LAPORAN FINAL PROJECT PERANCANGAN KOMPONEN TERPROGRAM

A logo with a flower in the middle

Description automatically generated

Disusun Oleh :

Muhammad Hafidzh

5022221069

Dosen Mata Kuliah :

Dr. Rudy Dikairono, S.T., M.T.

Departemen Teknik Elektro

Fakultas Teknologi Elektro dan Informatika Cerdas

Institut Teknologi Sepuluh Nopember

* **Langkah**

Langkah pertama dalam pengerjaan final project ini adalah melakukan sampling sinyal ADC dari function generator ke FPGA, menggunakan modul Qsys bernama hello\_adc. Data hasil sampling berupa sinyal digital dengan resolusi 12-bit. Selanjutnya, data tersebut difilter menggunakan FIR low-pass filter, di mana koefisien filter telah dihitung sebelumnya menggunakan MATLAB. Setelah itu, data hasil filter kemudian dikirimkan secara paralel dalam format 12-bit ke Arduino Due. Pada Arduino Due, data digital tersebut dibaca dan diubah kembali menjadi sinyal analog melalui pin DAC untuk output.

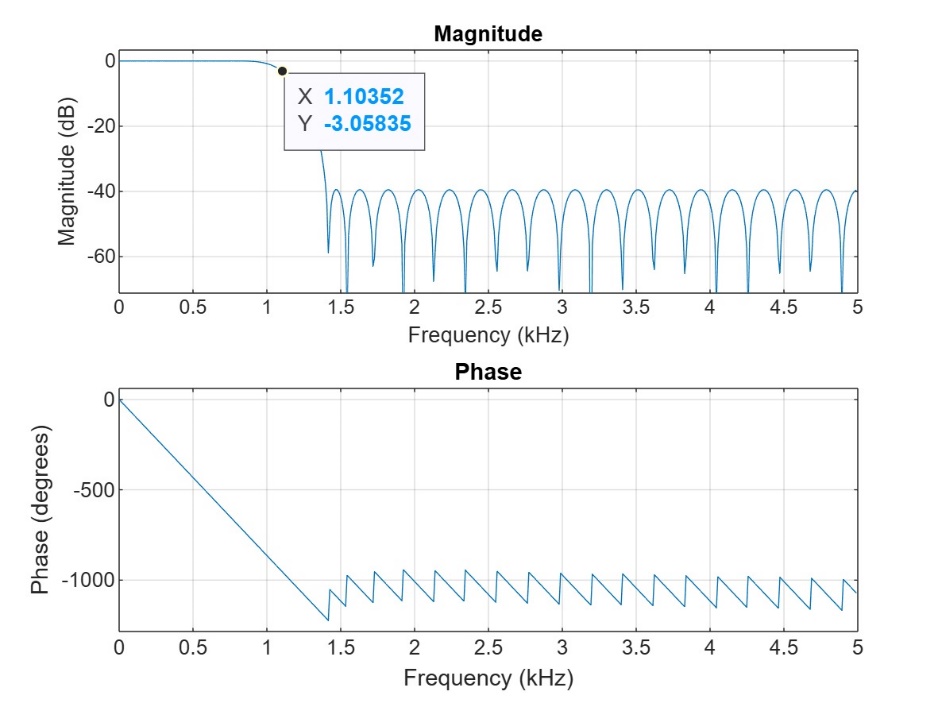
* **Hasil**

Plot Respon Frekuensi pada matlab

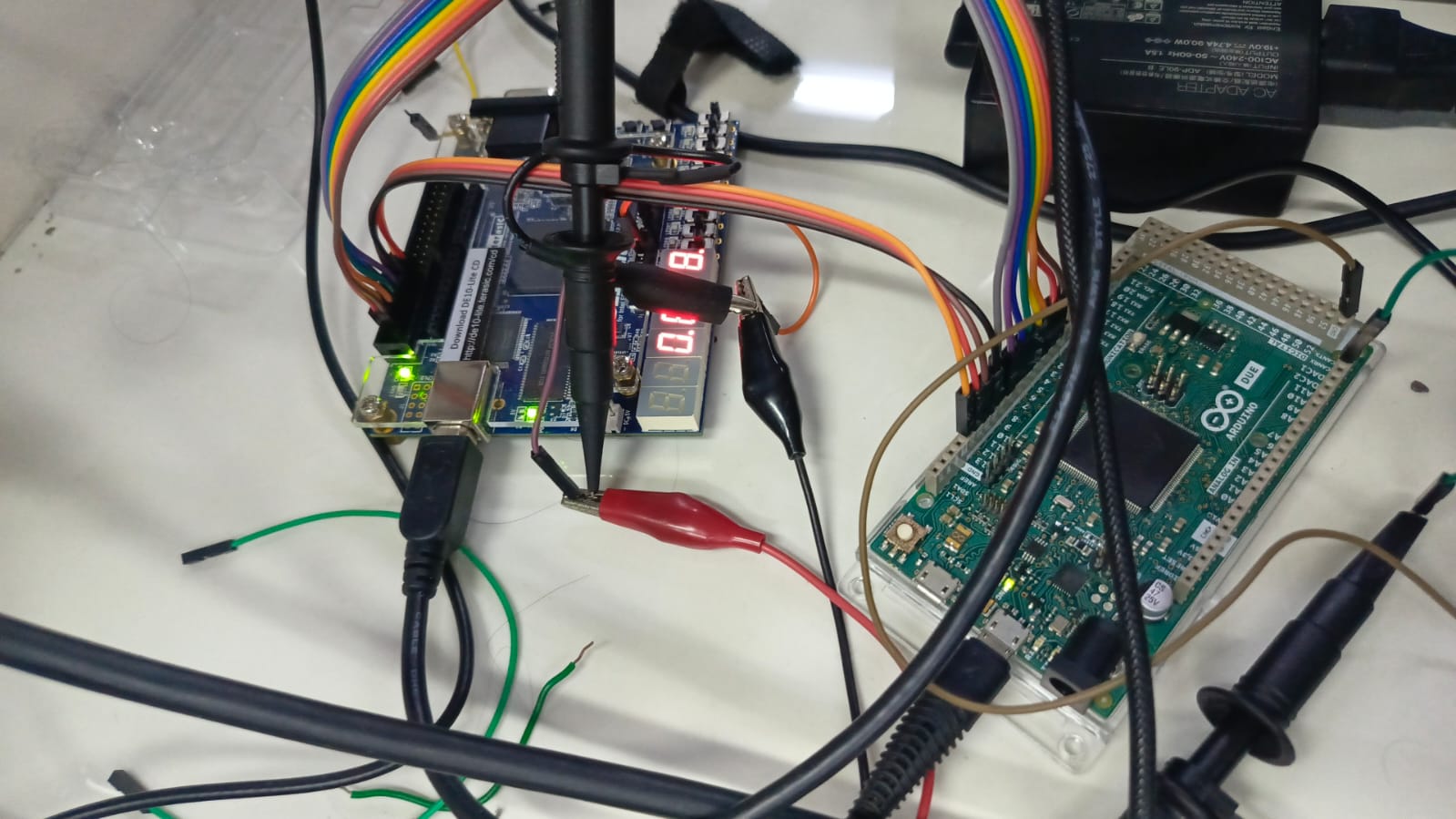
[n,fo,ao,w] = firpmord([860 1400],[1 0],[0.001 0.01],10000);

b = firpm(n,fo,ao,w);

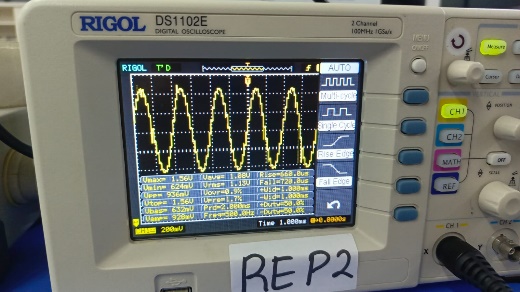
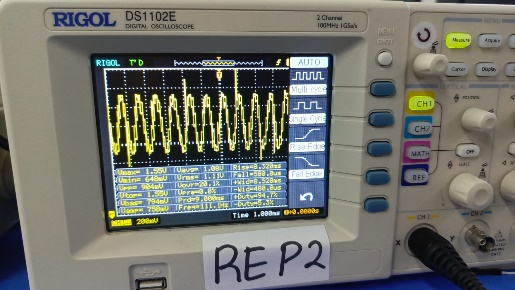
freqz(b,1,[],10000)

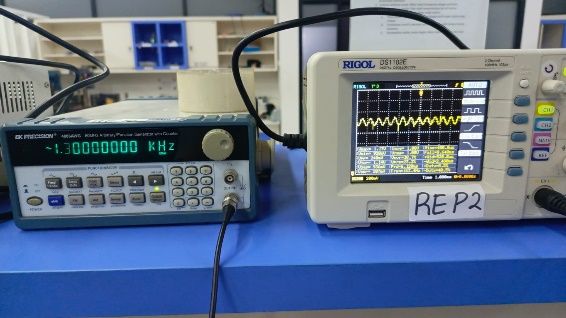
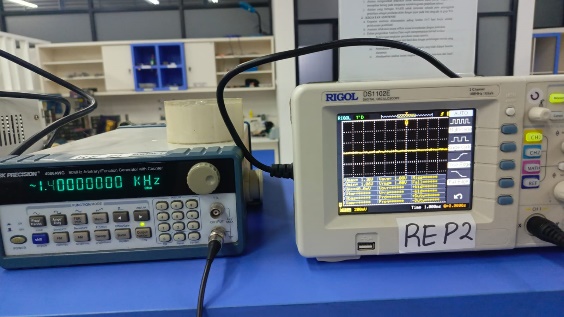


Rangkaian



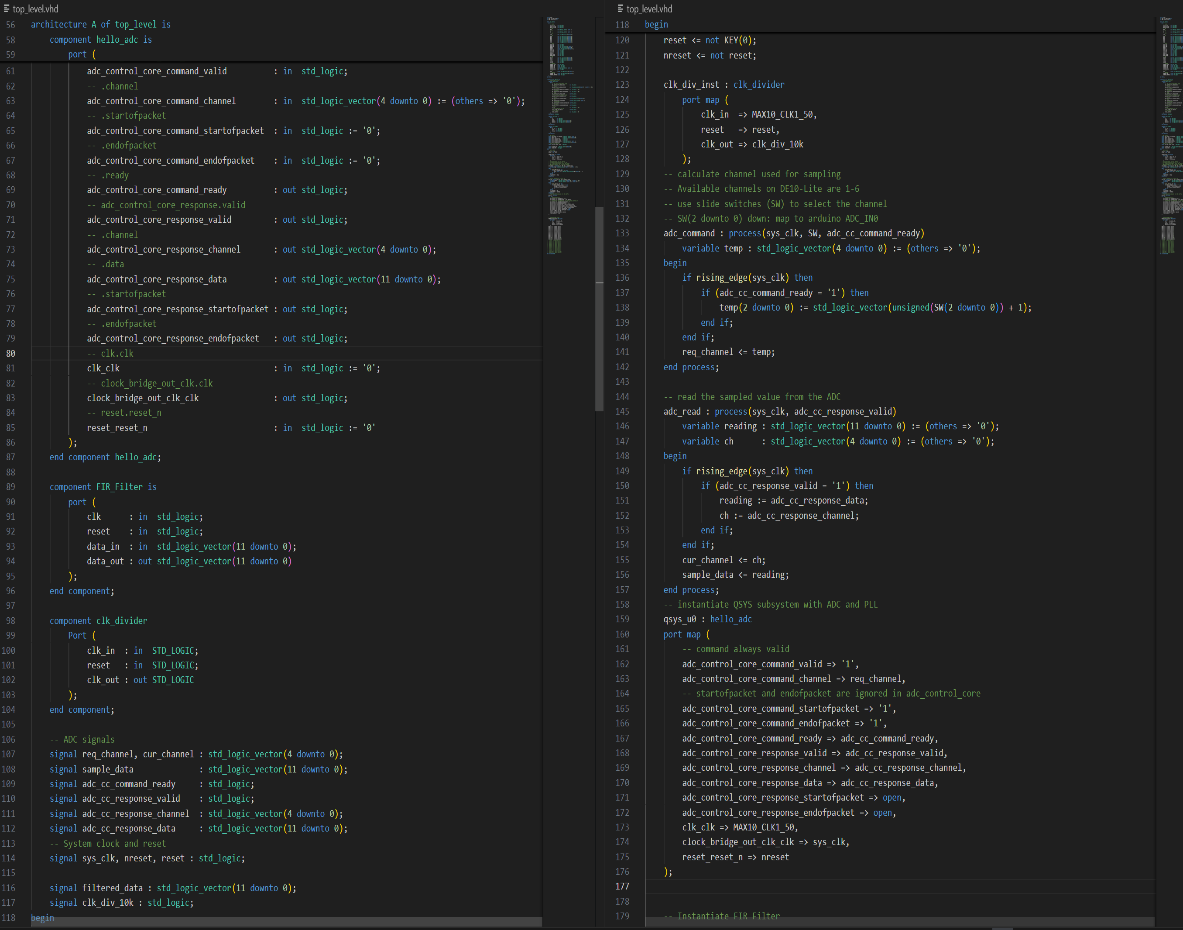
Hasil Filter dan DAC

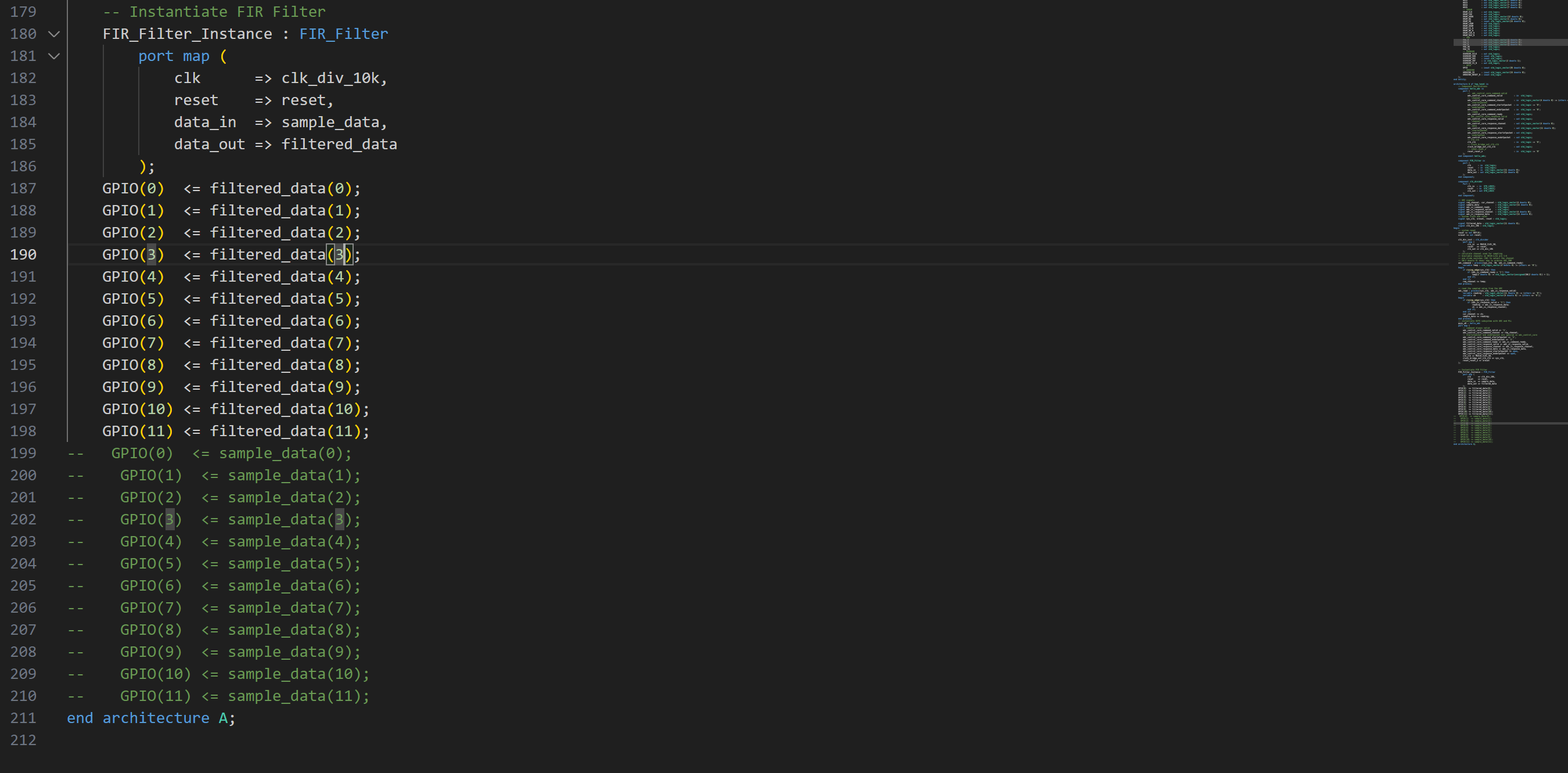
 

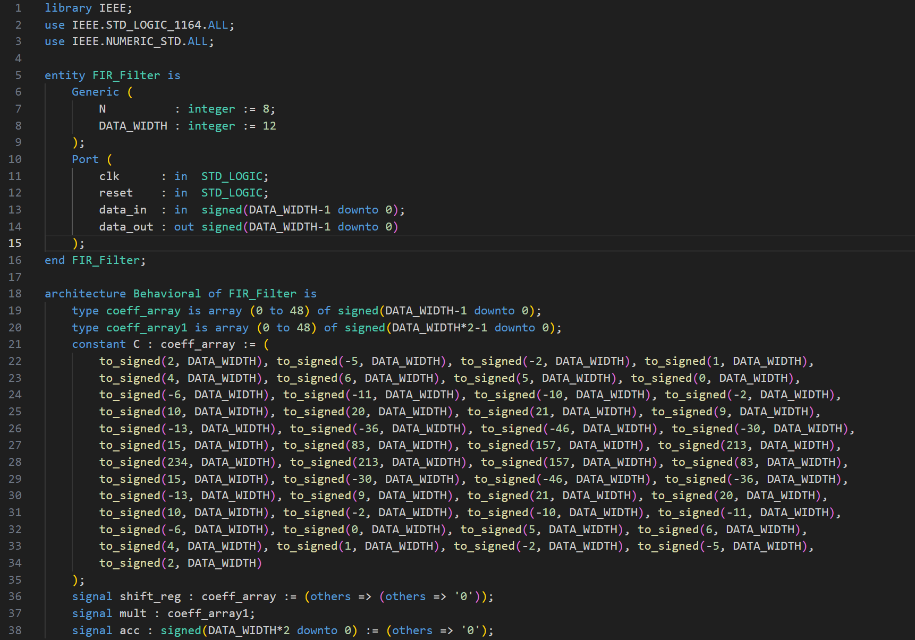
 

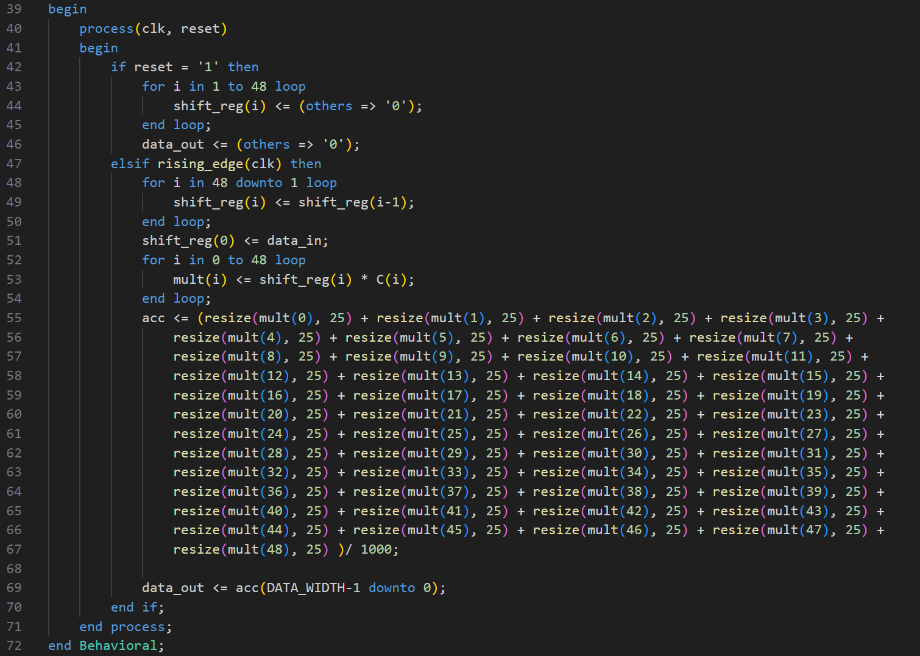
* VHDL

Top Level

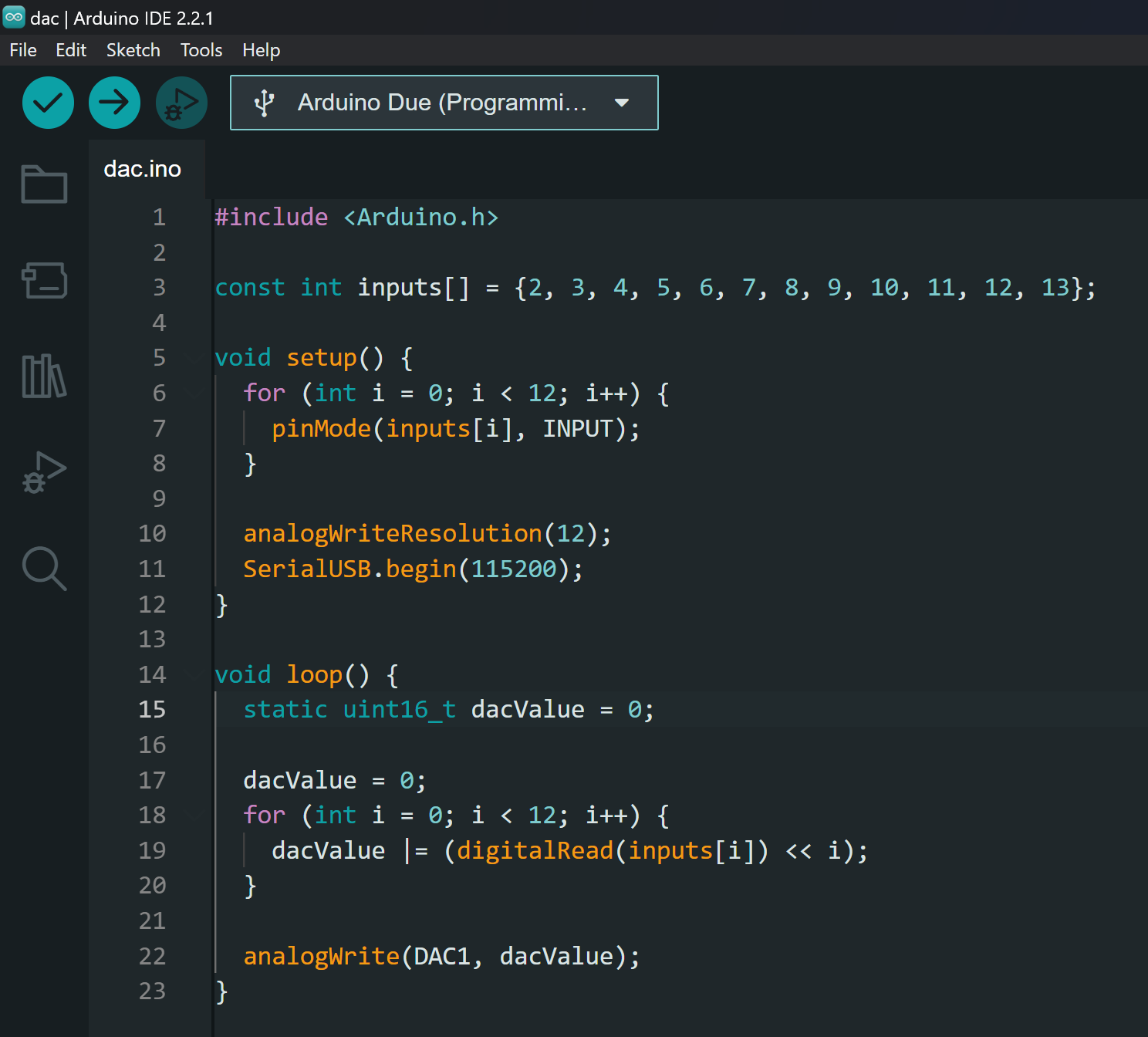








* Arduino



* Analisa

VHDL di atas merupakan implementasi dari sebuah Finite Impulse Response (FIR) lowpass filter digital dengan arsitektur berbasis shift register. Filter ini menerima masukan berupa data digital 12-bit dan menghasilkan keluaran yang juga berupa data 12-bit yang telah difilter. Koefisien filter yang telah didapatkan dari perhitungan matlab terdapat 49 orde. Setiap koefisien dalam array C merepresentasikan bobot yang digunakan untuk mengalikan data input yang tersimpan dalam shift register. Pada bagian top\_level, clock yang digunakan untuk filter ialah clock yang telah dibagi yakni menjadi 10KHz, dimana harus sesuai dengan frekuensi sampling saat menghitung koefisien filter pada matlab.

Proses filter dimulai dengan reset awal, di mana seluruh register geser (shift\_reg) diinisialisasi dengan nilai nol. Selanjutnya, saat sinyal clk mengalami rising edge, data input terbaru akan dimasukkan ke posisi paling awal dari register geser, sementara data lainnya digeser ke posisi berikutnya. Dengan cara ini, data input yang lebih lama secara bertahap bergerak ke ujung register. Proses ini menciptakan sebuah buffer yang menyimpan data input terbaru hingga data input yang telah diterima pada 48 siklus sebelumnya. Setelah data terbaru dimasukkan ke register, setiap nilai dalam register dikalikan dengan koefisien filter masing-masing menggunakan operasi perkalian. Hasil dari semua perkalian ini disimpan dalam array mult. Untuk mendapatkan hasil filter, semua nilai hasil perkalian dijumlahkan dalam sinyal akumulator (acc) setelah di-resize ke panjang data yang lebih besar untuk mencegah overflow. Akumulator ini membagi hasil penjumlahan dengan konstanta 1000 untuk mengatur skala keluaran. Hasil akhir dari proses ini adalah nilai keluaran filter (data\_out), yang diambil dari bit paling signifikan hingga bit paling tidak signifikan sesuai dengan lebar data yang didefinisikan.

Pada sisi Arduino, data digital hasil filter diterima dalam format paralel 12-bit dari FPGA melalui pin digital dan dikonversi menjadi sinyal analog menggunakan DAC internal. Proses ini dimulai dengan mengatur setiap pin sebagai input dan menginisialisasi resolusi DAC sebesar 12-bit. Data paralel dibaca bit-per-bit, disusun menjadi nilai 12-bit, dan dikeluarkan sebagai sinyal analog pada pin DAC1.