

عنوان پروژه:

شبیه‌سازی ساختاری مدارهای ترکیبی استاندارد ISCAS در سطح گیت و تولید فایل bench.

تعریف پروژه:

با استفاده از زبان‌های برنامه‌نویسی یک شبیه‌ساز طراحی و پیاده‌سازی نمایید که فایل‌های استاندارد ISCAS را بخواند و فایل bench آن را تولید نماید. این شبیه‌ساز همچنین باید قادر باشد که با مقادیر ورودی که به مدار اعمال می‌کنید، مقادیر تمام خروجی‌ها را تحویل دهد و در فایل‌ها با پسوند log ذخیره کند.

توجه: فایل‌های با فرمت isc را می‌توانید از آدرس زیر دریافت کنید.

<https://pld.ttu.ee/~maksim/benchmarks/iscas85/isc/>

نیازمندی‌ها:

شبیه‌ساز باید بتواند تمام گیت‌های منطقی را پوشش دهد که شامل AND, NAND, OR, NOR, XOR, XNOR, NOT, BUF, FANOUT می‌باشند.

شبیه‌ساز باید بتواند مقادیر 0 (LOW), 1 (HIGH) و U (Unknown) را پشتیبانی کند.

شبیه‌ساز باید به طور عمومی باشد و هر مداری که استاندارد با فرمت مد نظر را دارد پردازش کند.

گیت‌ها به صورت Lazy می‌باشند یعنی اگر در گیت AND یک ورودی برابر 0 باشد خروجی مستقل از ورودی دیگر 0 می‌باشد. این در حالی است که اگر یک ورودی 1 باشد، ورودی دیگر تعیین‌کننده خروجی این گیت است؛ اگر ورودی دیگر U خروجی نیز U و اگر 1 باشد، خروجی نیز 1 می‌باشد.

BUF گیتی است که فقط یک ورودی دارد و خروجی آن همانند ورودی آن می‌باشد.

FANOUT به معنی انشعاب است.

مرحله شبیه‌سازی فقط بر روی فایل bench انجام می‌شود.

ورودی مدار در یک فایل جدا قرار می‌گیرد که در خط اول نام سیم‌ها (که با شماره مشخص می‌شوند) و در خط دوم مقدار ورودی هر سیم تعیین می‌شود. از فاصله به عنوان جداسازی میان ورودی‌ها استفاده می‌شود.

قواعد نامگذاری با توجه به ISCAS تعیین می‌شود برای مثال اگر یک سیم به انشعاب برسد (FANOUT) انشعاب‌های آن به صورت سیم_۱، سیم_۲ و ... نامگذاری می‌شوند.

در این گام از پروژه، در فایل‌های isc قسمت‌های >sa# بدون استفاده می‌باشند و فقط برای رعایت استاندارد وجود دارند.

توضیحات بیشتر از ISCAS پیوست گردیده است.

نمونه مثال:

C17.isc file:

```
*c17 iscas example (to test conversion program only)
```

```

*-----
* total number of lines in the netlist ..... 17
* simplistically reduced equivalent fault set size = 22
* lines from primary input gates ..... 5
* lines from primary output gates ..... 2
* lines from interior gate outputs ..... 4
* lines from ** 3 ** fanout stems ... 6
*
* avg_fanin = 2.00, max_fanin = 2
* avg_fanout = 2.00, max_fanout = 2
*
*
1 1gat inpt 1 0 >sa1
2 2gat inpt 1 0 >sa1
3 3gat inpt 2 0 >sa0 >sa1
8 8fan from 3gat >sa1
9 9fan from 3gat >sa1
6 6gat inpt 1 0 >sa1
7 7gat inpt 1 0 >sa1
10 10gat nand 1 2 >sa1
1 8
11 11gat nand 2 2 >sa0 >sa1
9 6
14 14fan from 11gat >sa1
15 15fan from 11gat >sa1
16 16gat nand 2 2 >sa0 >sa1
2 14
20 20fan from 16gat >sa1
21 21fan from 16gat >sa1
19 19gat nand 1 2 >sa1
15 7
22 22gat nand 0 2 >sa0 >sa1
10 20
23 23gat nand 0 2 >sa0 >sa1
21 19

```

Expected C17.bench:

```
# c17
# 5 inputs
# 2 outputs
# 0 inverter
# 6 gates ( 6 NANDs )

INPUT(1)
INPUT(2)
INPUT(3)
INPUT(6)
INPUT(7)

OUTPUT(22)
OUTPUT(23)

10 = NAND(1, 3)
11 = NAND(3, 6)
16 = NAND(2, 11)
19 = NAND(11, 7)
22 = NAND(10, 16)
23 = NAND(16, 19)
```

Circuit Design:

