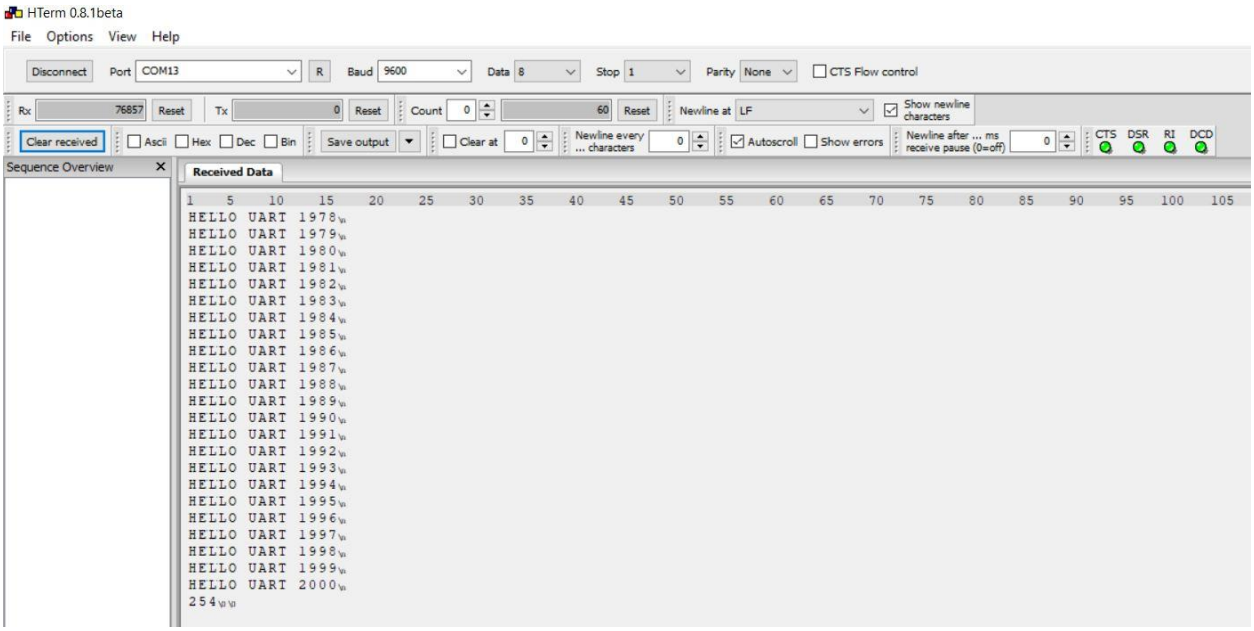
گزارش تمرین سری سوم

Workshop

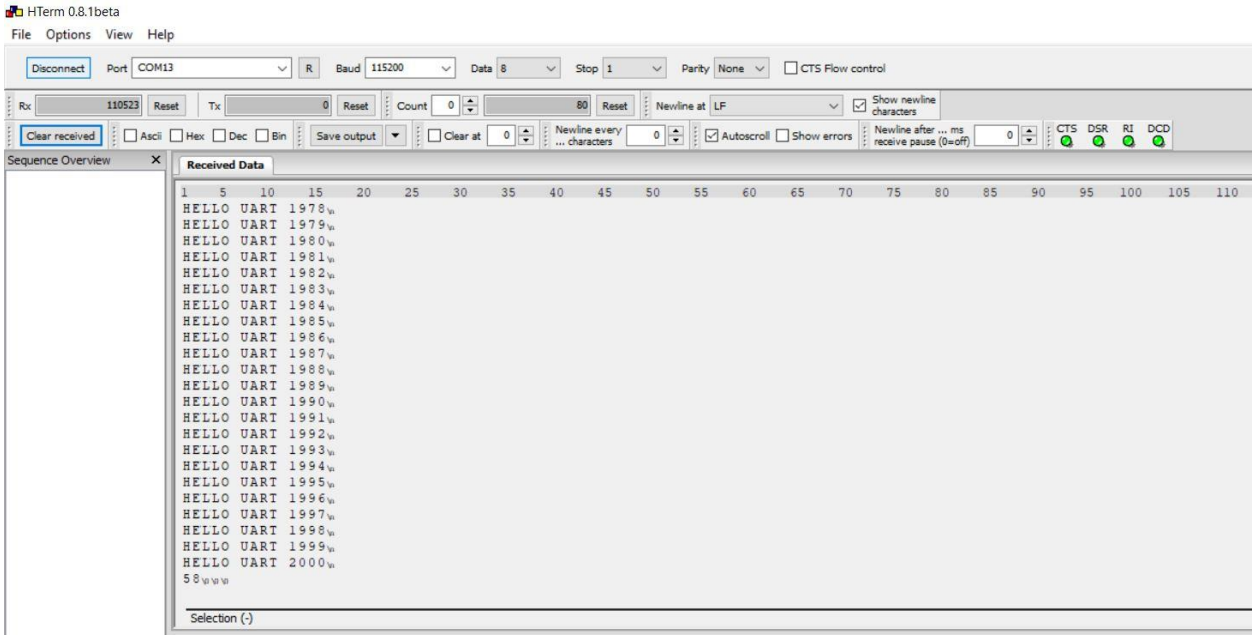
قسمت اول:یک پیام مشخص را به تعداد ۲۰۰۰ بار با سرعت های مختلف از pc به میکرو بفرستیم.baudrate همان bit per second میباشد بنابراین هر چه بالاتر باشد سرعت انتقال داده بیشتر است برای مقایسه ی زمان انتقال داده از تابع get tick() استفاده میکنیم.بدین صورت که قبل از ارسال و بعد از ارسال زمان را از این تابع دریافت کرده و اختلاف بین این دو عدد تقریبا زمان ارسال می باشد.البته نمیتوان گفت به طور دقیق این زمان زمان ارسال می باشد ولی خطایی که موجود است برای هر ۳ حالت یکی است پس مقایسه ی آن ها صحیح میباشد.

Baudrate هر چه بیشتر باشد سرعت انتقال داده بهتر است ولی هر چه بالاتر باشد نویز پذیری آن هم بیشتر است بنابراین دارای محدوده است و هر عددی نمی تواند باشد.

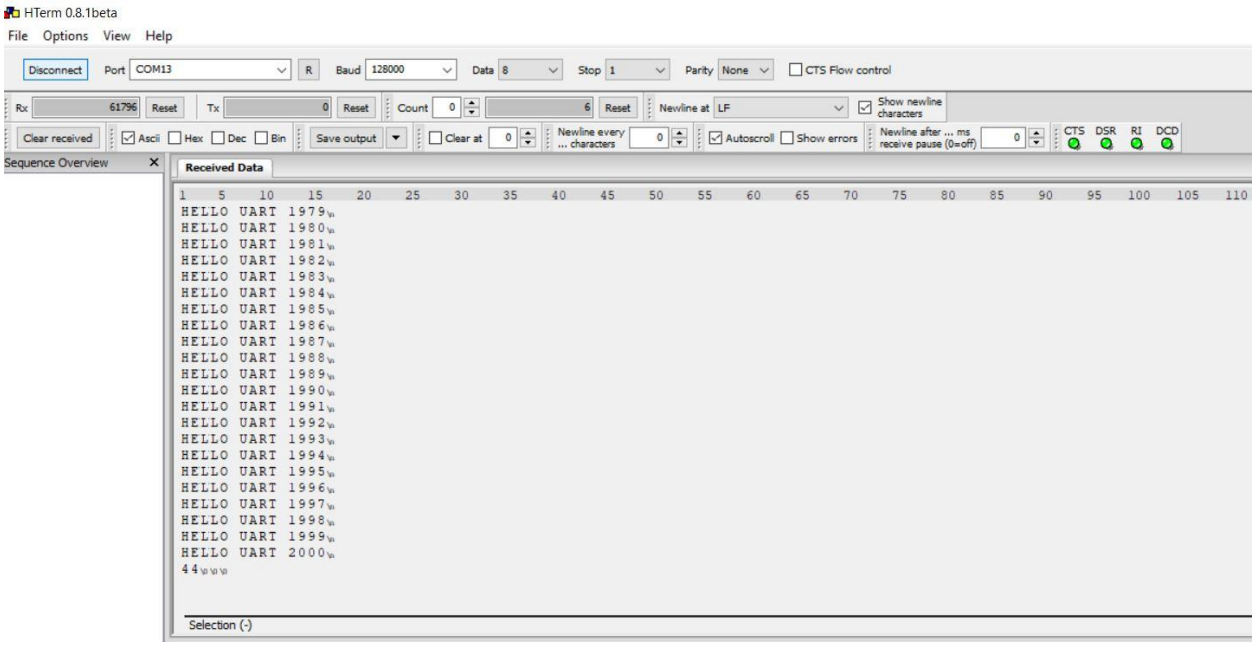
برای مثال عدد ۱۰۰۰۰۰۰ را ترمینال uart ایراد میگرفت و میگفت دارای خطاست و نمیتوان ازان استفاده کرد.و یا عدد ۲۵۶۰۰۰ را نیز وقتی برای baudrate تنظیم کردیم مشاهده ی نویز پروتکل uart شدیم و بنابراین بالاترین baudrate که میشد تنظیم کنیم ۱۲۸۰۰۰ بود.



BR=9600



BR=115200



BR=128000

قسمت دوم:در صورت سوال گفته شده که بسته ی ۸ بایتی را بین مبدا و مقصد تعریف کنیم که ۲ بایت اول و اخر ان عدد مشخص می باشد و ۴ بایت دیگر وضغیت ال ای دی ها را مشخص میکنند(میکرو F4 دارای ۴ ال ای دی می باشد)اما به دلیل اینکه برنامه را با میکرو F1 نوشتیم و این میکرو دارای یک ال ای دی است پس رشته را بین مبدا و مقصد ۵ بایتی تعریف میکنیم به نحوی که دو بایت اول و اخر ان همان فرمت و بایت وسط آن نشان دهنده ی وضعیت ال ای دی روی برد می باشد.

نکته ای که اهمیت دارد پایه ی ال ای دی open-drain می باشد و حالت اولیه آن را در high در نظر میگیریم حال هر موقع مقدار صفر روی پایه بیفتد ال ای دی روشن میشود و برعکس.

از تابع اینتراپتی در کد استفاده شده که هنگامی که دریافت ۵ بیت کامل شود وارد تابع میشود و در غیر این صورت وارد تابع اینتراپت نمی شود.

قسمت سوم:به دلیل اینکه میکروF1 فاقد دکمه ای به غیر از ریست می باشد و برای تست گرفتن روی میکرو یکی از پایه های ان را به عنوان push-button (A0)تعریف میکنیم که این پایه تابع اکسترنال اینتراپت لاین ۰ را فراخونی میکند.در حالت عادی پایه در حالت pull-up قرار دارد و هنگامی که به زمین وصل می شود مقوار ان از یک به صفر سوییچ میکند به همین دلیل اینتراپت را حساس به لبه ی پایین رونده تنظیم میکنیم.

قسمت چهارم:در این قسمت ابتدا به حالت it بایتهای ارسالی را دریافت میکنیم و سپس هنگامی که دریافت کامل شد ان را نمایش میدهیم.

پاسخ سوالات:

1.اگر در حالت polling میخواهیم ارسال کنیم اگر بیشتر از حد delay تابع ارسال گذشته باشد از تابع ارسال خارج میشود و اطلاعات از دست خواهند رفت.

2.طبق دیتاشیت با ۵ ولت

full-duplex.3

4.استفاده از پروتکل uart برای مسافت های بیشتر از چند متر و با baudrate بالا قابل اطمینان و پیاده سازی نیستند(به دلیل نویز پذیری بالا)زیرا در مسافت های طولانی اثر نویز های محیطی بیشتر می شوند و ثانیا در فرکانس های بالا تشعشع خط فرستنده روی گیرنده اثر میگذارد.برای اینکه دستگاه ها بتوانند در فاصله ی بیشتر از همدیگر قرار بگیرند و از طرسق پروتکل uart با هم ارتباط داشته باشند استانداردهایی تدوین شدند.پروتکل RS232 معمول ترین و فراوان ترین استاندارد جهت انتقال دیتا می باشد.در استاندارد RS232 در سرعت 20Kpbs بیت بر ثانیه حداکثر طول کابل قابل استفاده 7.5 متر میتواند باشد تا داده ها تقریبا به صورت صحیح به مقصد برسند.پروتکل RS422 شباهت زیادی به RS232 دارد ولی تا 10 گیرنده را پشتیبانی میکند.این پروتکل که از خطوط بالانس شده برای انتقال داده استفاده میکند اثر نویز پذیری را به شدت کاهش داده است به طوری که بیشترین فاصله ی بین گیرنده و فرشتنده در این پروتکل 1200 متر و بیشترین سرعت برابر 10 Kbps است.در پروتکل RS485 تعداد گیرنده ها می تواند حداکثر تا 32هم باشد ضمن اینکه بیشترین فاصله 1200 متر و بیشترین سرعت برابر 10 Kbps است.

دلیل تاخیر این تمرین مشکل داشتن برنامه ی CUBE اینجانب بوده همانطور که در ورک شاپ مشاهده فرمودید هنگام کامپایل دوباره ی یک برنامه error اضافه کردن library را میداد.دلیل اینکه در git تاریخ امروز ثبت شده و این تاخیر این بوده که هر بار من باید کل پزوژه را حذف میکردم و از اول دوباره پروژه میساختم.اقدام به نصب آپدیت cubeکردم اما پروسه ی زمانبری بود و کمبود وقت داشتم.

ممنون میشم توضیحات بالا رو در هنگام محاسبه ینمره مدنظر بگیرید.