



هم طراحی سخت افزار و نرم افزار

(سال تحصیلی ۱۴۰۰-۱۴۰۱، نیم سال اول)

تمرین چهارم: Finite State Machine with Datapath

طراحی و پیاده سازی FSM:FSMD

۱. یک FSM برای عملیات ضرب به روش جمع و شیفتم برای ورودی های ۶ بیتی طراحی نمایید.
۲. FSM طراحی شده در مرحله قبل را با استفاده از زبان GEZEL پیاده سازی کنید (مطابق شکل ۱ نیاز است که دو سیگنال start و done برای مشخص کردن شروع و پایان عملیات در نظر گرفته شوند).
- به منظور شبیه سازی ساختار پیاده سازی شده، نیاز است که یک datapath (dp) به صورت مجزا وظیفه مقداردهی ورودی ها را بر عهده داشته باشد. این dp نقش testbench را ایفا می کند و در کنار dp اصلی طرح، مطابق با شکل ۱ فراخوانی می گردد.
۳. پس از آماده سازی طرح، با استفاده از دستور fdlvhd، شبیه سازی را در محیط GEZEL انجام داده و خروجی های ابزار را گزارش نمایید.
۴. پس از اتمام شبیه سازی و اطمینان از صحت طرح، با استفاده از دستور fdlvhd، کد سخت افزاری طرح را به زبان VHDL استخراج کرده و ضمیمه گزارش خود نمایید.

```

dp multiplier( in x : ns(6);
               in y : ns(6);
               in start : ns(1);
               out result : ns(12);
               out done : ns(1))
{
    // Define the internals of the FSMD here ..
}
fsm mul_ctl(multiplier){

}

dp TB(out x,y: ns(6); out start: ns(1)){
    sfg s1{

    }
}
hardwired TB_ctl(TB) {s1;}

dp sysmul{
    sig x,y: ns(6);
    sig result: ns(12);
    sig start,done: ns(1);

    use multiplier(x,y,start,result,done);
    use TB(x,y,start);
}
system S{
    sysmul;
}

```

شکل ۱: نمونه ساختار کد به زبان GEZEL

۵. گزارش مختصری از فرآیند انجام این تمرین به همراه تصاویری از FSMD طراحی شده و خروجی برنامه در ترمینال linux، در قالب یک فایل PDF، به انضمام کدهای نوشته شده و فایل‌های خروجی ابزار را به صورت فشرده، همراه با نام و شماره دانشجویی در سامانه درس‌افزار بارگذاری نمایید.

موفق باشید