

# LAB-8

Pengantar Sistem Digital

Semester Ganjil 2022/2023

### Petunjuk Pengerjaan:

- Kerjakan semua soal sesuai dengan spesifikasi tiap soal
- Mohon baca setiap spesifikasi soal dengan seksama sebelum bertanya kepada asisten dosen
- Jika ada soal yang membingungkan atau kesalahan pada soal, silakan bertanya kepada salah satu asisten dosen yang sudah *stand-by*
- Lakukan submisi semua file (sesuai spesifikasi yang ada pada soal) di Scele sebelum **Jumat, 2 Desember 2022 pukul 11.45**
- Jika **mengumpulkan telat selama 10 menit  $< x \leq 2$  jam**, akan dikenakan **penalti sebesar 40 poin**. Lebih dari waktu tersebut, lab **tidak akan dinilai**

# Pattern (15 poin)

Revisi 0 - 2/12/2022

Pak Esde melihat performamu yang sangat keren saat membuat register. Kini, ia membutuhkan bantuanmu untuk membuatnya sebuah counter yang menaik dengan kelipatan 4. Namun, Pak Esde ingin membantumu untuk mengerjakan counter ini.

Sebelum membuat counter, kamu dapat memperhatikan pola perubahan digit bilangan biner yang kelipatannya berbeda di setiap clocknya. Berikut diberikan tabel angka yang direpresentasikan dengan 4 digit biner dan hexadecimal.

Biner	Hexadecimal
0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	9
1010	A
1011	B
1100	C
1101	D
1110	E
1111	F

---

Apakah kamu dapat melihat pola perubahan digit biner di setiap kenaikan 2 dan 4?  
Berikan tanggapanmu terhadap kalimat berikut:

- (1) Jelaskan pola perubahan tersebut pada masing-masing kenaikan 2 dan 4.
- (2) Apa yang perlu dilakukan untuk mengimplementasikannya dengan Counter with Parallel Load?

**Perhatikan bahwa plagiarisme adalah pelanggaran serius dengan sanksi nilai 0.**

Kumpulkan hasil dari **Pattern** dalam satu file pdf (yang nanti akan digabung dalam satu file zip dengan hasil bagian lainnya) dengan penamaan:

- **LAB8-Pattern-[kode-asdos]-[npm]-[nama].pdf**

<3 PSD 2022/2023-1

# Counter With Parallel Load

## (50 poin)

Revisi 0 - 2/12/2022

---

Setelah menemukan pola kenaikan tersebut, tentunya kamu sudah terbantu untuk membuat counter custom dengan pola **kenaikan 2 dan 4** pesanan Pak Esde. Counter ini harus diimplementasikan dengan rangkaian **Counter with Parallel Load**.

Spesifikasi rangkaian:

- Input: inc2, inc4, Count, Load, dan 4 bit biner untuk Load
- Output: 4 bit biner
- Gunakan juga "Hex Digit Display" pada folder Input/Output untuk menampilkan output di atas dalam bentuk hexadecimal
- Flip-flop yang digunakan sesuai yang ada di slide Bab 6: Registers and Register Transfers Slide 44

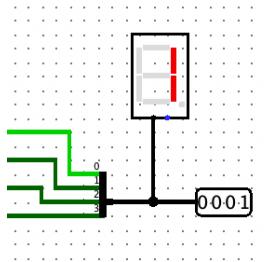
Petunjuk:

- Counter berjalan dengan normal (terjadi kenaikan pada output) hanya ketika Count = 1 dan Load = 0
- Jika Load = 1, pada clock berikutnya, output akan bernilai sama dengan load. Counter juga tidak berjalan pada keadaan ini (output setiap terjadi clock akan terus sama dengan load)
- Jika inc2 maupun inc4 bernilai 0, Counter berjalan dengan kenaikan 1. Jika inc2 = 1, Counter berjalan dengan kenaikan 2. Sama halnya dengan inc4, inc4 = 1 menandakan counter berjalan dengan kenaikan 4. Jika inc2 dan inc4 bernilai 1, Counter akan berjalan dengan kenaikan 4.

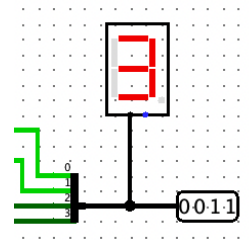
Contoh:

Inc2 = 1

Keadaan awal:

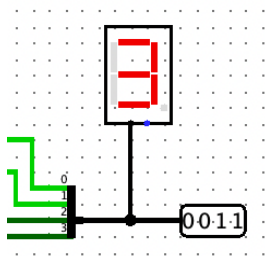


Setelah 1 Clock:

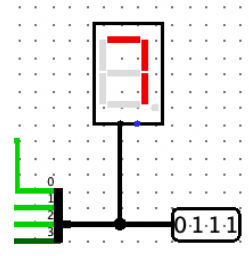


Inc4 = 1

Keadaan awal:



Setelah 1 Clock:



<3 PSD 2022/2023-1

# Bigger Counter (35 poin)

Revisi 1 - 2/12/2022

---

Pak Esde sangat puas dengan hasil kerjamu. Counter custom yang dipesannya sangat baik. Akan tetapi, seiring berjalannya waktu ia membutuhkan counter yang bisa menghitung lebih banyak. Ia pun meminta bantuanmu lagi untuk membuat counter yang mampu **menghitung dari  $(0000)_{16}$  atau  $(0000\ 0000\ 0000\ 0000)_2$  sampai  $(FFFF)_{16}$  atau  $(1111\ 1111\ 1111\ 1111)_2$** . Pak Esde tahu kamu sedang banyak tugas sehingga ia memintamu untuk merangkainya menggunakan **beberapa Counter custom atau Counter with Parallel Load** yang telah kamu buat **sebelumnya**.

Spesifikasi rangkaian:

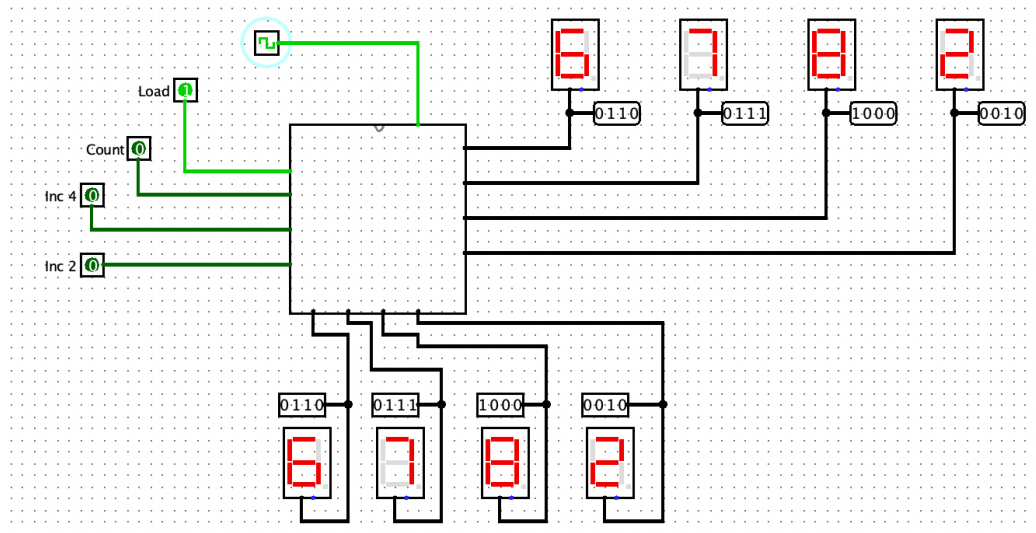
- Input: inc2, inc4, Count, Load, dan 16 bit biner untuk Load
- Output: 16 bit biner
- Gunakan juga "Hex Digit Display" pada folder Input/Output untuk menampilkan output di atas dalam bentuk hexadecimal

Petunjuk :

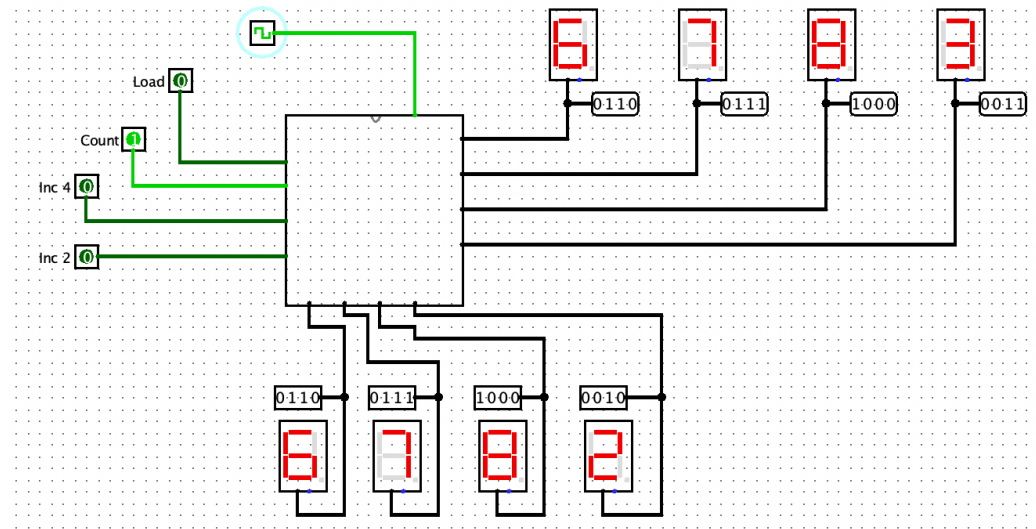
- Petunjuk sama seperti bagian sebelumnya
- Perlu diingat bahwa ini adalah Counter yang lebih besar saja cakupannya. Sehingga setiap terjadi kenaikan yang berubah tentu dimulai dari digit terkecil/LSB.

Contoh rangkaian jadi:

- **Load = 1, Count = 0, inc2 = 0, inc4 = 0. Saat clock:**

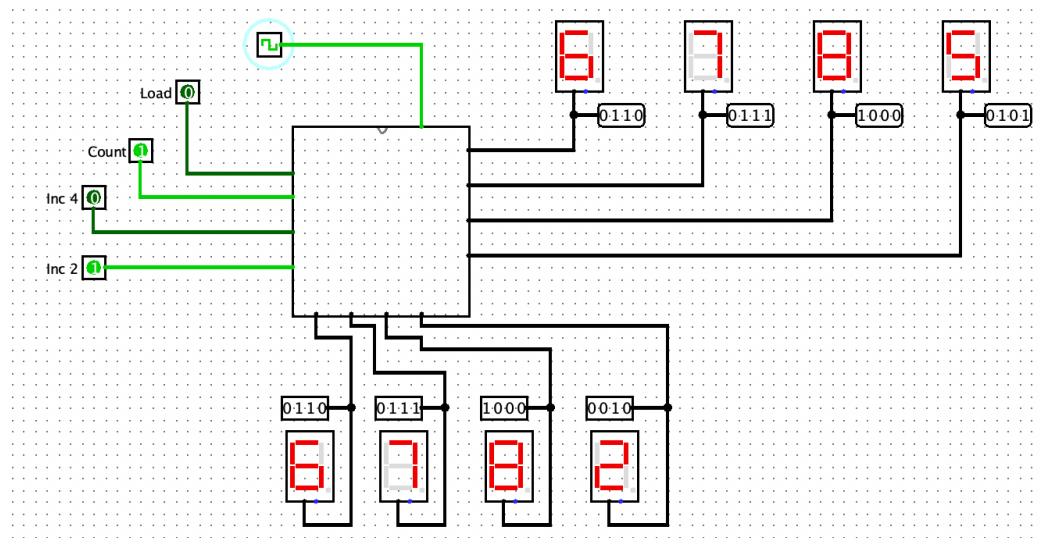


- **Load = 0, Count = 1, inc2 = 0, inc4 = 0. Saat clock:**

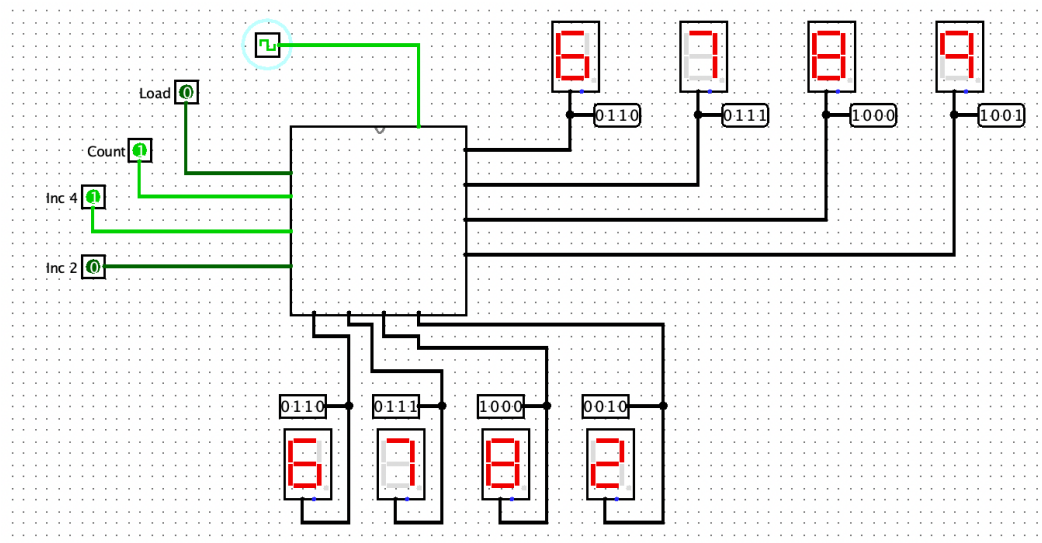




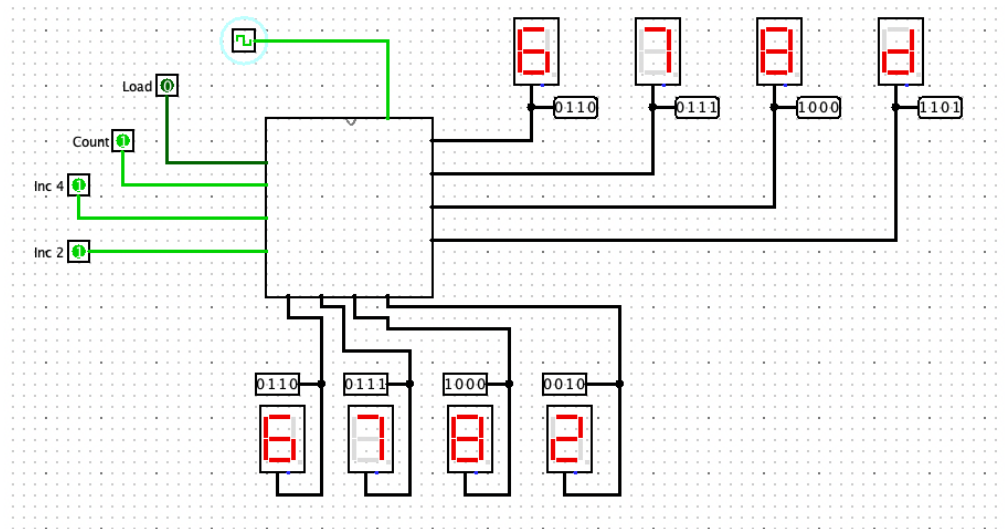
- Load = 0, Count = 1, inc2 = 1, inc4 = 0. Saat clock:



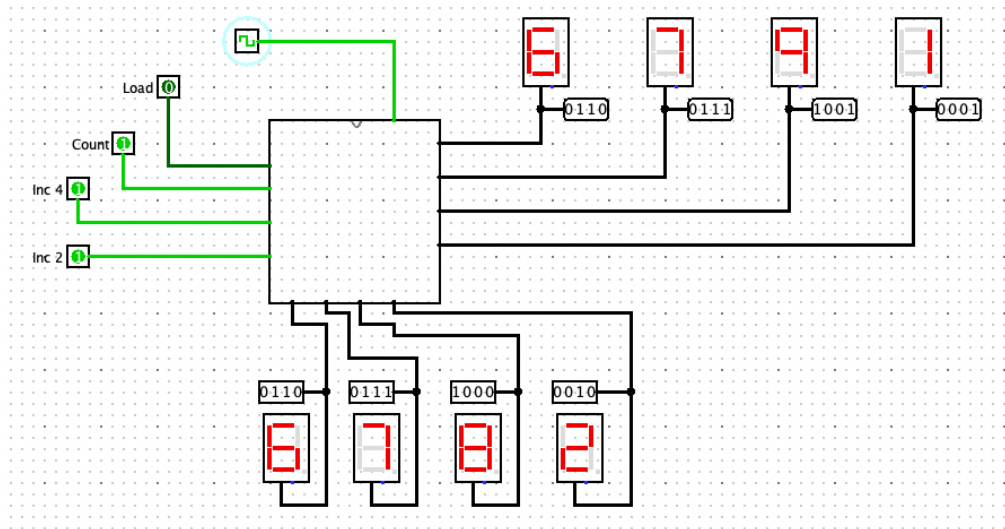
- Load = 0, Count = 1, inc2 = 0, inc4 = 1. Saat clock:



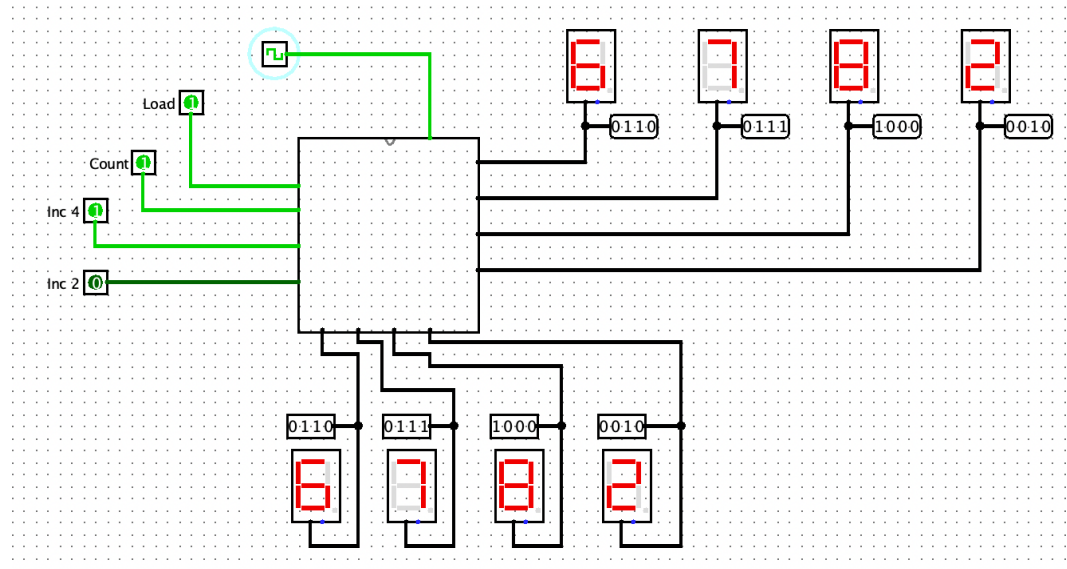
- Load = 0, Count = 1, inc2 = 1, inc4 = 1. Saat clock:



- Load = 0, Count = 1, inc2 = 0, inc4 = 1. Saat clock:



- Load = 1, Count = 1, inc2 = 0, inc4 = 1. Saat clock:



**Perhatikan bahwa plagiarisme adalah pelanggaran serius dengan sanksi nilai 0.**

Kumpulkan hasil dari bagian sebelumnya dan bagian ini ke dalam satu file logisim (yang nanti akan digabung dalam satu file zip dengan hasil bagian lainnya) dengan penamaan:

- **LAB8-Counter-[kode-asdos]-[npm]-[nama].circ**

<3 PSD 2022/2023-1

# Pengumpulan Submisi

- Kumpulkan semua file jawaban pada satu file .zip dengan format penamaan file (tanda '[' dan ']' tidak perlu ditulis):

**LAB8-[kode asdos]-[npm]-[nama].zip**

Contoh: LAB8-DY-1806146991-HarnindytoWicaksana.zip dengan isi:

- Jawaban Pattern dengan format penamaan  
**LAB8-Pattern-[kode-asdos]-[npm]-[nama].pdf**
- Circuit implementasi dengan format penamaan  
**LAB8-Counter-[kode-asdos]-[npm]-[nama].circ**

<3 PSD 2022/2023-1

