Soal Terkait:

Minekarnaugh (3 poin + 3 poin)

Ketentuan

- Simulasikan 7 Segment pada *minecraft* menggunakan *redstone* dan *redstone lamp* sebagai lampu
- Redstone lamp harus bisa memunculkan angka dari 0 hingga 9
- Input disarankan menggunakan binary untuk memudahkan
- Sebagai contoh pada gambar setiap *lever* merepresentasikan angka biner dan yang paling kanan dijalankan sehingga menjadi 0001 atau representasi angka 1 pada desimal sehingga redstone lamp menunjukan angka 1
- Tidak diwajibkan untuk menggunakan *minecraft* secara langsung, gunakan aplikasi seperti redstone simulator jika tidak memiliki *minecraft* (~~Apa itu tlauncher dan legacy launcher~~)
- Jelaskan kenapa 7 segment tersebut bisa berjalan pada dokumen
- Tips : Gunakan aljabar boolean (peta *karnaugh*) dan pengetahuan *logic gate*
- Spoiler : Penulis hampir muntah saat membuat ini di *minecraft*

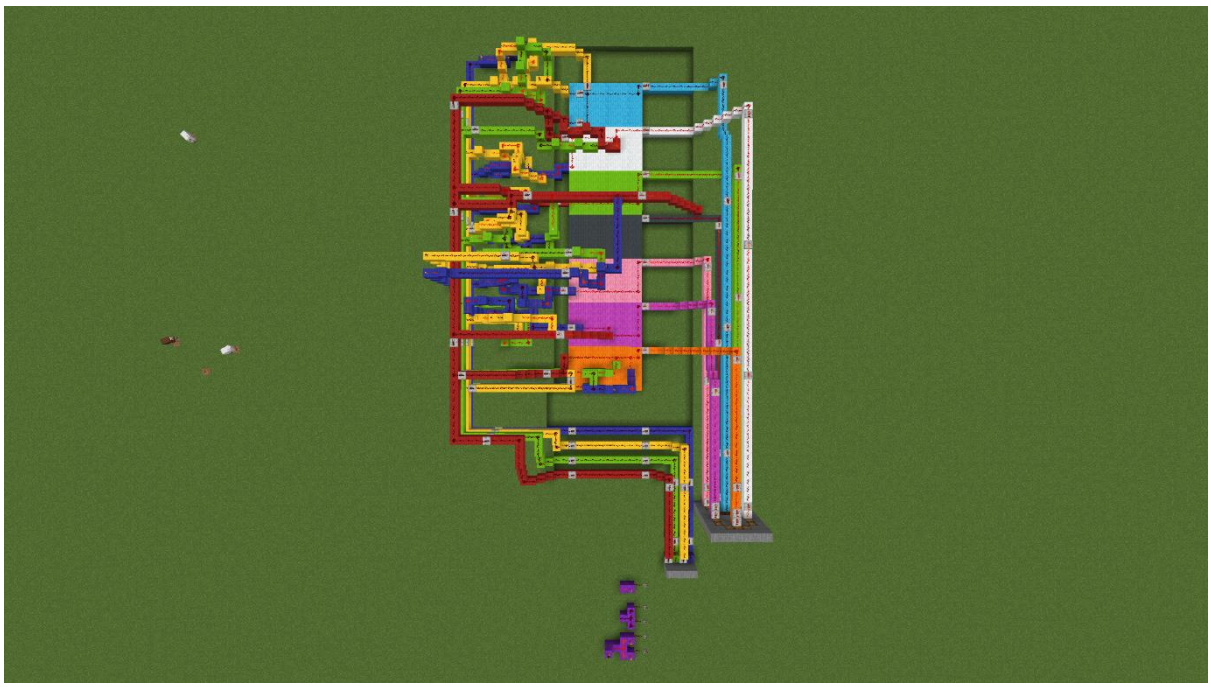
Tujuan

- Kapan lagi bisa simulasi *logic gate*
- “Oh jadi ini gunanya peta karnaugh”
- Bayangin dulu aja masuk sister karena jadi *redstone engineer*

Berkas

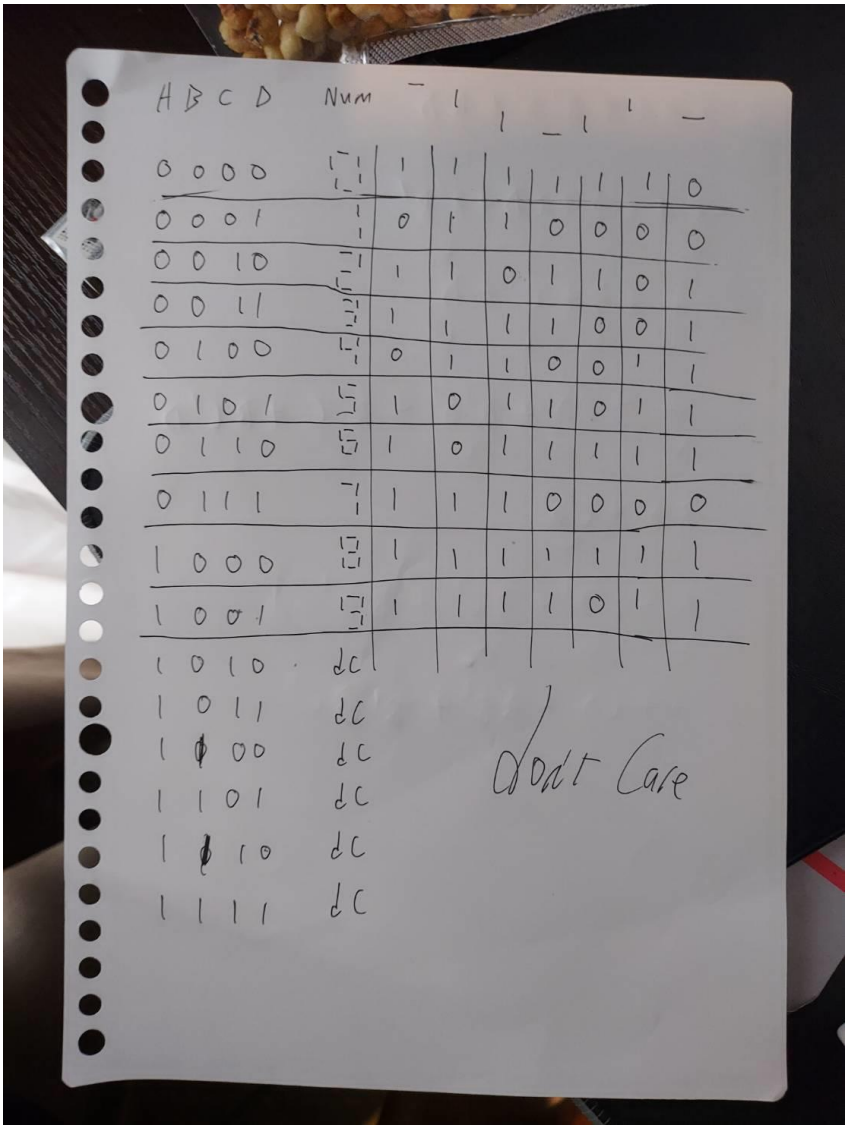
- Dokumen penjelasan implementasi *logic gate*
- Opsional : World minecraft dalam *zip*

Okeh jadi ini penulis buat di Minecraft Tlauncher versi optifine 1.17. World zip terdapat pada github (yang folder ‘redstone’ atau redstone.zip) dan hasilnya kurang lebih seperti ini:



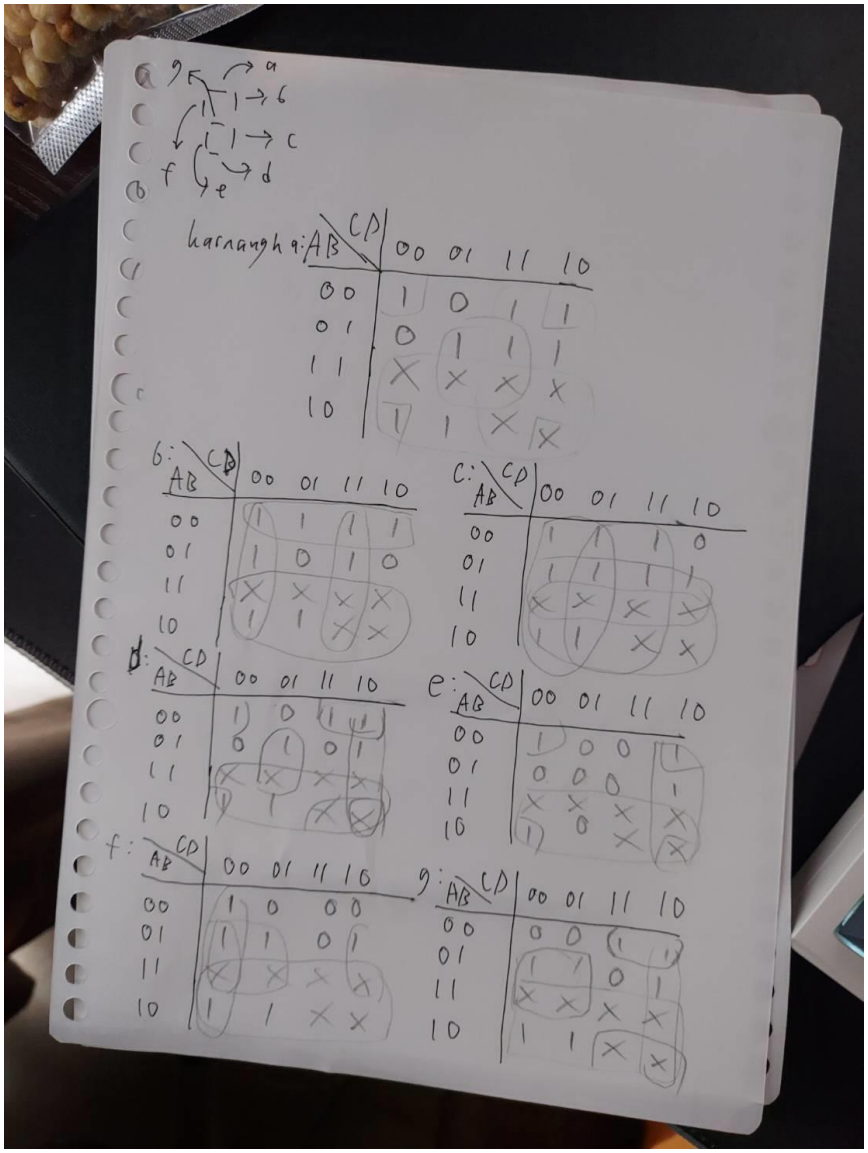
Pusing gak tuh. Oke penjelasan di halaman selanjutnya

Jadi sebelumnya mengenai prinsip pengerjaannya dulu, prinsip pengerjaannya menggunakan 4 lever yang merepresentasikan biner 0 dan 1. Dari situ didapat logic variabel untuk true false, direpresentasikan dengan A sampai D untuk masing – masing lever. Setelah itu untuk tiap segmen pada 7 segmen display dipetakan kondisi menyala (direpresentasikan 1) dan kondisi mati (direpresentasikan 0) pada setiap angka yang akan ditampilkan, hasil pemetaannya seperti berikut:

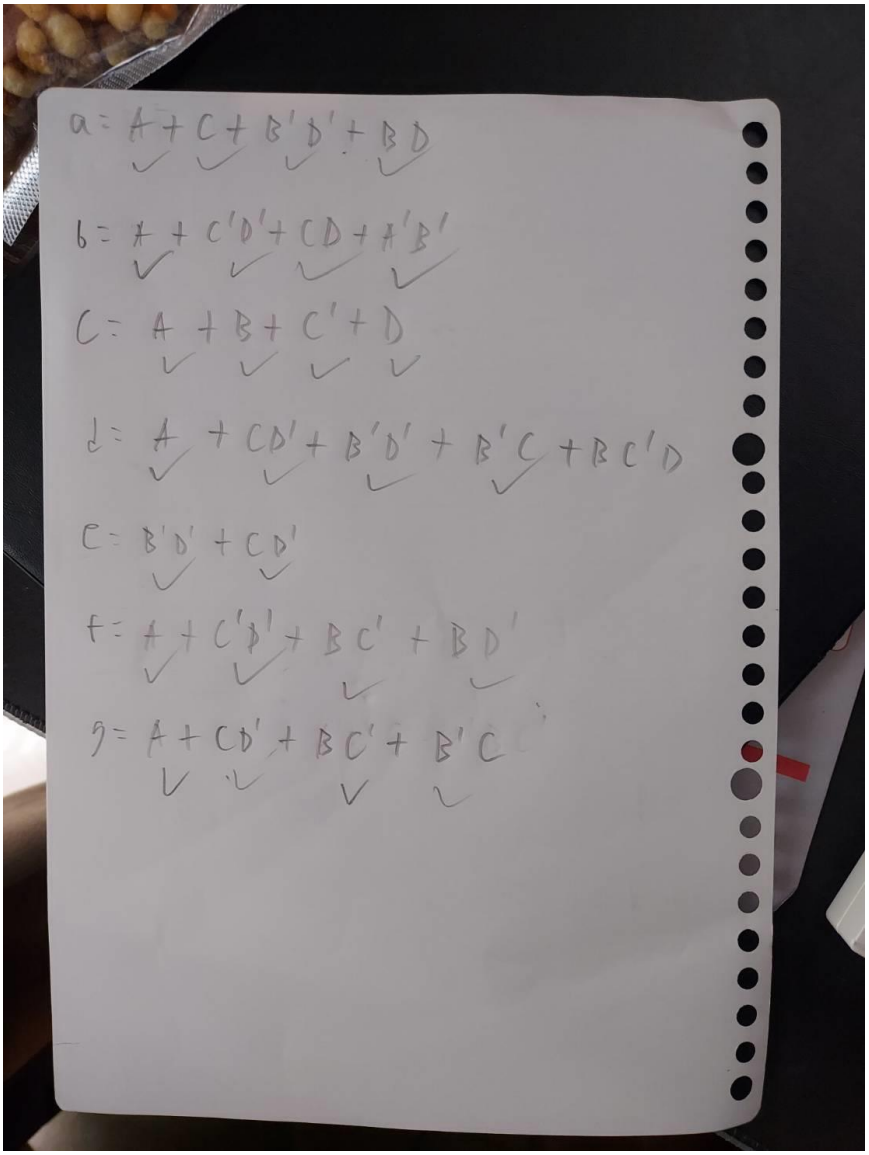


Adapula 10 sampai 15 adalah don’t care karena dalam seven segment display ini tidak membutuhkan hal tersebut.

Setelah itu didapat untuk setiap segmen sebuah peta Karnaugh yang menunjukkan kondisi dari segmen dalam setiap angka, adapula seperti berikut:



Dari peta Karnaugh ini bisa diambil logika yang paling sederhana dari 4 lever tersebut untuk membuat gerbang logika sesuai ke setiap segmen pada seven segment display. Hal ini dilakukan dengan mengelompokkan 1 dan don’t care (x). Untuk penjelasan peta Karnaugh dan penggunaannya mungkin bisa dipelajari lagi matdis (ehe penulis males ngejelasin). Didapat logika paling sederhana adalah sebagai berikut:



Tambah mengindikasikan logika or, perkalian mengindikasikan logika and, sementara kutip (‘) mengindikasikan operasi not. Nah jika sudah sampai di sini, tinggal membuat gerbang logika tersebut di Minecraft. Centangan yang ada di kertas tersebut penulis gunakan untuk menandai gerbang logika yang sudah dibuat jadi bisa diabaikan saja.

Sebelum membuat rangkaian di Minecraft, perlu dirancang dulu gerbang logika dalam redstone. Kebetulan penulis tidak terlalu jago Minecraft ya, gak begitu sering juga main, jadi rangkaian yang dibuat rada acak – acakan tapi yang penting bisa lah ya. Masalah sinkronisasi juga agak sulit soalnya travel distance redstone yang terbatas dan penggunaan repeater memberikan delay ke segment yang diimplementasikan lebih jauh. Adapula penulis mengimplementasikan logic gate di Minecraft dengan cara berikut:

1. Not Gate

Not gate diimplementasikan dengan menggunakan mekanisme redstone torch, yaitu Ketika suatu redstone torch diberi sinyal redstone, redstone torch tersebut akan mati. Dengan demikian, selalu didapatkan sinyal yang berlawanan dari sinyal lever yang diberikan

2. Or Gate

Or gate diimplementasikan dengan sangat sederhana, yaitu dengan menyambungkan redstone ke sinyal yang dapat berasal dari dua lever

3. And Gate

And Gate merupakan gerbang yang relatif lebih kompleks disbanding yang lain dalam pengimplementasiannya (mungkin karena penulis kurang main minecraft). Penulis mengimplementasikan and gate dengan cara memberikan dua atau lebih torch yang terhubung pada blok satu torch lain. Dua atau lebih torch ini lalu dihubungkan bloknya dengan lever dan memanfaatkan prinsip yang sama pada not gate. Setiap lever akan berperan mematikan satu dari dua atau lebih torch. Jika semua dari dua atau lebih torch yang terhubung pada blok satu torch lain itu mati, maka satu torch yang terhubung bloknya akan menyala dan memberikan sinyal.

Dengan didapatkannya rancangan setiap logic gate yang dibutuhkan pada kasus ini, penulis dapat membuat gerbang logika yang dibutuhkan untuk setiap segmen dan menghubungkannya pada lever. Penulis menggunakan wol berwarna untuk menandai sinyal lever dan segmen yang dikerjakannya.

Adapula hasil penyalaan lampu terlampir pada halaman selanjutnya.