



درس مدارهای منطقی  
تکلیف کامپیوتری ۴  
استفاده از رجیسترها و شمارنده‌ها در کاربرد انتقال سریال داده  
دانشکده‌گان فنی دانشگاه تهران  
دانشکده مهندسی برق و کامپیوتر  
دکتر بیژن علیزاده  
نیم‌سال دوم سال تحصیلی ۱۴۰۲-۰۳  
دستیار آموزشی: نیما خدابخشی

## مقدمه

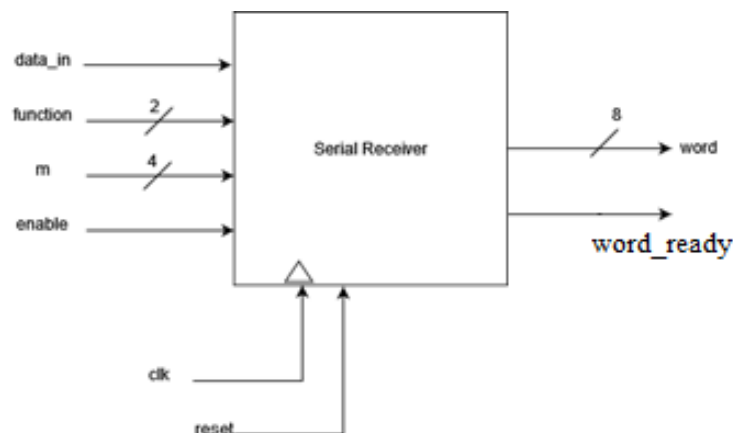
روش‌های انتقال داده بین المان‌های سخت‌افزاری مختلف، به طور کلی به دو دسته سریال و موازی تقسیم می‌شوند. در روش‌های انتقال داده به صورت سریال، بیت‌های داده پشت سرهم از طریق یک رابط، با پروتکلی مشخصی ارسال می‌شوند. در تمامی بردهای FPGA رابطی به منظور انتقال سریالی بیت‌ها وجود دارد که بر اساس پروتکل UART کار می‌کند. عموماً سخت‌افزاری با همین عنوان در بردهای FPGA این مسوولیت را به‌عهده دارد. استاندارد انتقال سریالی داده به نام RS-232 معروف است.

## بخش اول

هدف از این تمرین، طراحی مداری است که به صورت سریالی، یک داده هشت‌بیتی را از یک ورودی تک‌بیتی دریافت کرده و آنها را به صورت یک کلمه هشت‌بیتی بر روی خروجی قرار می‌دهد. ساختار موردنظر را serial\_receiver می‌نامیم که وظیفه آن دریافت بیت‌های کلمه ورودی از رابط سریالی با تنظیمات مشخص است. همانطور که در شکل ۱ نشان داده شده است، ورودی‌های serial\_receiver شامل ورودی تک‌بیتی data\_in (یک بیت از بیت‌های یک کلمه هشت‌بیتی)، ورودی ۲ بیتی function، ورودی ۴ بیتی m، ورودی reset و ورودی clk است.

ورودی function مشخص‌کننده نوع عملیات مطابق جدول ۱ است. بیت اول function (یعنی function[0]) نشان‌دهنده جهت ورودی‌ها است (ابتدا بیت‌های LSB دریافت می‌شود یا MSB؟). اگر function[0] = 1 باشد، نشان‌دهنده آن است که مقادیر روی ورودی data\_in از بیت‌های کم‌ارزش (LSB) شروع می‌شوند. اگر function[0] = 0 باشد، نشان‌دهنده آن است که مقادیر روی ورودی data\_in از بیت‌های پرارزش (MSB) شروع می‌شوند. بیت دوم function (یعنی function[1]) مشخص می‌کند که فاصله زمانی بین دو بیت متوالی قابل دریافت، چند سیکل است. اگر function[1] = 0 باشد، به این معنی است که هر ۱۶ سیکل یک بیت جدید داده آماده دریافت (از طریق ورودی data\_in) است. اگر function[1] = 1 باشد، به این معنی است که هر ۱۶-m سیکل یک بیت جدید داده آماده دریافت (از طریق ورودی data\_in) است. در UART به این فاصله زمانی BaudRate گفته می‌شود.

همچنین یک سیگنال خروجی به نام word\_ready نیز وجود دارد که هنگامی که ۸ بیت یک کلمه به طور کامل آماده شد، یک می‌شود. این بیت تا دریافت اولین بیت بعدی، یک باقی مانده و دوباره صفر می‌شود.



شکل ۱- نمایی از ورودی‌ها و خروجی‌های مدار serial\_receiver

جدول ۱- عملکرد مدار برای حالات مختلف function

function[1:0]	توضیح
00	داده‌ها به ترتیب از بیت پرارزش ارسال می‌شوند و نرخ ارسال هر ۱۶ سیکل یک بیت است.
01	داده‌ها به ترتیب از بیت کم‌ارزش ارسال می‌شوند و نرخ ارسال هر ۱۶ سیکل یک بیت است.
10	داده‌ها به ترتیب از بیت پرارزش ارسال می‌شوند و نرخ ارسال ۱۶-m سیکل یک بیت است.
11	داده‌ها به ترتیب از بیت کم‌ارزش ارسال می‌شوند و نرخ ارسال ۱۶-m سیکل یک بیت است.

بنابراین با تعریف ورودی دو بیتی function، تنظیمات زیر در طراحی لحاظ خواهند شد:

- **قابلیت تنظیم سرعت دریافت داده ورودی (function[1]):** توسط ورودی ۴ بیتی m تنظیم شده که برای دریافت هر بیت از کلمه، چند سیکل clock باید صبر کنیم.
- **قابلیت تنظیم ترتیب دریافت بیت‌های داده (function[0]):** بسته به تنظیم‌های رابط، گاهی در ابتدا بیت‌های کم‌ارزش (LSB) و گاهی ابتدا بیت‌های پرارزش (MSB) ارسال می‌شوند. در نتیجه، باید قابلیت چنین تنظیمی را داشته باشیم.

همچنین یک سیگنال enable نیز در مدار وجود دارد که تنها در صورتی که این سیگنال یک باشد، مدار کار موردنظر را انجام می‌دهد. خروجی مدار نیز یک کلمه ۸ بیتی می‌باشد که قاعدتا باید برابر با کلمه دریافت‌شده باشد.

در این بخش، طراحی مدار خواسته‌شده را به صورت دستی انجام داده و نتیجه را در گزارش کار بیاورید. برای انجام این طراحی، انتظار می‌رود که از بلوک‌هایی مانند Counter، Shift Register، Multiplexer و سایر بلوک‌ها و گیت‌های موردنیاز استفاده کنید. بلوک دیاگرام طراحی انجام‌شده شامل این بلوک‌ها و نحوه اتصال آنها را با دقت کافی مشخص کنید.

**راهنمایی:** برای شمارش تعداد سیکل‌ها، از Counterی استفاده کنید که خروجی CarryOut دارد. برای ذخیره و دریافت تک بیت‌ها می‌توانید از Shift Register استفاده کنید. بهتر است Shift Register را طوری طراحی کنید که با یک سیگنال کنترلی قابلیت شیفت به چپ یا راست را داشته باشد. سپس با اتصال مناسب این المان‌ها به یکدیگر، به عملکرد موردنظر برسید.

## بخش دوم

مدار طراحی شده در بخش اول را با استفاده از SystemVerilog توصیف کنید. توجه کنید که نام گذاری ورودی ها و خروجی ها در top-module حتما به صورت زیر باشد.

```
module serial_receiver (input clk, input reset, input [1:0] function, input data_in, input enable,  
                        input [3:0] m, output [7:0] word, output word_ready);
```

**نکته:** در صورت نوشتن کد بصورت رفتاری، می بایست تطابق بخش های مختلف کد با مدار طراحی شده در بخش اول را در گزارش شرح دهید.

**امتیازی:** به کدهایی که به صورت ساختاری و مطابق با ساختار طراحی شده در بخش اول نوشته شده باشند، نمره امتیازی تعلق می گیرد.

## بخش سوم

یک test-bench برای کد بخش دوم نوشته و در هر ۴ حالت تعریف شده مطابق جدول ۱، عملکرد مدار را مورد بررسی قرار دهید. در گزارش کار خود شکل موج های حاصل را بیاورید.

موارد قابل تحویل در گزارش کار:

- بلوک دیاگرام ساختارهای طراحی شده
- شکل موج های خروجی test-bench و تحلیل آنها
- لطفا فایل کدهای خود را به همراه pdf گزارش به صورت یک فایل zip به فرم Name\_UID\_CA4 آپلود نمایید.

با آرزوی بهترین ها برای شما