# CC2650LaunchPad

پروژه های نمونه اصلی :

ProjectZeroApp

این پروژه به همراه فایل ProjectZerostack بوده ولی تمام تغییراتی که در آن انجام شده مربوط به بخش projectZeroApp است. بغیر از فایل main که در زیرپوشه startup قرار دارد مهمترین فایل ProjectZero.c بوده که در پوشه Application است. تمام تعاریف اصلی مورد نیاز برای این ارتباط در این فایل قرار دارد. وضعیت LED ها توسط کاربر می تواند تغیر کند و وضعیت کلیدها در صورت فعال بودن Notification در لحظه به کاربر ارسال می شود. همچنین امکان دریافت و ارسال داده به صورت string نیز وجود دارد. خط 633 (**static** **void** **user\_processApplicationMessage**) مربوط به بخشی است که به هنگام دریافت دستور از کاربر اجرا می شود. در بخش اول Case تغییراتی برای عوض کردن پیام دریافتی انجام شده و تا زمان نوشتن گزارش امکان استفاده از پیام دریافتی برای تغییر پیام ارسالی فراهم شده است. فایل های این پروژه در مسیر C:\Users\Sadra\Desktop\Texas\Prj2 در ProjectZeroBackUp.rar ذخیره شده تا در صورت خرابی کد با اعمال UART بتوان از استفاده کرد. در فایل ProjectZero فعلی موجود بر روی CCS تغییراتی برای اضافه شدن USART انجام گرفته است.

اضافه کردن USART

بخش مربوط به ارسال پیغام داده از طریق USART در پروژه موجود بوده و خود کد متناسب با مرحله اجرای کد پیام هایی را از طریق USART ارسال می کند. با انتقال پایه ها به DIO4 و DIO5 می توان این پیام ها را مشاهده کرد. (پایه DIO3 بورد به درستی کار نمی کند) همچنین در خط 660 فایل Project.c پیام هایی برای ارسال داده دریافتی اضافه شد پیام اصلی و پیام تغییر داده شده به درستی ارسال می شد.

پس از راه اندازی USART و دریافت داده از طریق آن، نوبت به پیاده­سازی SPI می­رسد. با استفاده از نمونه های موجود SPI در حالت Slave با پایه­های DIO8، DIO9، DIO10 و DIO11 به ترتیب بعنوان MISO، MOSI، CLK و NSS فعال شد. این پورت داده را در حالت CALLBACK از میکرو دیگر دریافت کرده و در آرایه ای ذخیره می­کند.

در ساختار نوشته شده موجود برای PROJECT0 ارسال داده از دو طریق stream و string امکان پذیر است. در حالت string حداکثر می­توان 40 بایت داده از طریق ble ارسال کرد اما اگر حجم داده ارسالی بیشتر از این مقدار باشد برنامه به صورت خودکار در بخش­های 40 بایتی داده را ارسال می­کند و طبیعتا زمان ارسال طولانی تر از حالت ارسال یک پیام خواهد بود. در حالت stream حداکثر می­توان 20 بایت داده ارسال کرد و چنانچه تعداد بایت های داده ارسالی بیشتر باشد هیچ بایتی ارسال نمی شود.

برای زمان بندی ارسال اگر در حالت string باشیم در برنامه موجود فعلی دستور ارسال باید از طریق موبایل ارسال شود و نمی توان به صورت خودکار از طریق خود TI داده ها را ارسال کرد ولی در حالت stream می توان با فعال کردن notify به محض فشردن یکی از کیلدهای روی بورد TI داده را ارسال کرد.

در مجموع چنانچه دستور ارسال از طریق برنامه دیگر به صورت متوالی ارسال شود می­توان با برنامه موجود داده های دریافتی از طریق SPI را با بلوتوث ارسال کرد اما اگر قرار باشد خود TI تصمیم به ارسال بگیرد باید تغییرات زیر انجام شود:

1. راه اندازی دستور ارسال توسط تایمر بجای فشردن کلید
2. راه اندازی دستور notify برای string مشابه stream برای ارسال راحت تر داده های با حجم بالا

کد نوشته شده برای PROJECT0 دارای بخش های زیادی است که در برنامه مورد نظر ما کاربردی ندارد بنابراین شاید بهتر باشد ابتدا بخش­های اضافی این مجموعه حذف شوند.