

دکتر موحسن:

instance گرفتن از بلوک مادرها

اسلاید 40: دستورهایی که بطور موزنی اجرا می شوند: module instantiation

Initiation blocks → testbenches استفاده (از)

logical loop: assign کردن یک متغیر wire به خورش

sequential blocks: { always blocks, initiation blocks }
always@(*)
always@ (clkedge)

اسلاید 61: در display اول: (دستورهای display 1 تا 4)، مقدار آیزدهای a, b را تغییر می دهد و مقدار جاری پس از display مقدار جاری را نشان می دهد، هنوز a, b عوض نشده اند.

در display دوم: #1 یک واحد زمانی گذشته و مقدار آیزده در مقدار جاری را ثبت شده است.

⚠ در wire ها اگر دو تعاریف هم، مقدار نامشخص خواهد شد ولی در reg و integer تعارض ایجاد نمی شود و فقط آخرین دستور اجرا می شود.

⚠ اگر وسط اجرای always، دوباره شرط تحریک قرار شود، به اول نمی گردد و صرفاً ادامه می دهد به اجرا

تولید اولین کلاک: (اسلاید 64) از همان ابتدا `always` ترکیبی شود چون در ابتدا $clk = X$ و پس از زمان 1 به آن مقدار 1 را نسبت داده ایم پس یعنی مقدار `clk` تغییر کرده در زمان 0

برای ساده سازی `combinational logic` { (1) استفاده از `continuous assignment`

(2) استفاده از `sequential code` (قوانین اسلاید 67)

اگر زیر یک `if` یا `else if`، دستور نویسیم `error` نمی رسد ولی `latch` تشکیل می شود و مقدار قبلی متغیر حفظ می شود (اسلاید 71)