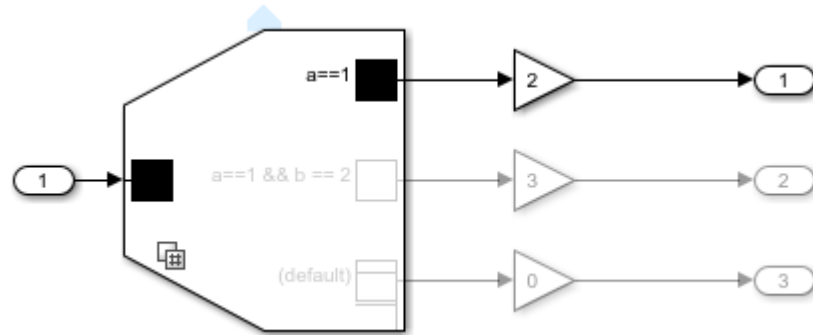


■ 通らない条件を作成した場合



b!=2にして
コード生成

端子	バリエーション制御式	条件 (読み取り専用)
1	a==1	✓ (N/A)
2	a==1 && b == 2	✓ (N/A)
3	(default)	✓ (N/A)

上記において端子2が選択されることはない。

◆ a==1、b!=2の場合、端子1が選択される。
エラーは出ない。

◆ a==1、b==2の場合、
複数端子が選択されるのでコンパイルエラー
(次ページ参照)

```
void sub3_step(void)
{
    /* Gain: '<Root>/Gain' incorporates:
     * Inport: '<Root>/In1'
     * Outport: '<Root>/Out1'
     */
    #if a == 1

        sub3_Y.Out1 = 2.0 * sub3_U.In1;

    #endif

    /* End of Gain: '<Root>/Gain' */

    /* Gain: '<Root>/Gain1' incorporates:
     * Gain: '<Root>/Gain3'
     * Inport: '<Root>/In1'
     * Outport: '<Root>/Out2'
     * Outport: '<Root>/Out4'
     */
    #if a == 1 && b == 2

        sub3_Y.Out2 = 3.0 * sub3_U.In1;

    #elif a != 1

        /* Outport: '<Root>/Out4' incorporate
         * Gain: '<Root>/Gain3'
         * Inport: '<Root>/In1'
         */
        sub3_Y.Out4 = 0.0 * sub3_U.In1;

    #endif

    /* End of Gain: '<Root>/Gain1' */
}
```

コード上でb==2に書き換えると、
上の#ifと、この#ifのどちらも通す
ことが可能になるため、意図しない
動作になる可能性がある

↓
2つめの条件を削除、もしくは
すべてを#if - #elifでつなげて排他
で処理するのが意図通りのコード
と思われる

もし(a==1 && b==2)が成り立つようなことがあったとき、Cコードのコンパイル時に以下の56行目を通るためコンパイルエラーになります。これによってモデルとコードの挙動が一致します。

