

الف) تفاوت بین سیگنال‌ها و متغیرها در VHDL و کاربرد هرکدام

سیگنال‌ها (Signals):

- تعریف: سیگنال‌ها عناصری هستند که بین فرآیندها و کامپوننت‌ها ارتباط برقرار می‌کنند.
- زمان بندی: تغییرات سیگنال‌ها با تاخیر انجام می‌شود (در پایان فرآیند یا بعد از تاخیر مشخص).
- محدوده: در (architecture) یا (package) تعریف می‌شوند.
- عملگر انتساب: از `'>='` برای انتساب استفاده می‌شود.
- کاربرد:

- برای ارتباط بین فرآیندها
- پیاده‌سازی عناصر حافظه مانند فلیپ‌فلاپ‌ها
- مدل‌سازی خطوط ارتباطی در سخت‌افزار

متغیرها (Variables):

- تعریف: متغیرها برای ذخیره‌سازی موقت داده‌ها در یک فرآیند استفاده می‌شوند.
- زمان بندی: تغییرات فوراً اعمال می‌شود (بلافاصله پس از انتساب).
- محدوده: فقط در داخل فرآیندها، توابع یا روال‌ها تعریف می‌شوند.
- عملگر انتساب: از `'='` برای انتساب استفاده می‌شود.
- کاربرد:

- برای محاسبات موقت در فرآیندها
- پیاده‌سازی الگوریتم‌های پیچیده
- شمارنده‌های حلقه‌ها
- ذخیره‌سازی نتایج میانی محاسبات

ب) مقادیر ابتدایی و نهایی در 3 سیکل کلاک

مقادیر اولیه (قبل از اولین سیکل کلاک):

'signal_a' = '0' -

'signal_b' = '0' -

'var_a' = '0' -

'var_b' = '0' -

سیکل کلاک 1:

1. ('0') `signal_a` <= `var_a`

2. ('0') `signal_b` <= `var_b`

3. ('1') `var_a` := not `var_a`

4. ('1') `signal_b` := not `signal_b` - این تغییر فقط روی نسخه محلی اثر دارد

مقادیر پس از سیکل 1:

'signal_a' = '0' - (تغییر در پایان سیکل اعمال می‌شود)

'signal_b' = '0' - (تغییر با `>` بر تغییر با `:=` غالب می‌شود)

'var_a' = '1' -

'var_b' = '0' -

سیکل کلاک 2:

1. ('1') `signal_a` <= `var_a`

2. ('0') `signal_b` <= `var_b`

3. ('0') `var_a` := not `var_a`

4. ('1') `signal_b` := not `signal_b` - باز هم فقط نسخه محلی

مقادیر پس از سیکل 2:

'signal_a' = '1' -

'signal_b` = '0` -

'var_a` = '0` -

'var_b` = '0` -

سیکل کلاک 3:

('0') `signal_a` <= `var_a` .1

('0') `signal_b` <= `var_b` .2

('1') `var_a` := not `var_a` .3

4. ('1') `signal_b` := not `signal_b` - نسخه محلی

مقادیر پس از سیکل 3:

'signal_a` = '0` -

'signal_b` = '0` -

'var_a` = '1` -

'var_b` = '0` -