

## بسمه تعالی

### پاسخ پرسش تشخیص Baudrate به صورت خودکار بدون از دست رفتن داده

برای تشخیص Baud بدون از دست رفتن داده، می توان از روش زیر استفاده نمود. با فرض 8N1.

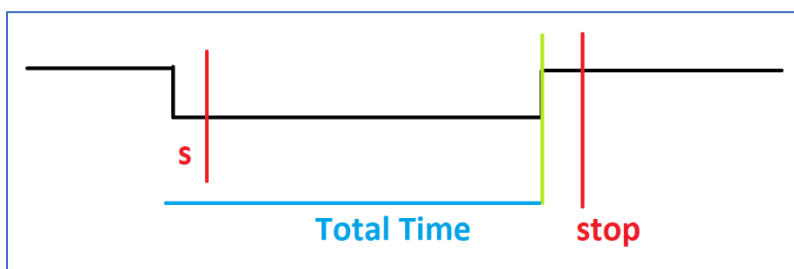
ابتدا یک تایمر در مود اندازه گیری عرض پالس (input capture) حساس به هر دو لبه فعال می کنیم. از آنجا که خط دریافت سریال، در زمانی که هیچ داده ای نمی گیرد یک است، با آمدن بیت شروع (Start Bit)، اولین شمارش با لبه ی پایین رونده شروع می شود، و با تغییر یک بیت در داده دریافتی، شمارش اندازه گیری زمان بیت (ها) قطع می شود. صرفه نظر از آن که مشخص نیست چند بیت مشابه پشت سر هم ارسال شده است، زمان اندازه گیری شده را با مقدار آن ذخیره می کنیم. این کار را تکرار می کنیم تا زمانی که زمان اندازه گیری شده از حدی که بسته به حداقل و حداکثر Baudrate قابل تشخیص تعیین شده است عبور کند. در نهایت حداکثر ۹ داده ی زمانی وجود دارد، که با جمع کردن زمان ها و تقسیم بر ۹ می توان عرض هر بیت را بدست آورد و از مدت زمان هر بیت و مقادیر متناظر با زمان ها که ذخیره شده اند، دیتا را استخراج نمود.

در صورتی که تایمر را با ضرب صحیح تنظیم کنیم، یک بایت داده برای اندازه گیری زمان ها می تواند کافی باشد و در نهایت کل کار حدود ۱۰ بایت حافظه مصرف خواهد نمود، در صورتیکه که رزولوشن بالاتر مد نظر باشد می توان متغیر های اندازه گیری زمان را بزرگتر انتخاب نمود.

نکته ای که در این روش قابل ذکر است این است که فقط یک بیت نهایی می بایست یک باشد، در غیر این صورت اگر دو بیت یک پشت هم در انتهای داده ی ارسالی وجود داشته باشد تشخیص آن امکان پذیر نخواهد بود. یعنی 0b10111111 بزرگترین عددی است که می تواند در این روش دریافت شود و از آنجا که از محدوده اسکی ( تا ۱۲۸) بزرگتر است، در دریافت داده های کاراکتری مشکلی وجود نخواهد داشت.

مثال: (LSB و MSB در سریال معکوس شکل های زیر می باشد، برای بهتر دیده شدن ابتدا MSB بعد از استارت قرار گرفته است.)

بررسی تکنیک با فرض داده در ورودی 0x00 باشد:

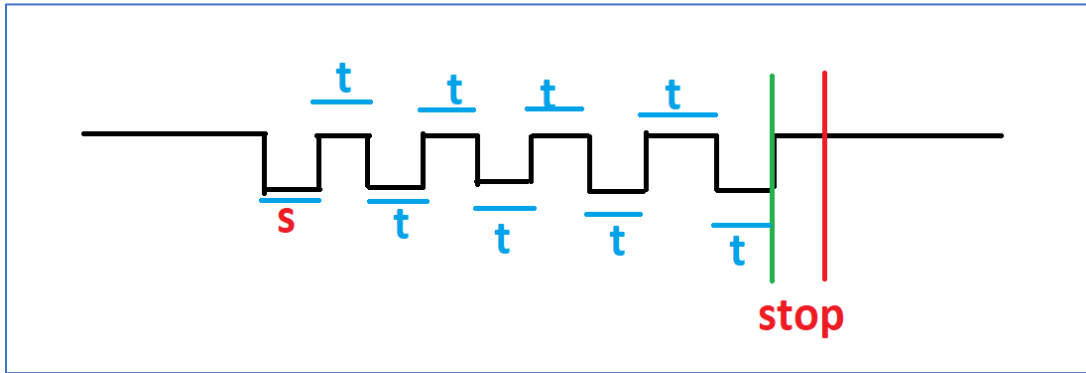


در این حالت زمان اندازه گیری شده برابر است با آغاز استارت تا حد سبز رنگ، و زمان دوم می شود از سبز رنگ تا حد مجاز شمارش که تغییر رخ نداده است و از آنجا که یک می باشد، می تواند انتهای کار در نظر گرفته شود. این تایم اوت، را می توان حداقل ۱۰ برابر زمان بیت در پایین ترین نرخ قابل تشخیص در نظر گرفت. حال در این مثال فقط یک زمان معتبر وجود دارد که با تقسیم بر ۹ عرض هر بیت بدست می آید و از آنجا که این زمان فقط داده ی صفر است پس داده نیز صفر خواهد بود. از

این پس اگر پین کانال تایمر، با سریال یکی بود، مستقیم از پورت سریال نیز با باد بدست آمده می توان استفاده کرد یا اینکه از همین روش استفاده نمود.

مثال:

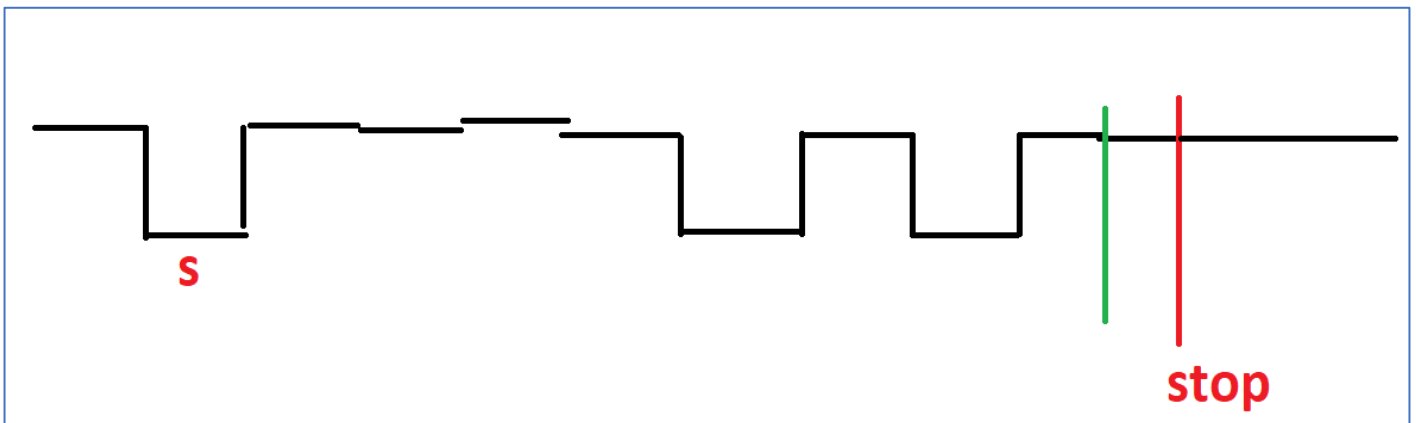
بررسی تکنیک با فرض داده در ورودی 0xAA باشد:



در این حالت نیز با همان تکنیک زمان ها را اندازه می گیریم و هر زمان را با مقدار آن ذخیره می کنیم، در تایم اوتی که در stop bit رخ می دهد، تمام زمان ها را جمع کرده و مجدد با تقسیم، زمان هر بیت را بدست آورد. مجدد برای ساخت بیت ها نیز زمان هر کدام از آن ها همراه مقدار آن ها مشخص است که می توان به راحتی با عملیات های ساده ای چون شیفت و OR آن دیتا را بازیابی کرد.

مثال:

بررسی تکنیک با فرض داده در ورودی 0xF5 باشد: (خطوط شکسته در تصویر زیر برای بهتر دیده شدن بیت ها می باشند).



در اینجا نیز:

- زمان بخش اول (بیت استارت)
- زمان تک تک بیت ها

با هم جمع می شوند. تایم اوت نهایی نیز مانند قبل ایجاد می شود یعنی هیچ تغییری در سطح یک به مدت چندین برابر زمان بیت در پایین ترین باد ریت مد نظر. حال مجدد تقسیم بر ۹، عرض بیت و در نهایت استخراج داده ها با توجه به زمان هایی که داشته اند.

### پیشنهاد در پیاده سازی:

زمان هر تغییر می تواند در یک آرایه ۹ بایتی (بسته به تنظیم تایمر) ذخیره شود. محاسبات در تایم اوت مربوط به عدم تغییر سطح، نگارش می شود که زمان ها جمع می گردند. مقادیر تغییر وضعیت نیز می توانند در یک بیت فیلد ۹ بیتی نگه داری شوند، این بیت فیلد می تواند کنار بیت فیلد های دیگر برنامه باشد تا اینگونه بایت اضافه ای برای این مورد مصرف نشود. برای بهبود اثر نویز نیز می توان، زمان های خیلی کوچک که توسط تایمر خوانده شده است و از عرض بیت در حداکثر باد ریت کمتر است صرفه نظر کرد. چرا که این تغییرات معتبر نخواهند بود.