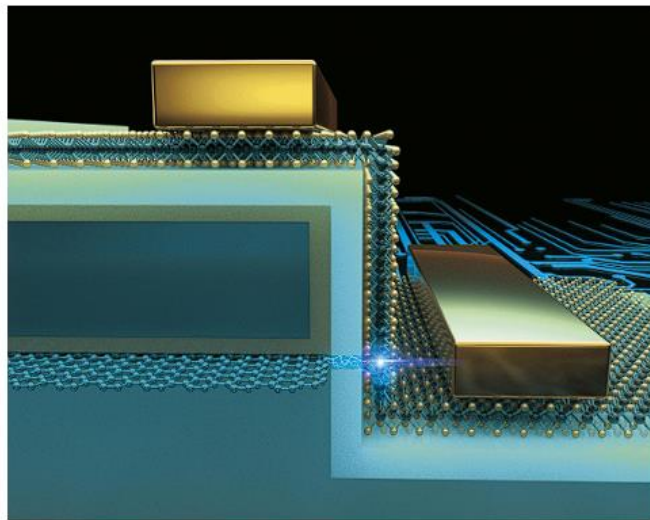


Nama : Muhammad Dzaki Arta

NIM : 13522149

3. 2 nm TranSISTER (7 poin + 3 poin bonus + ???)



[source](#)

Ketentuan

Buatlah sebuah simulasi komputer 4-bit. Simulasikan komponen CPU dan memori. Komponen CPU dapat menerima input melalui *switch* dan *clock signal* melalui *button*, sementara komponen memori digunakan untuk menyimpan *output* dari pengujian.

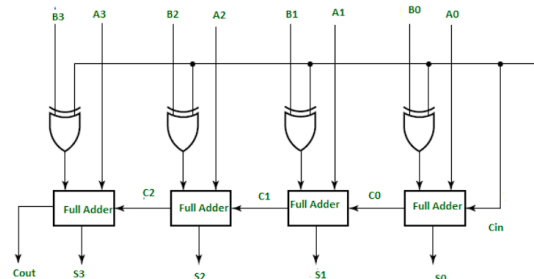
- Komponen CPU wajib memiliki ALU untuk menghitung aritmatika dasar.
- Simulasi komputer diuji dengan melakukan operasi aritmatika dasar (penjumlahan dan pengurangan).
- Pengerjaan dapat dilakukan dalam sebuah program simulasi, misal Logisim, Minecraft, [Simulator Redstone](#), TinkerCAD, atau program simulasi lainnya.

Prinsip Kerja

- **Adder** : Menambahka 2 angka biner 3 bit dengan membawa *carry* dari 1 bit ke bit berikutnya
- **Subtractor** : Mengurangkan satu angka biner 4-bit dari yang lain dengan hasil berupa signed bit

Desain Rangkaian

2.1 4 Bit Binary Adder

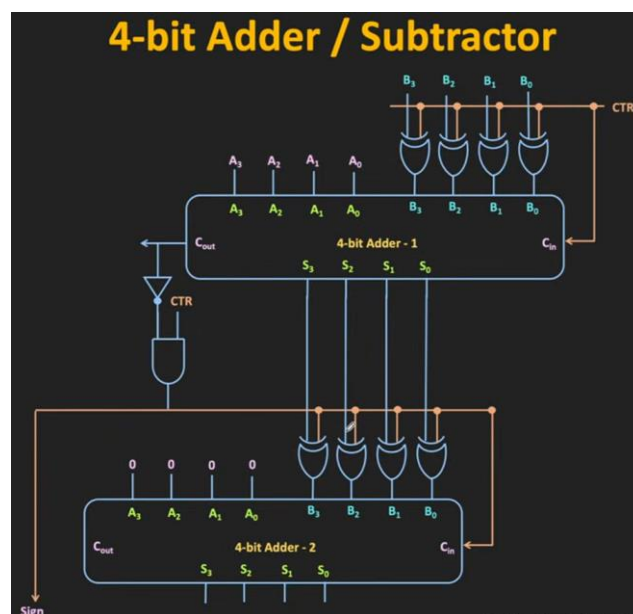


Gambar 2.1 Logika 4 Bit Binary Adder/Subtractor

(Sumber : <https://www.geeksforgeeks.org/4-bit-binary-adder-subtractor/>)

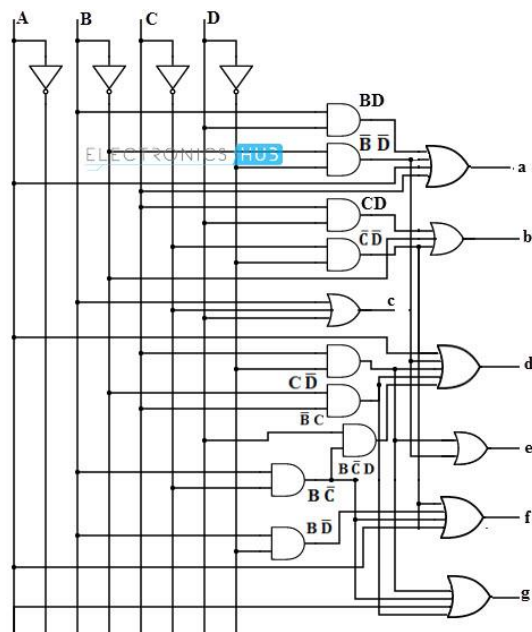
Rangkaian ini terdiri dari:

- **4 Full Adder:** Menggunakan full adder untuk setiap bit dari bilangan 4-bit.
- **XOR Gates:** Digunakan untuk membalikkan (invert) bit dari salah satu input ketika operasi pengurangan dilakukan.
- **Switch/Control Bit (Subtract):** Menentukan apakah operasi yang dilakukan adalah penjumlahan atau pengurangan.
- **Signed Bit:** Berisi bit yang menentukan negasi dari output program
- **7 Segment Display :** menampilkan hasil output dengan 7 segment



Gambar 2.2 Logika 4-bit Adder/Subtractor dengan output menggunakan signed bit

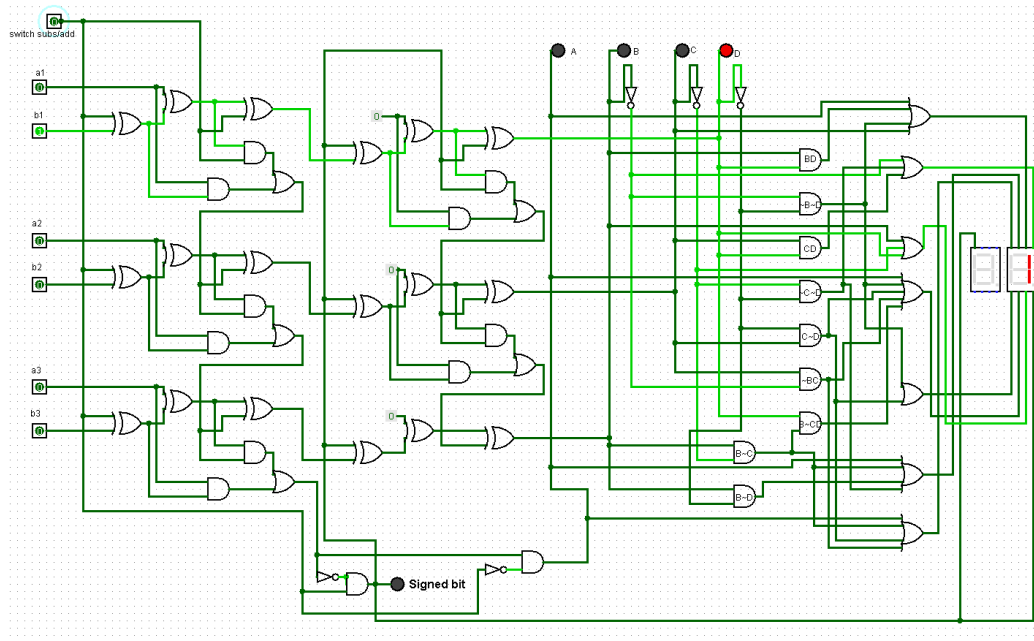
(Sumber: <https://www.youtube.com/watch?v=J7gPUP0aRug>)



Gambar 2.3 Logika untuk 7 segment display

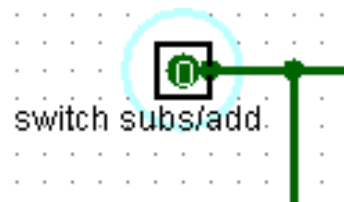
(Sumber : <https://www.electronicshub.org/bcd-7-segment-led-display-decoder-circuit/>)

Implementasi dalam Logisim

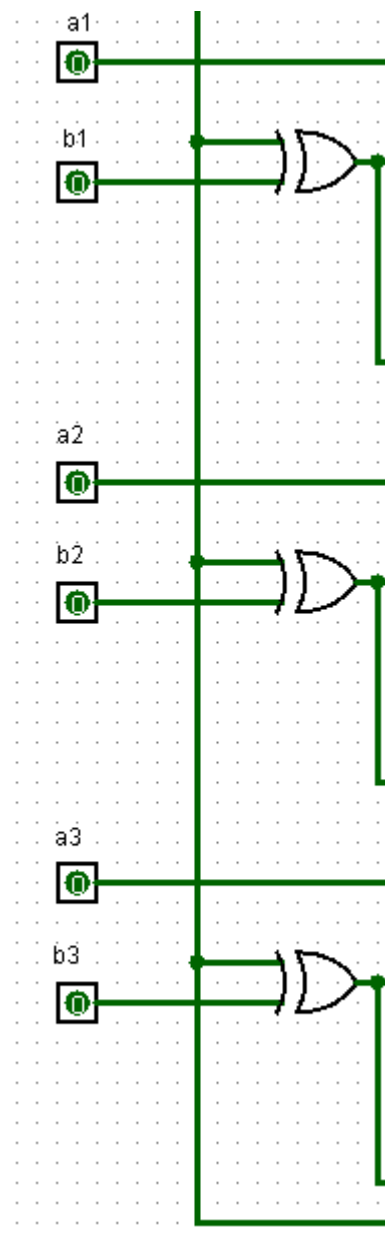


Gambar 3.1 Rangkaian dalam logisim

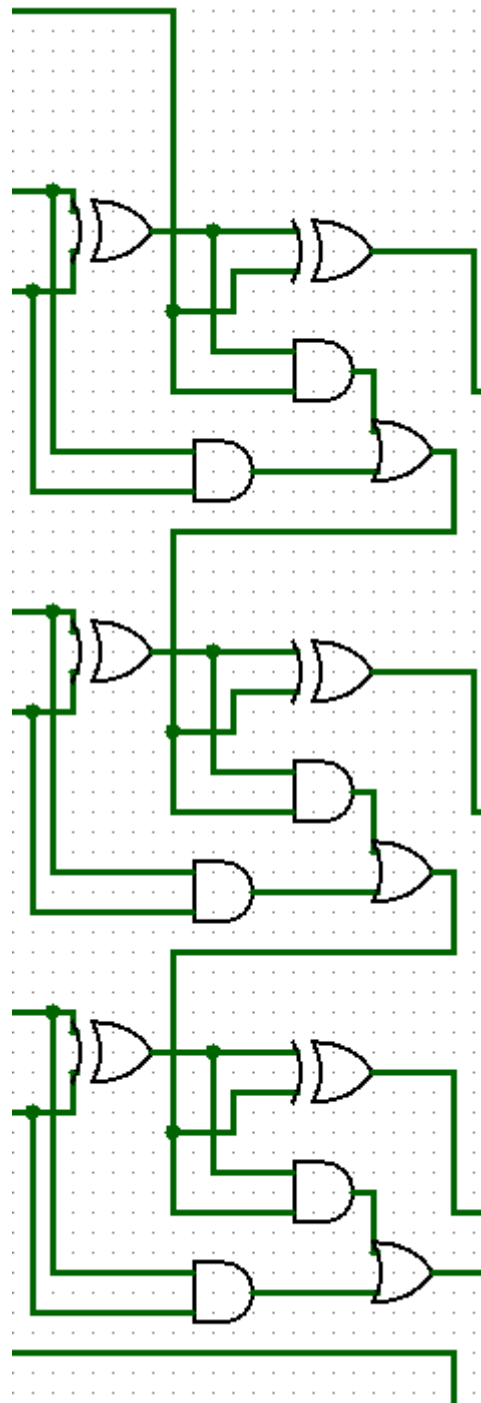
3.1 Switch/Control bit



3.2 Input Bit



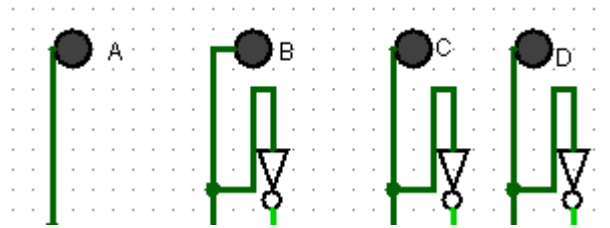
3.3 Full Adder



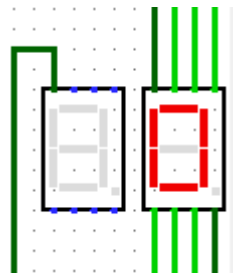
3.4 Signed Bit



3.5 Binary Output

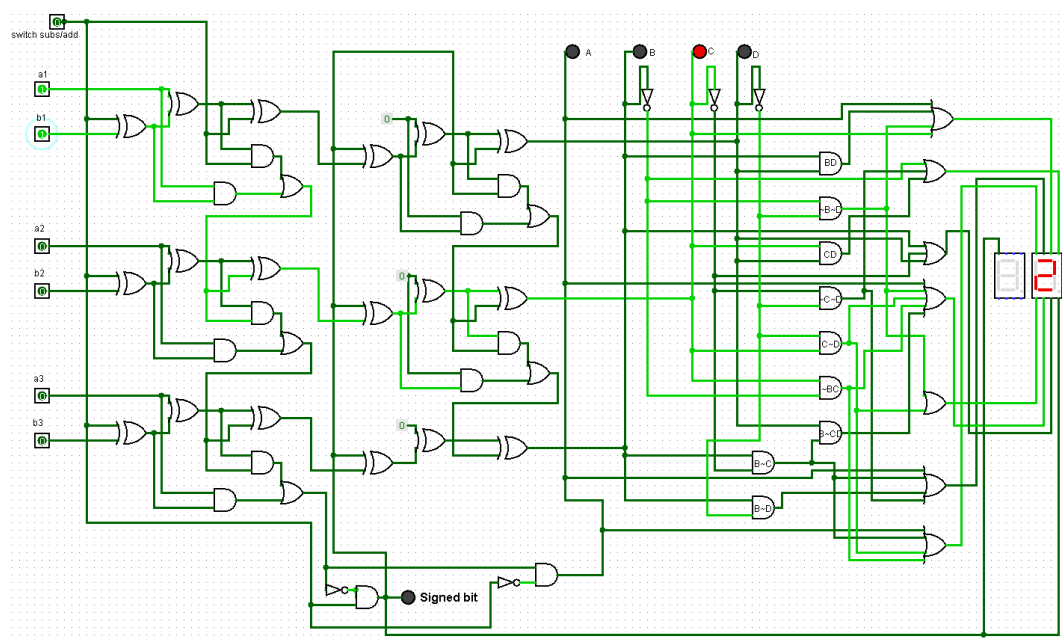


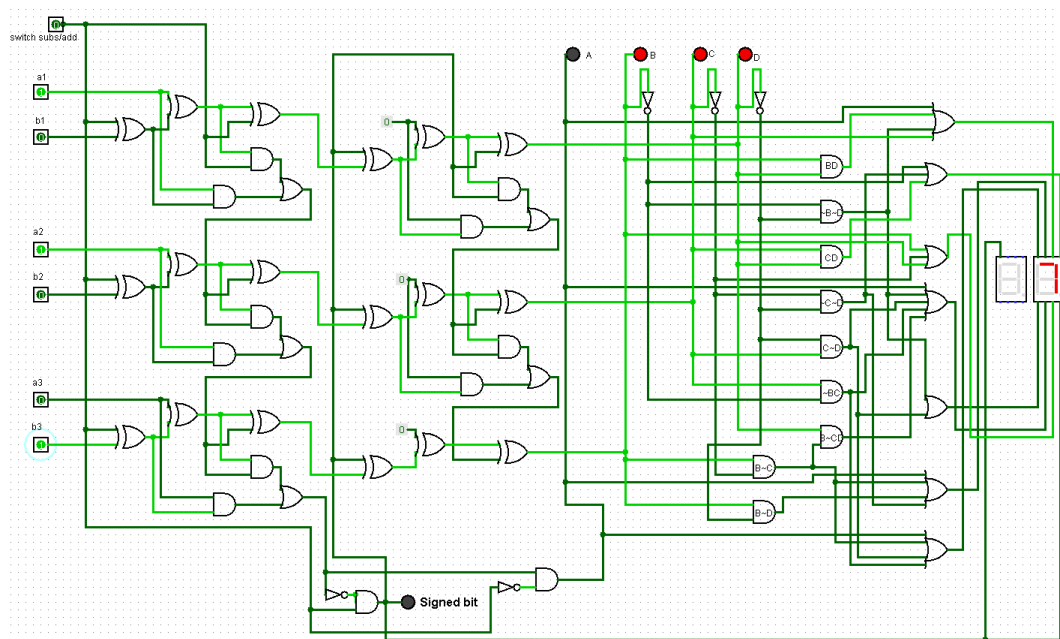
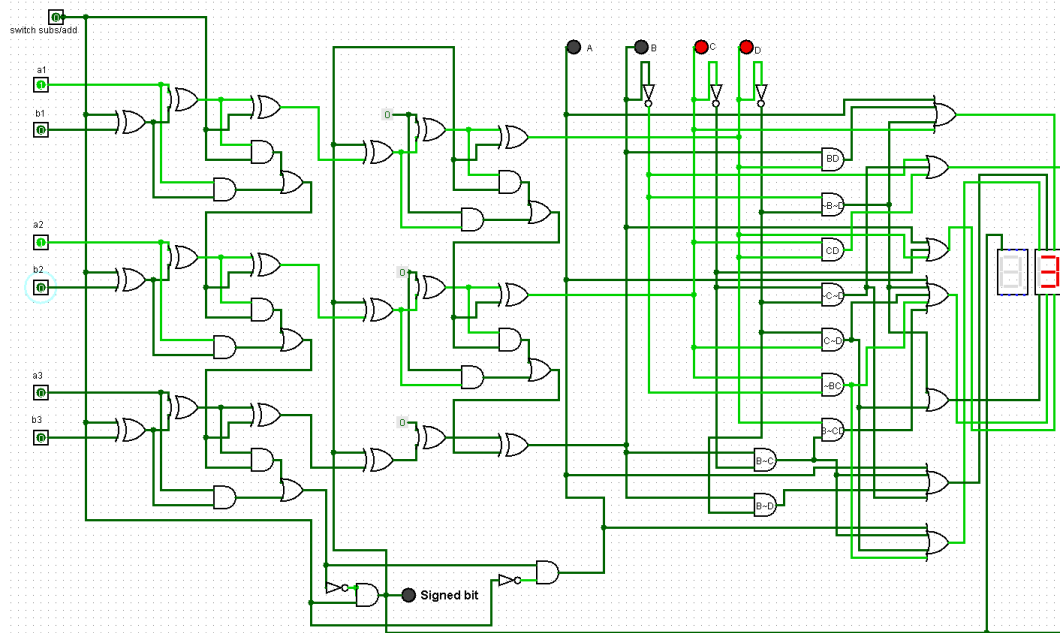
3.6 7 Segment Display



Simulasi

4.1 Adder





3.2 Subtractor

