰۰ بدست می ایدکه به صفر تبدیل میشود. به همین صورت سطر دوم با and کردن 0111 با ۰۱۰۰ بدست می ایدکه عددی غیر از ۰ است در نتیجه به یک کست می شود.

در مرحله دوم باید تشخیص دهیم که کدام دکمه فشرده شده است.

برای جلوگیری از اینکه میکرو همیشه در حال اسکن کردن دکمه ها باشد,از اینترآپت استفاده می کنیم.به این صورت که اگر دکمه ای فشرده شود,مقدار پین صفر شده و با تنظیم کردن اینترآپت در حالت falling edge , به روتین وقفه منتقل شده و interrupt_flag را برابر یک می کنیم که نشان دهنده این است که دکمه ای فشرده شده است.

: Int Keypad_scan()

حال از تابع keypad_scan برای تشخیص دکمه فشرده شده استفاده می کنیم. به این صورت که ابتدا از تابع shift استفاده می کنیم و از متغیر col استفاده می کنیم که در حالت عادی 1- است و نشان دهنده این است دکمه ای فشرده نشده است. حال مقدار ستون ها را خوانده و هرکدام برابر صفر بودند,مقدار col را اپدیت می کنیم.در اخر نیز موقعیت دکمه فشرده شده را با توجه ستون و سطر بدست اورده و انرا بر می گردانیم تا در تابع set_segment پایه های سون سگمنت مقدار دهی شوند. ستون ها هم به ترتیب به پین های PB10 , PB11 , PB12 و PB13 متصل شده اندکه این

پین ها به صورت اینترآپت خارجی با مقاومت پول آپ داخلی و با حالت trigger falling edge تنظیم شده اندکه همه آنها روتین وقفه یکسانی دارند و دیگر نیازی به استفاده از یک گیت And برای تشخیص امدن اینتراپت نیست و هرکدام از اینها جداگانه می توانند باعث تریگر شدن اینترآپت شوند.