

NYCU-ECE DCS-2024

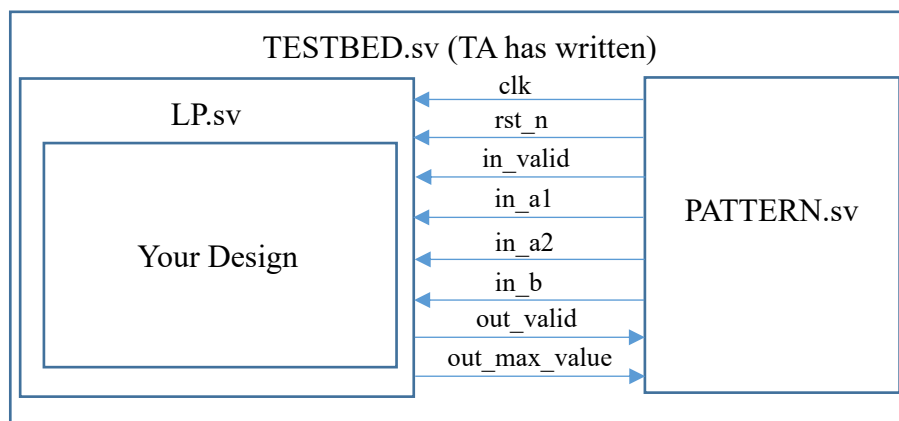
Online Test

Design: Linear Programming

資料準備

1. 從 TA 目錄資料夾解壓縮:
`% tar -xvf ~dcsTA01/ OT.tar`
2. 解壓縮資料夾 OT 包含以下:
 - a. 00_TESTBED/
 - b. 01_RTL/
 - c. 02_SYN/
 - d. 03_GATE/
3. 在 00_TESTBED/中，有助教寫好的簡易 pattern 可供參考，demo 時會使用另一個完整 pattern

Block Diagram



設計描述

在數學中，線性規劃 (Linear Programming) 特指目標函數和約束條件皆為線性的最佳化問題。這次考試要大家完成一個整數線性規劃運算電路，找出符合約束條件**最佳整數解**。

任意線性規劃問題可表達成以下形式：

$$\begin{aligned} &\text{maximize } \mathbf{c}^T \mathbf{x} \\ &\text{subject to } \mathbf{Ax} \leq \mathbf{b} \quad (1) \end{aligned}$$

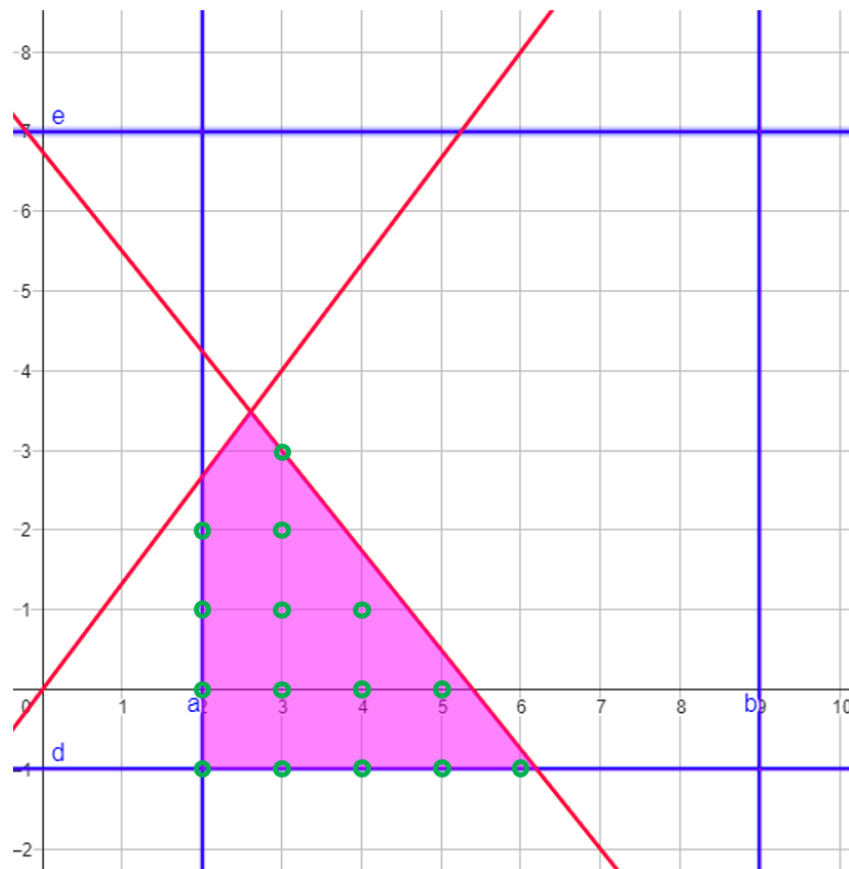
本題設計的電路所要計算的是二維，以 x_1, x_2 作為變數，具有六個約束條件的線性規劃問題，並且找出符合條件且 x_1, x_2 皆為整數的目標函數最大值。

因此在上述表示式(1)中， $\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ ； $\mathbf{c}^T = [c_1 \ c_2]$ ，為目標函數的係數； \mathbf{A} 為 6×2 矩陣，第一、二行分別代表約束條件的 x_1, x_2 係數。

以下列線性規劃問題為例：

目標函數： $x_1 + 3x_2$

約束條件： $x_1 \geq 2 \quad x_1 \leq 9 \quad x_2 \geq -1 \quad x_2 \leq 7 \quad 5x_1 + 4x_2 \leq 27 \quad 8x_1 - 6x_2 \geq 0$



本範例套用表示式(1)， $\mathbf{c}^T = [1 \ 3]$

$$\mathbf{A} = \begin{bmatrix} -1 & 0 \\ 1 & 0 \\ 0 & -1 \\ 0 & 1 \\ 5 & 4 \\ -8 & 6 \end{bmatrix} \quad \mathbf{b} = \begin{bmatrix} -2 \\ 9 \\ 1 \\ 7 \\ 27 \\ 0 \end{bmatrix}$$

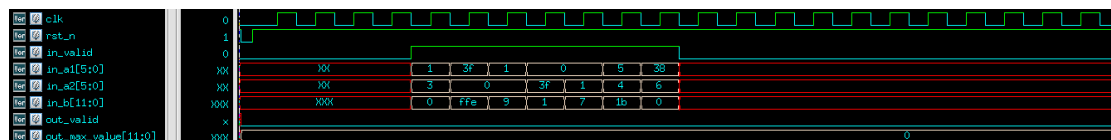
上圖圓圈的座標點為符合約束條件的整數解，目標函數在 $\mathbf{x} = \begin{bmatrix} 3 \\ 3 \end{bmatrix}$ 有最大值12。

在本題中，pattern每筆輸入皆包含四個如同範例，會在二維座標限制出一個矩形區域的約束條件，i.e. 每筆輸入的A會有四列和範例中A的前四列相同(但約束條件之間順序不一定和範例相同)。此矩形區域長寬皆不會大於15，矩形區域上下界的 x_1, x_2 不會超過正負31。

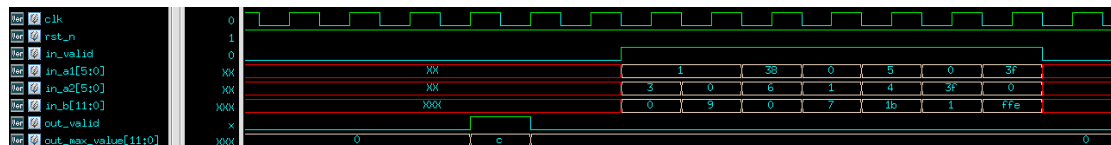
Pattern提供的所有約束條件交集不會是空集合，i.e.一定會有解。

輸入與輸出

每一筆輸入Pattern會拉起in_valid並對in_a1, in_a2, in_b連續給值7 cycles。第一cycle，Pattern會分別在in_a1, in_a2給目標函數係數 c_1, c_2 的值，in_b則會輸入0。第二至七cycle，Pattern會依序在in_a1輸入六個約束條件的 x_1 係數，i.e. A的第一行；在in_a2輸入六個約束條件的 x_2 係數，i.e. A的第二行；在in_b輸入b。



請注意，輸入階段的第二至七cycle，輸入六個約束條件的先後順序是隨機的，例如下圖和上圖都是前頁的線性規劃例題，但約束條件輸入順序不同。



當輸出結果時，out_valid會拉起並在out_max_value 輸出符合約束條件且 x_1, x_2 皆為整數的目標函數最大值，持續1 cycle。

Hint

由於本題求的是 x_1, x_2 皆為整數時的解，且其中四個約束條件框出大小不超過15x15的矩形範圍，因此可窮舉矩形範圍內所有整數座標點，並檢查剩餘的兩個約束條件，即可找出符合條件的最佳解。

Inputs

Signal name	Number of bit	Description
clk	1	10ns clock signal
rst_n	1	Asynchronous negedge reset signal
in_valid	1	Pull high during digit input
in_a1	6	Signed integer, the coefficient of x_1
in_a2	6	Signed integer, the coefficient of x_2
in_b	12	Signed integer, the constant b

Outputs

Signal name	Number of bit	Description
out_valid	1	Pulled high during out_max_value output (reset required)
out_max_value	12	Signed integer, maximum of the objective function. (reset required)

Specifications

1. Top module name: **LP**(File name : **LP.sv**)
2. 在非同步負準位 reset 後，所有的 output 訊號必須全部歸零。
3. Output 要在 Input 結束後的 300 cycles 內輸出。
4. Output 僅能輸出 1 cycle，不能多不能少。
5. 下一筆 input 會在 out_valid pull down 後 1~4 個 cycle 送入
6. 02_SYN result 不行有 **error** 且不能有任何 **latch**。
7. Timing report 的 slack 必須為 non-negative 且 result 為 **Met**。
8. Gate level simulation 不能有 timing violation。
9. Clock period 10 ns。
10. Input delay = 0.5 * clock period; Output delay = 0.5 * clock period
11. 請勿使用 ***error***, ***latch***, ***congratulation*** or ***fail*** 當作 logic / wire / reg / submodule / parameter 的名稱，否則 demo 結果會是 fail
Note: *代表在該 word 前後的任何符號，比如: error_test 即為禁止的。

上傳檔案

1. 請將LP.sv重新命名後上傳E3。命名規則：LP_dcsxxx.sv 例如：工作站帳號是dcs175的同學須將他的Design LP.sv重新命名為LP_dcs175.sv並上傳E3。
2. 6/4 18:20考試時間結束前上傳E3的OT繳交區
3. 2nd Demo deadline: 6/9 23:59

Grading policy

1. Students who pass 1st Demo yield 100 points.
2. Students who pass 2nd Demo yield 60~80 points, depending on performance.
3. Performance: Latency * Area

Note

Template folders and reference commands:

1. 01_RTL/ (RTL simulation) → **./01_run**
2. 02_SYN/ (synthesis) → **./01_run_dc**
3. 03_GATE/ (gate-level simulation) → **./01_run**