

# **Pengembangan Sistem Square Root Digital dengan Algoritma Newton-Raphson untuk Pengolahan Data dari ADC 16-bit**

**Kelompok 01**

**Anggota:**

1. 13224001 – Yozia Gedalya Marcho Ginting
2. 13224002 – Jeva Steve Sinaga
3. 13224003 – Benedictus Kenneth Setiadi

## **Daftar Isi**

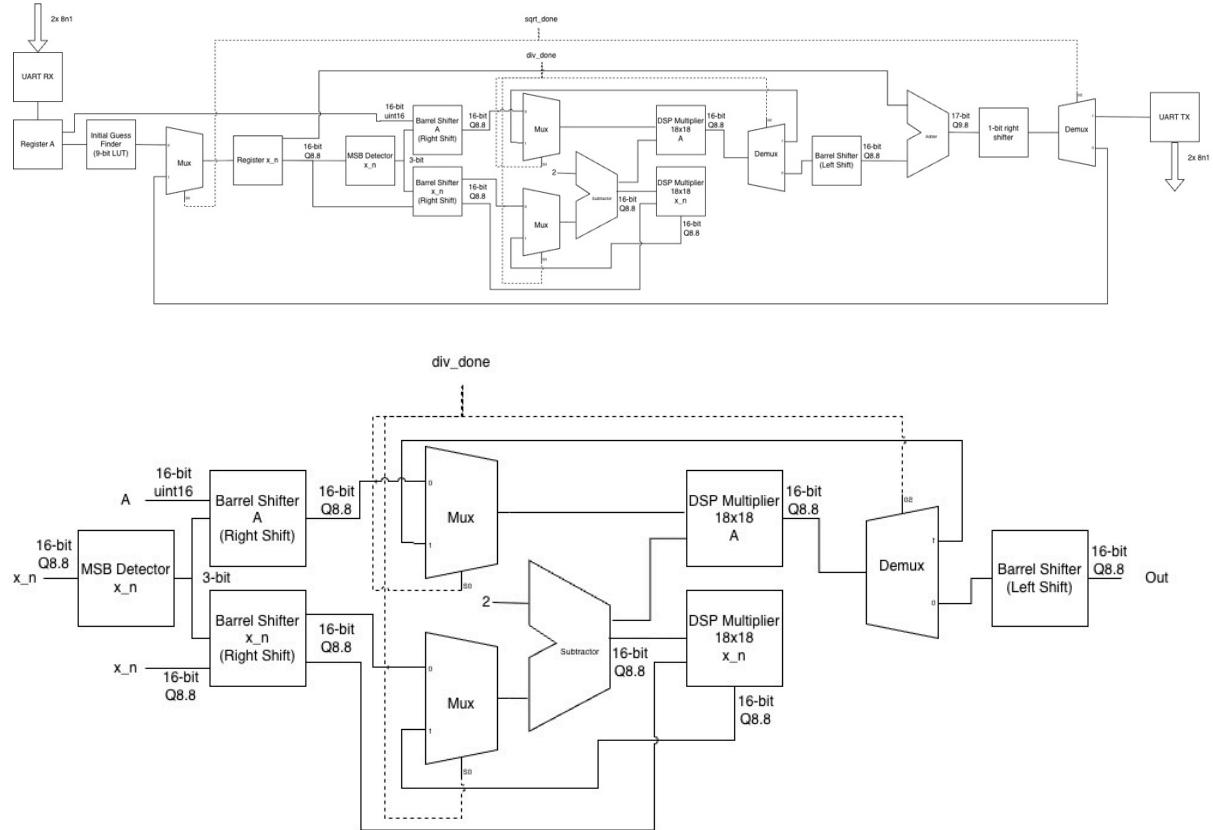
<i>Konsep Sistem</i> .....	1
<i>Rancangan Sistem</i> .....	2
Datapath.....	2
FSM .....	7

## Konsep Sistem

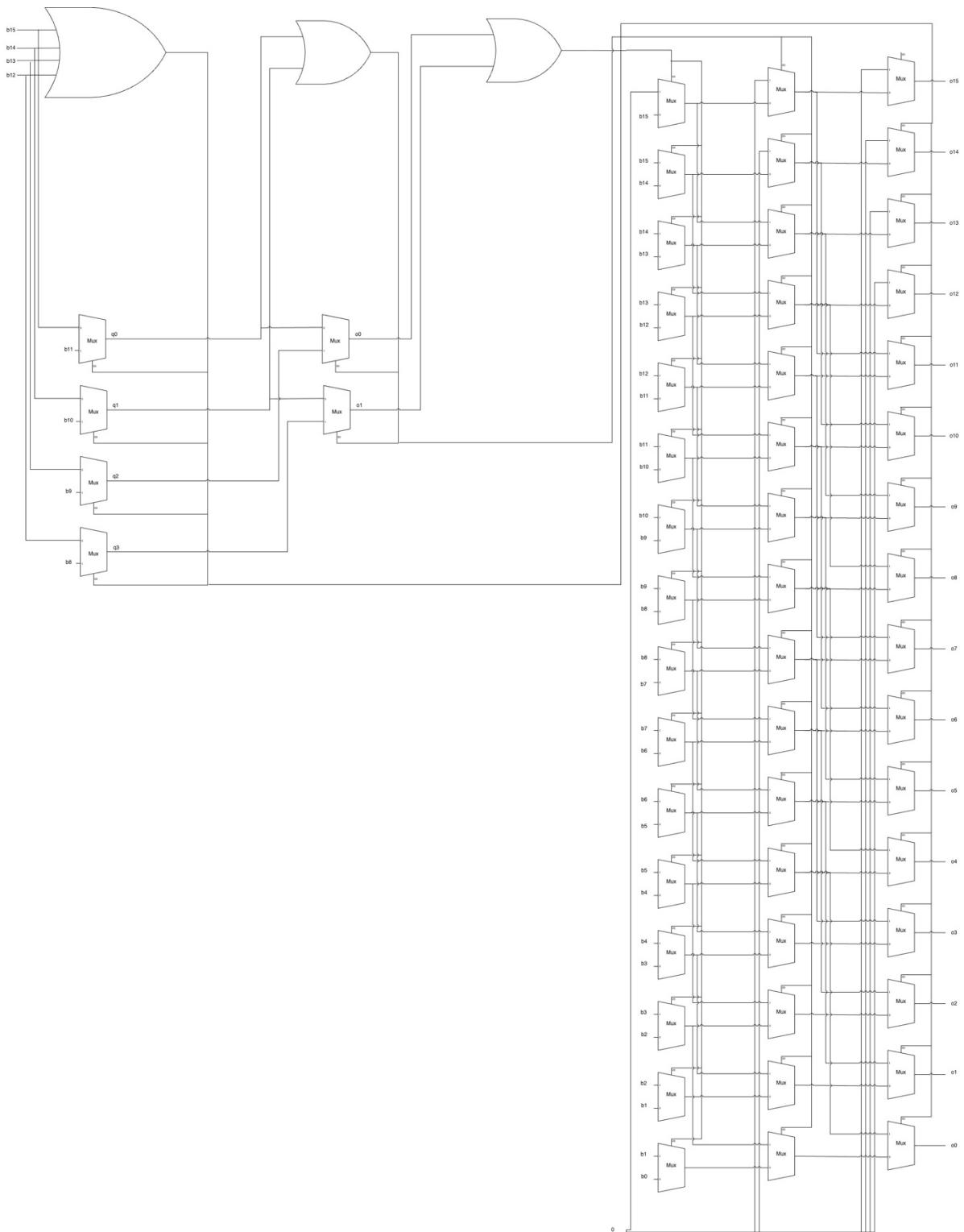
Sistem secara keseluruhan akan menerima input berupa 16-bit unsigned integer seperti aplikasi pada dunia nyata dengan ADC 16-bit. Sistem akan seringkali berada pada lebar data 16-bit dengan format U-Q8.8. Algoritma-algoritma yang dipakai adalah aproksimasi berdasarkan iterasi yakni Newton-Raphson untuk akar kuadrat sebagai struktur utama dan pembagi Goldschmidt untuk aproksimasi pembagian. Kedua algoritma tersebut menjamin kekonvergenan hasil dengan syarat-syarat tertentu. Selain itu, akan dialokasikan beberapa sumber daya komputasi seperti BRAM untuk implementasi LUT sebagai penunjang algoritma Newton Raphson. DSP multiplier 18x18 juga akan digunakan untuk membantu berjalannya komputasi algoritma Goldschmidt. Rangkaian MSB detector dengan pendekatan binary tree serta barrel shifter untuk penunjang fungsi normalisasi juga akan diimplementasikan dengan tujuan kebutuhan stage yang sedikit agar menghemat waktu komputasi. FSM akan diimplementasikan pada bagian UART RX, UART TX, dan kendali utama sistem dengan sistem control flags agar semua rangkaian sekuensial dapat berjalan sesuai alur yang diharapkan.

# Rancangan Sistem

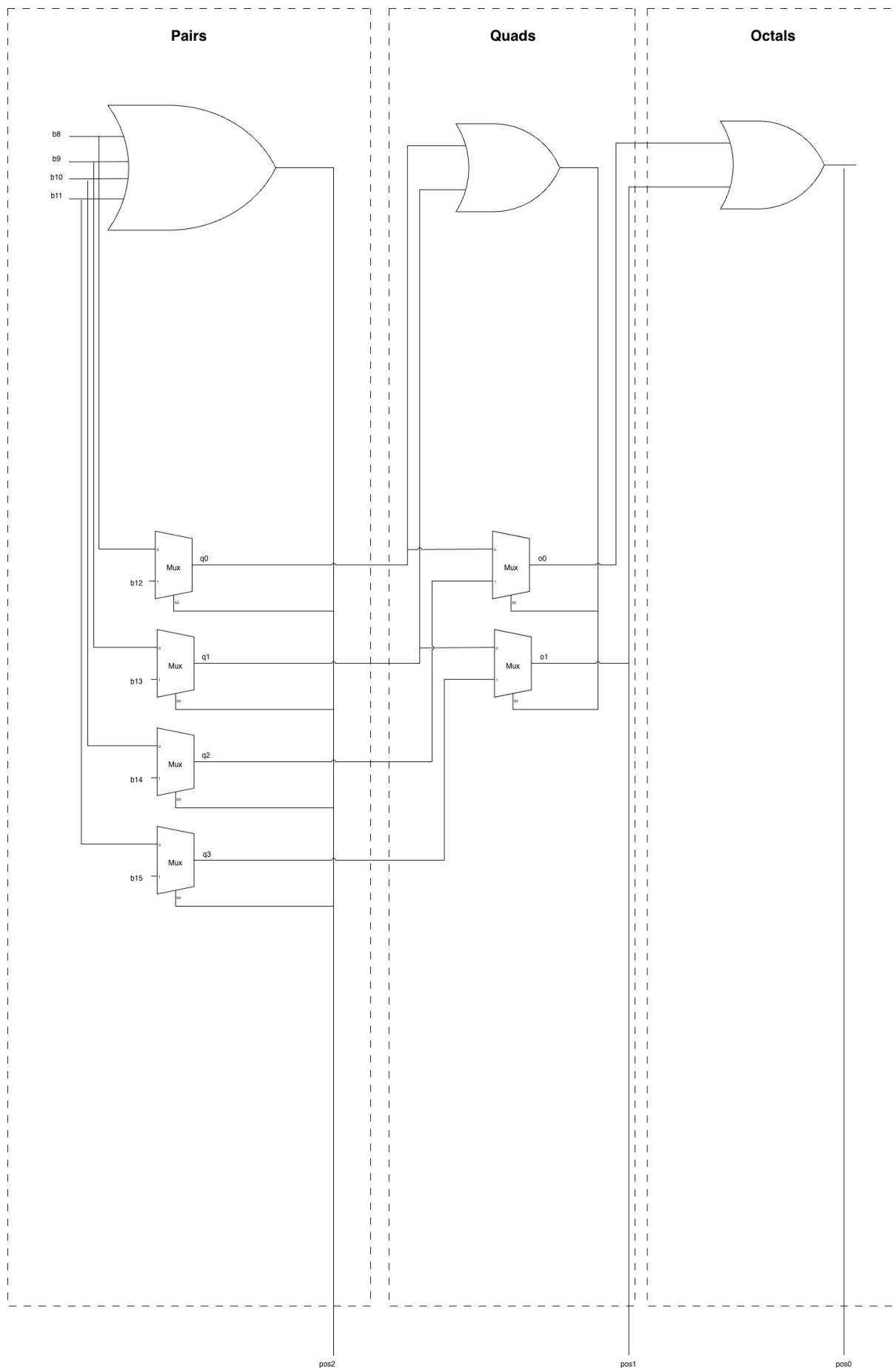
## Datapath



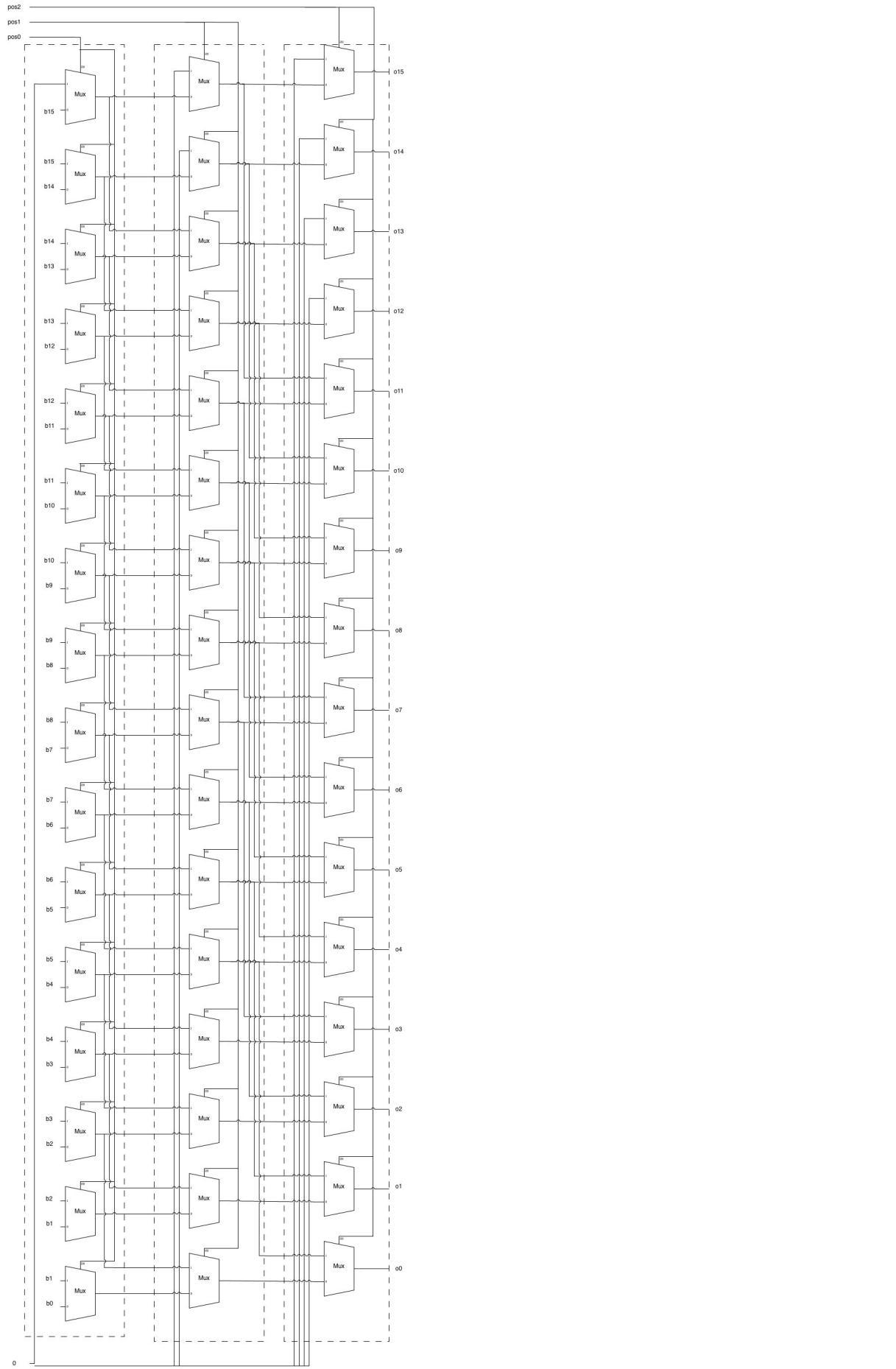
Pendekatan Goldscmidt digunakan karena sangat menghemat waktu komputasi dibandingkan restoring maupun non-restoring division. Selain itu, algoritma ini menjamin konvergensi ketika operand telah dinormalisasikan. Hal lain yang membuat algoritma ini diimplementasikan adalah karena penggunaan perkalian yang dapat menggunakan DSP  $18 \times 18$  yang terintegrasi dalam Cyclone IV.



Fungsi normalisasi di atas adalah gabungan dari MSB detector binary tree dan Barrel shifter kiri.

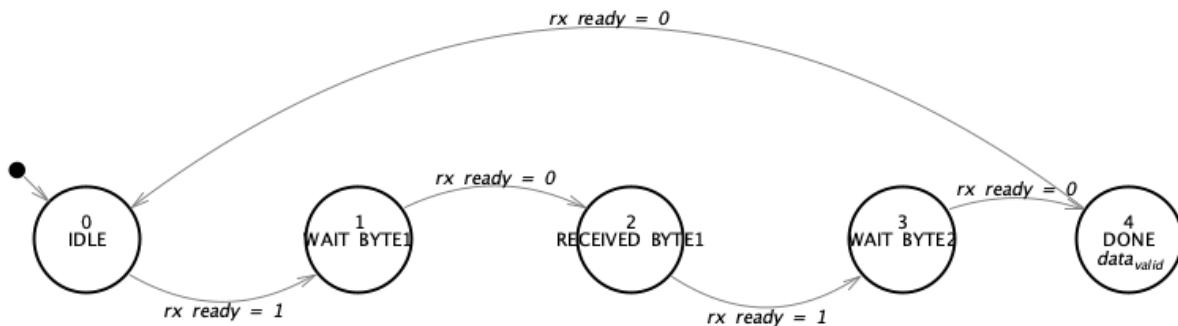


MSB detector dengan pendekatan binary tree digunakan karena menghemat stage yang ada sehingga delay dapat ditekan serta keluaran yang dihasilkan sangat berguna untuk menunjang berjalannya barrel shifter. Untuk blok ini, tidak ada masukan berupa angka 0 karena sudah ditangani di keadaan awal sistem dengan FSM (dijelaskan di bagian FSM).

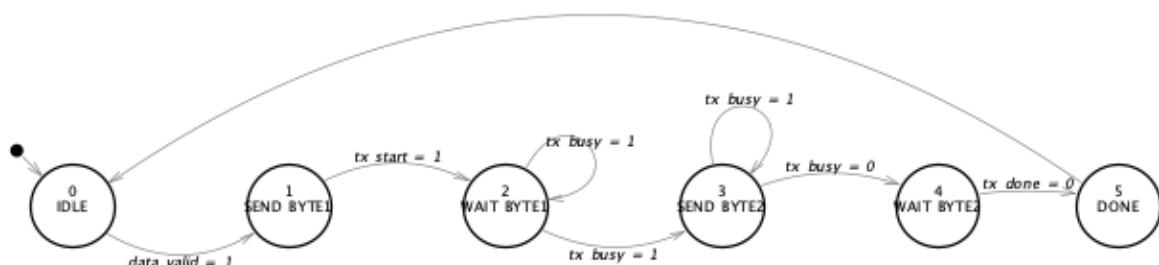


Pendekatan barrel shifter digunakan untuk menghemat waktu siklus yang digunakan untuk menggeser bit. Jika kita memakai cara naif, maka metode tersebut tidak memiliki tingkat skalabilitas yang tinggi. Berbeda dengan barrel shifter yang tiap stage menggeser bit sebesar pangkat dua.

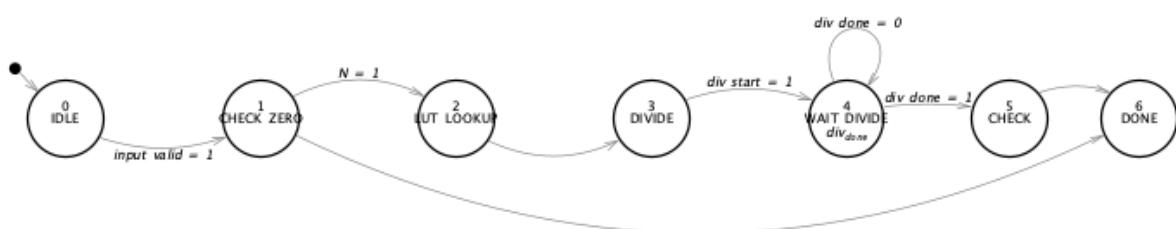
## FSM



FSM untuk penerimaan komunikasi UART memiliki dua state “tunggu” karena data yang ingin didapatkan memiliki lebar 16 bit atau 2 byte.



Sama juga untuk FSM pengiriman UART akan ada dua fasa tunggu karena UART hanya bisa mengirim 8-bit dalam satu kali perjalanan.



FSM kendali utama memiliki satu failsafe yang kuat yakni Check Zero. Fasa tersebut berguna karena akan menjaga sistem dari error akibat pembagian dengan nol atau pemborosan komputasi. Selain itu, terdapat fasa untuk mengecek jumlah interasi yang telah dilakukan