

Implementasi Filter Digital FIR dan IIR pada Mikrokontroler STM32 Cortex-M4 untuk Pengolahan Sinyal Sensor: Studi Simulasi Fixed-Point

Beni Satria^{1*}

¹Fakultas Sains dan Teknologi, Program Studi Teknik Elektro, Universitas Pembangunan Panca Budi, Medan, Indonesia

Email: ^{1*}benisatria@dosen.pancabudi.ac.id

(*Email Corresponding Author: benisatria@dosen.pancabudi.ac.id)

Received: 21 Juni 2026 | Revision: 19 Agustus 2026 | Accepted: 30 September 2026

Abstrak

Sinyal sensor yang dibaca mikrokontroler hampir selalu bercampur dengan derau acak dan interferensi jala-jala 50 Hz, sehingga perlu difilter di perangkat sebelum diolah lebih lanjut. Pada mikrokontroler tanpa unit titik mengambang atau ketika efisiensi memori dan energi menjadi prioritas, filter dijalankan dalam aritmetika fixed-point yang menimbulkan galat kuantisasi koefisien dan galat pembulatan. Penelitian ini membandingkan filter lolos-rendah FIR equiripple dan IIR eliptik dengan spesifikasi identik (frekuensi sampling 500 Hz, tepi pita lolos 20 Hz, tepi pita henti 40 Hz, riak 0,5 dB, atenuasi 60 dB) untuk mikrokontroler STM32 berinti ARM Cortex-M4. Implementasi dimodelkan secara bit-true mengikuti perilaku kernel CMSIS-DSP dalam format Q15, Q31, dan float32, kemudian diuji dengan sinyal sensor sintesis yang mengandung komponen 1,5–12 Hz, interferensi 50/150 Hz, derau putih, dan kuantisasi ADC 12-bit. Spesifikasi dipenuhi oleh FIR 59 tap dan IIR orde 5 (tiga seksi orde-dua). Pada Q15, FIR mencapai SQNR 81,4 dB dan SNR keluaran 31,65 dB, praktis sama dengan referensi float64. IIR Q15 tanpa penskalaan hanya mencapai SQNR 28,1 dB dan SNR keluaran 26,68 dB; penskalaan norma L_∞ antarseksi menaikkan SQNR menjadi 47,0 dB dan SNR keluaran menjadi 32,20 dB. Realisasi IIR bentuk langsung orde tinggi dengan koefisien terkuantisasi menempatkan kutub pada lingkaran satuan, sedangkan realisasi seksi orde-dua tetap stabil. IIR membutuhkan 15 MAC per sampel dan memori 60 byte (Q15), dibandingkan 59 MAC dan 298 byte pada FIR, tetapi fasenya tidak linear dengan tunda grup 27–102 ms pada pita lolos, sedangkan FIR memiliki tunda konstan 58 ms. Hasil ini memberi pedoman pemilihan struktur dan format bilangan filter pada mikrokontroler STM32 sebelum diimplementasikan pada perangkat keras.

Kata Kunci: CMSIS-DSP, Filter FIR, Filter IIR, Fixed-Point, Kuantisasi, STM32

Abstract

Microcontrollers that read sensors almost always receive random noise and 50 Hz power-line interference along with the signal, so the signal must be filtered on the device. Without a floating-point unit, or when memory and energy are tight, the filter runs in fixed-point arithmetic, which adds coefficient quantization and round-off errors. This study compares an equiripple FIR and an elliptic IIR low-pass filter for an STM32 microcontroller with an ARM Cortex-M4 core. Both filters share one specification: a 500 Hz sampling rate, a 20 Hz passband edge, a 40 Hz stopband edge, 0.5 dB ripple, and 60 dB attenuation. Bit-true models reproduced the behavior of CMSIS-DSP kernels in Q15, Q31, and float32. The test signal combined 1.5–12 Hz components, 50/150 Hz interference, white noise, and 12-bit ADC quantization. A 59-tap FIR and a fifth-order IIR with three second-order sections met the specification. In Q15, the FIR reached an SQNR of 81.4 dB and an output SNR of 31.65 dB, equal to the float64 reference. The unscaled Q15 IIR reached only 28.1 dB SQNR and 26.68 dB output SNR. L_∞ -norm scaling between sections raised these values to 47.0 dB and 32.20 dB. A high-order direct-form IIR with quantized coefficients placed a pole on the unit circle, whereas the cascade of second-order sections remained stable. The Q15 IIR needed 15 MACs per sample and 60 bytes of memory; the Q15 FIR needed 59 MACs and 298 bytes. The IIR, however, has a nonlinear phase with a passband group delay of 27–102 ms, while the FIR delays every component by a constant 58 ms. These results guide the choice of filter structure and number format on STM32 microcontrollers before hardware implementation.

Keywords: CMSIS-DSP, FIR Filter, IIR Filter, Fixed-Point, Quantization, STM32

1. PENDAHULUAN

Sistem tertanam berbasis sensor saat ini semakin banyak melakukan pengolahan data secara langsung pada perangkat (edge processing) tanpa harus mengirim seluruh data mentah ke komputer atau server[1]. Pendekatan tersebut dapat mengurangi volume data yang dikirim, menekan konsumsi energi, serta mendukung pemrosesan data yang lebih cepat. Mikrokontroler kelas menengah dengan inti ARM Cortex-M4 juga telah digunakan untuk menjalankan berbagai proses pengolahan sinyal digital menggunakan pustaka CMSIS-DSP. Kemampuan tersebut menjadikan mikrokontroler tidak hanya berfungsi sebagai pengambil data sensor, tetapi juga sebagai perangkat yang mampu melakukan pemrosesan sinyal sebelum data digunakan pada tahap berikutnya[2].

Salah satu tahapan penting dalam pengolahan sinyal sensor adalah proses penyaringan atau filtering. Sinyal yang diperoleh dari sensor umumnya tidak hanya mengandung informasi yang diinginkan, tetapi juga bercampur dengan derau

acak dan interferensi dari lingkungan. Salah satu gangguan yang umum ditemukan adalah interferensi jala-jala listrik pada frekuensi 50 Hz atau 60 Hz. Oleh karena itu, filter digital diperlukan untuk melewatkan komponen frekuensi yang dibutuhkan sekaligus meredam komponen yang tidak diinginkan. Beberapa penelitian telah menggunakan berbagai pendekatan untuk mengurangi interferensi tersebut, seperti filter sinkron, filter takik berbasis FFT, dan filtrasi takik kuasi waktu nyata. Penelitian pada sistem fotopletismografi juga telah menggabungkan filter analog dan digital, tetapi pemrosesan filter digital dilakukan menggunakan MATLAB pada komputer sehingga belum secara khusus membahas keterbatasan implementasi pada mikrokontroler[3].

Dua jenis filter digital yang banyak digunakan adalah Finite Impulse Response (FIR) dan Infinite Impulse Response (IIR). FIR merupakan filter yang tidak menggunakan umpan balik sehingga keluaran hanya bergantung pada sampel masukan saat ini dan sebelumnya[4]. Karakteristik tersebut membuat FIR selalu stabil dan memungkinkan perancangan fase linear ketika koefisiennya simetris[5]. Namun, FIR umumnya membutuhkan orde yang lebih tinggi ketika harus menghasilkan pita transisi yang sempit. Sebaliknya, IIR menggunakan umpan balik sehingga dapat mencapai selektivitas frekuensi yang sama dengan orde yang lebih rendah. Keuntungan tersebut membuat IIR lebih hemat operasi komputasi dan memori, tetapi struktur rekursifnya membuat filter lebih sensitif terhadap perubahan koefisien akibat kuantisasi.

Permasalahan tersebut menjadi lebih penting ketika filter dijalankan pada mikrokontroler menggunakan aritmetika fixed-point[6]. Pada format bilangan seperti Q15 dan Q31, jumlah bit yang digunakan untuk merepresentasikan nilai memiliki batas tertentu sehingga proses kuantisasi koefisien dan pembulatan dapat menghasilkan galat[7]. Pengaruh galat tersebut pada FIR dan IIR tidak selalu sama[8]. FIR tidak mempunyai jalur umpan balik sehingga galat kuantisasi relatif tidak terakumulasi, sedangkan IIR dapat mengalami pengaruh yang lebih besar karena galat pada setiap seksi dapat diteruskan dan diperkuat oleh struktur rekursif. Penelitian sebelumnya menunjukkan bahwa struktur biquad atau Second-Order Sections (SOS) memiliki ketahanan numerik yang berbeda terhadap kesalahan perkalian dan kuantisasi. Urutan seksi serta pasangan kutub dan nol juga dapat memengaruhi respons filter setelah koefisien mengalami kuantisasi[9].

Dalam konteks mikrokontroler STM32 Cortex-M4, permasalahan tersebut berkaitan langsung dengan format bilangan dan pustaka pemrosesan sinyal yang tersedia[10]. STM32F4 menggunakan inti ARM Cortex-M4 yang memiliki instruksi DSP, termasuk operasi Multiply Accumulate (MAC) dan instruksi SIMD yang dapat meningkatkan efisiensi pengolahan data. Pustaka CMSIS-DSP menyediakan implementasi filter FIR dan biquad dalam beberapa format, antara lain Q15, Q31, dan float32[11]. Dengan demikian, pemilihan struktur filter tidak hanya ditentukan oleh karakteristik respons frekuensi, tetapi juga oleh kebutuhan presisi, memori, jumlah operasi, serta kemampuan aritmetika mikrokontroler yang digunakan[12].

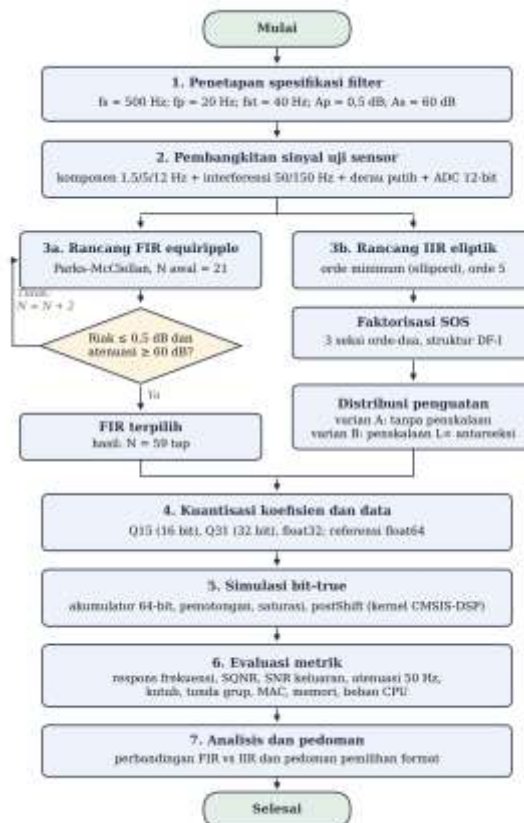
Meskipun penelitian sebelumnya telah membahas implementasi filter digital pada STM32 maupun pengaruh kuantisasi pada struktur IIR, masih terdapat celah penelitian dalam membandingkan FIR dan IIR dengan spesifikasi filter yang sama secara bit-true pada platform mikrokontroler Cortex-M4. Beberapa penelitian pada STM32 berfokus pada pengukuran kinerja implementasi perangkat keras, sedangkan penelitian lain membahas ketahanan struktur IIR atau perancangan filter fixed-point pada FPGA dan ASIC[13]. Perbedaan karakteristik platform menyebabkan hasil tersebut belum secara langsung memberikan gambaran mengenai pengaruh format Q15, Q31, dan float32, distribusi penguatan antarseksi, serta kebutuhan komputasi dan memori pada implementasi filter untuk STM32[14].

Berdasarkan celah tersebut, penelitian ini membandingkan filter lolos-rendah FIR equiripple dan IIR eliptik dengan spesifikasi yang identik pada mikrokontroler STM32 berinti ARM Cortex-M4. Perbandingan dilakukan melalui simulasi bit-true yang meniru perilaku kernel FIR dan biquad pada CMSIS-DSP dalam format Q15, Q31, dan float32. Penelitian juga menguji pengaruh penskalaan antarseksi pada IIR Q15 untuk mengetahui sejauh mana distribusi penguatan memengaruhi galat kuantisasi dan kualitas keluaran. Evaluasi dilakukan berdasarkan SQNR, SNR keluaran, atenuasi interferensi 50 Hz, kestabilan kutub, tunda grup, jumlah operasi MAC, kebutuhan memori, dan estimasi beban CPU.

Dengan pendekatan tersebut, penelitian ini diharapkan dapat memberikan dasar pemilihan struktur dan format bilangan filter yang sesuai untuk pengolahan sinyal sensor pada mikrokontroler STM32 sebelum dilakukan implementasi pada perangkat keras. Secara khusus, penelitian ini bertujuan untuk mengetahui perbedaan kinerja FIR dan IIR pada berbagai format bilangan, mengidentifikasi pengaruh kuantisasi dan penskalaan terhadap kualitas sinyal, serta membandingkan kebutuhan komputasi dan memori dari kedua struktur filter[15].

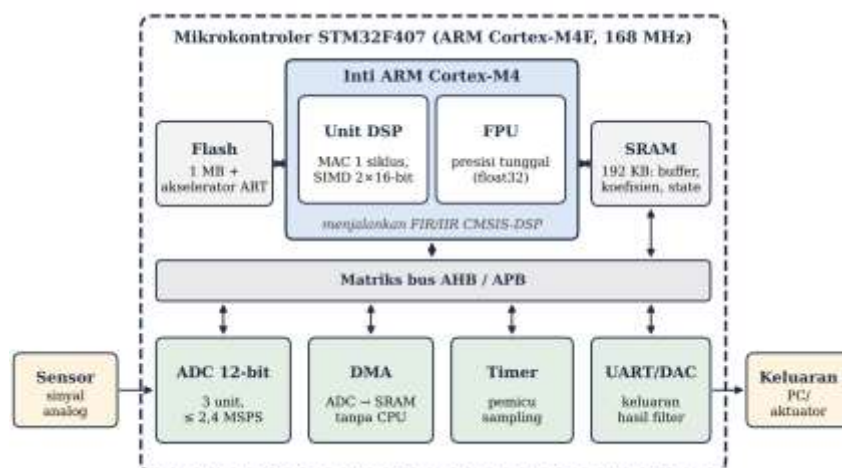
2. METODOLOGI PENELITIAN

Seluruh percobaan berupa simulasi numerik dengan Python (NumPy dan SciPy). Gambar 3 merangkum tujuh tahap penelitian. Cabang FIR memuat perulangan penambahan panjang filter sampai spesifikasi terpenuhi, sedangkan cabang IIR menghasilkan dua varian distribusi penguatan yang diuji terpisah pada tahap kuantisasi dan simulasi.



Gambar 1. Diagram Alir Penelitian

Platform target adalah mikrokontroler STM32F407 berinti ARM Cortex-M4F dengan frekuensi detak 168 MHz. Inti ini memiliki instruksi MAC 32-bit satu siklus, instruksi SIMD yang mengalikan dua pasang data 16-bit sekaligus, dan unit titik mengambang presisi tunggal. Gambar 4 menunjukkan alur data yang dimodelkan. ADC 12-bit mencuplik sinyal sensor atas pemicu timer, DMA memindahkan sampel ke SRAM tanpa membebani CPU, lalu inti Cortex-M4 menjalankan fungsi filter CMSIS-DSP per blok sampel. Koefisien dan *state* filter tersimpan di SRAM, dan hasilnya dikirim melalui UART atau DAC.



Gambar 2. Arsitektur STM32F407 dan Alur Data Pemfilteran yang Dimodelkan

2.1 Spesifikasi Filter dan Sinyal Uji

Filter yang dirancang adalah filter lolos-rendah untuk sinyal sensor berfrekuensi rendah, misalnya sensor regangan, percepatan, atau biopotensial yang telah dinormalisasi ke rentang ± 1 . Parameter spesifikasi dan sinyal uji disajikan pada Tabel 2. Sinyal uji berdurasi 20 s terdiri atas tiga komponen informasi (1,5 Hz, 5 Hz, dan 12 Hz), interferensi jala-jala 50 Hz beserta harmonik ketiga 150 Hz, serta derau putih Gaussian. Sinyal kemudian dikuantisasi dengan resolusi ADC 12-

bit seperti ADC internal STM32F4. SNR masukan terhadap sinyal bersih adalah 4,40 dB. Detik pertama keluaran dibuang dari perhitungan agar transien awal tidak memengaruhi metrik.

Tabel 1. Parameter Spesifikasi Filter dan Sinyal Uji

Parameter	Nilai
Frekuensi sampling f_s	500 Hz
Tepi pita lolos / pita henti	20 Hz / 40 Hz
Riak pita lolos maksimum A_p	0,5 dB
Atenuasi pita henti minimum A_s	60 dB
Komponen informasi (amplitudo)	1,5 Hz (0,30); 5 Hz (0,15); 12 Hz (0,08)
Interferensi (amplitudo)	50 Hz (0,20); 150 Hz (0,05)
Derau putih (simpangan baku)	0,02
Resolusi ADC	12 bit
Panjang blok pemrosesan	32 sampel

2.2 Perancangan Filter FIR dan IIR

Filter FIR dirancang dengan algoritma Parks–McClellan (*equiripple*) dan direalisasikan sesuai persamaan (3). Panjang filter N dinaikkan bertahap dari 21 sampai spesifikasi riak dan atenuasi terpenuhi. Filter IIR dirancang dengan aproksimasi eliptik karena menghasilkan orde terendah untuk spesifikasi yang sama, lalu difaktorkan menjadi kaskade SOS sesuai persamaan (7). Setiap seksi dijalankan dengan fungsi *arm_biquad_cascade_df1* pada CMSIS-DSP yang mengikuti persamaan (8). Struktur DF-I dipilih untuk format fixed-point karena seluruh hasil kali dijumlahkan dalam satu akumulator, sehingga hanya terjadi satu pembulatan per seksi.

A. Filter FIR

FIR (*finite impulse response*) adalah filter yang respons impulsnya berhingga. Seluruh koefisien a_k pada persamaan (1) bernilai nol, sehingga keluaran hanya bergantung pada sampel masukan saat ini dan sampel sebelumnya. Persamaan beda FIR menjadi konvolusi (3) dengan fungsi alih (4), dengan N panjang filter atau jumlah *tap*.

$$y[n] = \sum_{k=0}^{N-1} h[k] x[n-k] \quad (3)$$

$$H(z) = \sum_{k=0}^{N-1} h[k] z^{-k} \quad (4)$$

Gambar 1(a) memperlihatkan struktur bentuk langsung FIR. Sampel masukan melewati rantai elemen tunda z^{-1} , setiap sampel tertunda dikalikan satu koefisien, lalu seluruh hasil kali dijumlahkan. Satu sampel keluaran membutuhkan N operasi MAC. FIR tidak memiliki umpan balik dan seluruh kutubnya berada di titik asal, sehingga filter ini selalu stabil. Bila koefisiennya simetris, fasenya linear sesuai persamaan (5): semua komponen frekuensi tertunda $(N-1)/2$ sampel dan bentuk gelombang tidak terdistorsi. Kekurangannya, pita transisi yang sempit menuntut N yang besar.

$$h[k] = h[N-1-k] \Rightarrow \angle H(e^{j\omega}) = -\frac{N-1}{2} \omega \quad (5)$$

B. Filter IIR

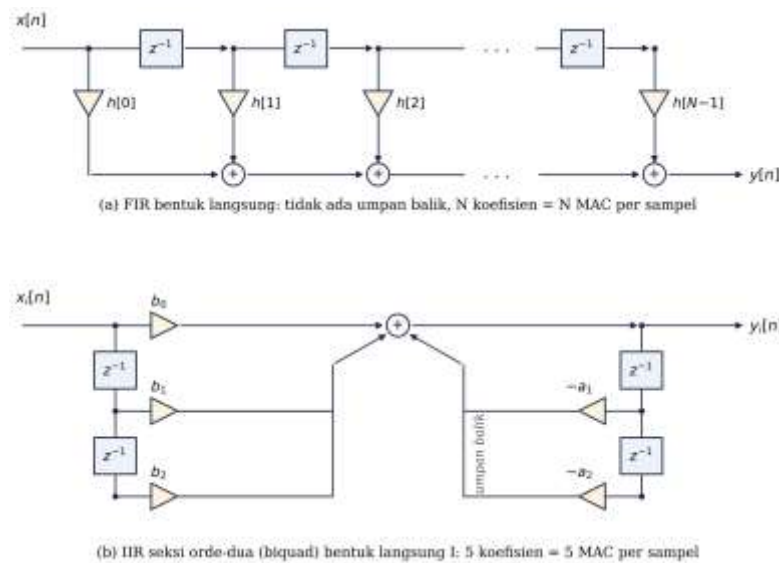
IIR (*infinite impulse response*) adalah filter yang respons impulsnya tak berhingga. Sebagian koefisien a_k tidak nol, sehingga keluaran sebelumnya diumpankan kembali ke perhitungan. Akibat umpan balik ini, respons impuls tidak pernah tepat nol, tetapi meluruh secara eksponensial bila filter stabil. Filter IIR kausal stabil jika seluruh kutubnya berada di dalam lingkaran satuan, sesuai persamaan (6).

$$|p_i| < 1, \quad A(p_i) = 1 + \sum_{k=1}^L a_k p_i^{-k} = 0, \quad i = 1, \dots, L \quad (6)$$

Keunggulan IIR adalah selektivitas tinggi dengan orde rendah. Kelemahannya, fasenya tidak linear dan filter ini peka terhadap kuantisasi: pergeseran kecil koefisien a_k dapat memindahkan kutub yang dekat lingkaran satuan ke luar lingkaran. Karena itu, IIR orde tinggi lazim difaktorkan menjadi kaskade K seksi orde-dua (SOS) sesuai persamaan (7). Setiap seksi, yang juga disebut biquad, dihitung dengan persamaan beda (8) menggunakan struktur DF-I pada Gambar 1(b). Satu seksi membutuhkan lima operasi MAC per sampel.

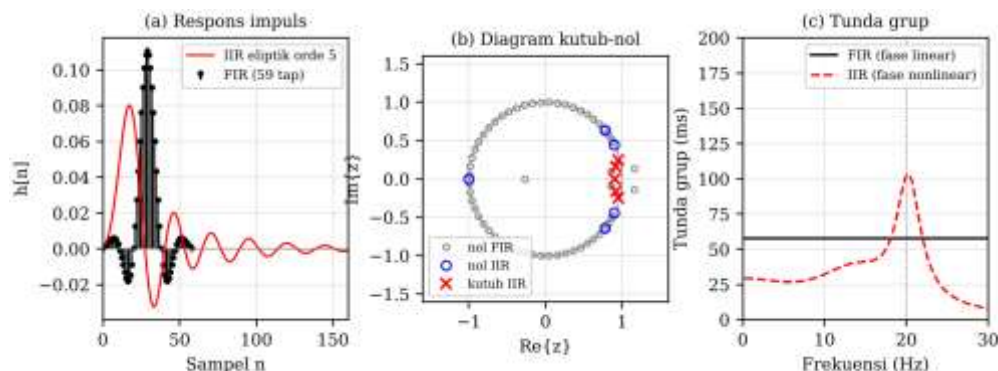
$$H(z) = \prod_{i=1}^K \frac{b_{0i} + b_{1i}z^{-1} + b_{2i}z^{-2}}{1 + a_{1i}z^{-1} + a_{2i}z^{-2}} \quad (7)$$

$$y_i[n] = b_{0i}x_i[n] + b_{1i}x_i[n-1] + b_{2i}x_i[n-2] - a_{1i}y_i[n-1] - a_{2i}y_i[n-2] \quad (8)$$



Gambar 3. Struktur Realisasi Filter: (a) FIR Bentuk Langsung, (b) Seksi Orde-Dua IIR Bentuk Langsung

Gambar 3 membandingkan kedua kelas filter menggunakan filter yang dirancang dalam penelitian ini (Subbab 4.1). Respons impuls FIR berhenti tepat setelah 59 sampel dan simetris terhadap sampel ke-29. Respons impuls IIR berosilasi dan meluruh tanpa titik akhir; amplitudonya masih sekitar $1,4 \times 10^{-3}$ setelah 150 sampel. Kelima kutub IIR berada di dalam lingkaran satuan dengan jari-jari 0,903–0,977, sedangkan nol-nolnya tepat pada lingkaran satuan dan menghasilkan titik redaman total di pita henti. Nol FIR yang tidak terletak pada lingkaran satuan muncul berpasangan resiprokal (z dan $1/z$), ciri khas FIR fase linear. Panel tunda grup memperlihatkan akibatnya: FIR menunda semua frekuensi 58 ms, sedangkan tunda IIR berubah dari 27 ms hingga 102 ms di dalam pita lolos.



Gambar 4. Karakteristik FIR 59 Tap dan IIR Eliptik Orde 5: (a) Respons Impuls, (b) Diagram Kutub-Nol, (c) Tunda Grup

2.3 Mikrokontroler STM32 Cortex-M4 dan CMSIS-DSP

STM32 adalah keluarga mikrokontroler 32-bit buatan STMicroelectronics. Seri STM32F4 memakai inti ARM Cortex-M4 berarsitektur ARMv7E-M, yang menambahkan instruksi DSP pada set instruksi Thumb-2. Instruksi tersebut mencakup MAC 32-bit satu siklus dan instruksi SIMD seperti SMLAD, yang mengalikan dua pasang data 16-bit lalu menjumlahkan hasilnya dalam satu instruksi. Varian Cortex-M4F, misalnya STM32F407, juga dilengkapi FPU presisi tunggal.

ARM menyediakan pustaka CMSIS-DSP yang berisi fungsi FIR dan biquad untuk format Q15, Q31, dan float32. Q15 dan Q31 adalah bilangan pecahan fixed-point bertanda 16 dan 32 bit (Subbab 3.3). Instruksi SIMD membuat FIR Q15 dapat menghitung dua MAC sekaligus, sedangkan FPU membuat versi float32 berjalan tanpa emulasi perangkat lunak. Pengujian pada papan STM32 menunjukkan bahwa FIR dan IIR dapat dijalankan secara waktu nyata pada keluarga mikrokontroler ini.

2.4 Representasi Fixed-Point dan Model Bit-True

Format $Qm.n$ menyatakan bilangan bertanda dengan panjang kata B bit. Pada format Q15 ($B = 16$) dan Q31 ($B = 32$) seluruh bit selain bit tanda adalah bit pecahan, sehingga nilai v yang direpresentasikan oleh bit d_j mengikuti persamaan (9).

$$v = -d_{B-1} + \sum_{j=0}^{B-2} d_j 2^{j-(B-1)}, \quad -1 \leq v \leq 1 - 2^{-(B-1)} \quad (9)$$

Koefisien c dikuantisasi dengan pembulatan ke kelipatan terdekat dan dibatasi (saturasi) pada rentang format sesuai persamaan (10). Dengan asumsi galat pembulatan terdistribusi seragam, variansi derau kuantisasi dengan langkah $\Delta = 2^{-(B-1)}$ dinyatakan oleh persamaan (11).

$$\hat{c} = \text{sat}[\text{round}(c \cdot 2^{B-1})] \cdot 2^{-(B-1)} \quad (10)$$

$$\sigma_e^2 = \frac{\Delta^2}{12} = \frac{2^{-2(B-1)}}{12} \quad (11).$$

Model bit-true meniru perilaku kernel CMSIS-DSP. Untuk FIR Q15 dan Q31, hasil kali dijumlahkan dalam akumulator 64-bit tanpa pembulatan antara, kemudian digeser ke kanan sebanyak $B - 1$ bit (pemotongan) dan disaturasi ke B bit. Untuk biquad Q15 dan Q31, koefisien yang nilai mutlaknya melebihi 1 (misalnya $a_1 \approx -1.89$) disimpan dalam bentuk terbagi 2^{*p} dengan $p = \text{postShift}$, lalu hasil akumulator digeser sebanyak $B - 1 - p$ bit sesuai persamaan (12). Keluaran setiap seksi disimpan kembali dalam B bit sebelum menjadi masukan seksi berikutnya. Implementasi float32 menggunakan bentuk langsung II tertransposisi (DF-II T) sesuai fungsi `arm_biquad_cascade_df2T_f32`.

$$c' = c \cdot 2^{-p}, \quad y_q[n] = \text{sat}[\text{acc}[n] \gg (B - 1 - p)] \quad (12)$$

Fungsi perancangan eliptik SciPy menempatkan seluruh penguatan pada seksi pertama. Akibatnya, sinyal antarseksi bisa jauh lebih kecil dibandingkan rentang penuh format sehingga sebagian besar bit terbuang. Untuk menguji pengaruh ini, disusun varian IIR dengan penskalaan norma L_∞ : koefisien pembilang $\mathbf{b}_i$ setiap seksi diskalakan sehingga penguatan puncak kaskade hingga seksi ke- i bernilai satu, sesuai persamaan (13). Penskalaan ini tidak mengubah fungsi alih total, tetapi memanfaatkan rentang dinamik format secara penuh tanpa risiko luapan untuk masukan sinusoidal.

$$g_i = \max_{\omega} |\prod_{l=1}^i H_l(e^{j\omega})|, \quad \mathbf{b}_i \leftarrow \frac{g_{i-1}}{g_i} \mathbf{b}_i, \quad g_0 = 1 \quad (13)$$

2.5 Metrik Evaluasi

Galat akibat aritmetika fixed-point diukur dengan *signal-to-quantization-noise ratio* (SQNR) pada persamaan (14), yaitu perbandingan daya keluaran referensi float64 y_{ref} terhadap daya selisih keluaran terkuantisasi y_q . Kemampuan filter memulihkan sinyal sensor diukur dengan SNR keluaran pada persamaan (15), dengan s_f adalah sinyal bersih yang dilewatkan pada filter referensi yang sama, sehingga tunda filter tidak dihitung sebagai galat.

$$\text{SQNR} = 10 \log_{10} \frac{\sum_n y_{\text{ref}}^2[n]}{\sum_n (y_q[n] - y_{\text{ref}}[n])^2} \text{ dB} \quad (14)$$

$$\text{SNR}_{\text{out}} = 10 \log_{10} \frac{\sum_n s_f^2[n]}{\sum_n (y_q[n] - s_f[n])^2} \text{ dB} \quad (15)$$

Metrik lain adalah riak pita lolos dan atenuasi pita henti dari respons frekuensi dengan koefisien terkuantisasi, atenuasi komponen 50 Hz pada spektrum keluaran, jari-jari kutub maksimum, dan tunda grup. Tunda grup FIR berfase linear bernilai konstan sesuai persamaan (16), sedangkan tunda grup IIR dihitung numerik pada beberapa frekuensi pita lolos.

$$\tau_{\text{FIR}} = \frac{N-1}{2f_s} \quad (16)$$

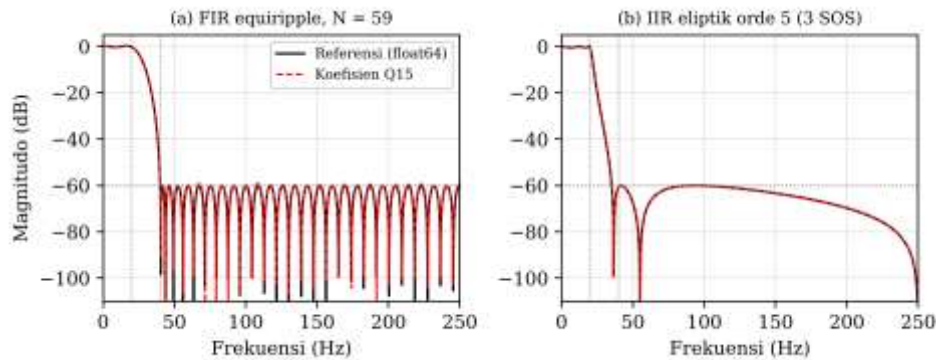
Kebutuhan komputasi dinyatakan dalam jumlah operasi MAC per sampel M dan kebutuhan memori koefisien serta *state*. Jumlah siklus C diestimasi dengan model orde pertama pada persamaan (17). Untuk Q15 dipakai $\kappa = 1$ siklus/MAC karena instruksi SIMD menghitung dua MAC 16-bit sekaligus. Untuk Q31 dan float32 dipakai $\kappa = 2$ siklus/MAC, termasuk pemuatan operan. Overhead C_0 ditetapkan 20 siklus ditambah 6 siklus per seksi IIR. Beban CPU U dihitung pada $f_{\text{clk}} = 168$ MHz. Nilai κ dan C_0 adalah asumsi pemodelan yang perlu divalidasi dengan pencacah siklus DWT pada perangkat keras.

$$C \approx \kappa M + C_0, \quad U = \frac{C f_s}{f_{\text{clk}}} \times 100\% \quad (17)$$

3. HASIL DAN PEMBAHASAN

3.1 Hasil Perancangan Filter

Spesifikasi pada Tabel 1 dipenuhi oleh FIR equiripple dengan panjang $N = 59$ tap dan IIR eliptik orde 5 yang direalisasikan sebagai tiga seksi orde-dua (seksi pertama berorde satu dengan $b_2 = a_2 = 0$). Respons magnitudo kedua filter untuk koefisien float64 dan Q15 ditunjukkan pada Gambar 5. Pada skala grafik, kurva Q15 hampir berimpit dengan referensi; perbedaan terlihat pada dasar-dasar *ripple* pita henti FIR yang tidak lagi turun di bawah -100 dB.



Gambar 5. Respons Magnitudo Filter dengan Koefisien Float64 dan Q15: (a) FIR, (b) IIR

Tabel 2 merangkum pengaruh kuantisasi koefisien terhadap respons frekuensi dan kestabilan. Kuantisasi Q15 menurunkan atenuasi pita henti FIR dari 60,04 dB menjadi 59,09 dB, sehingga spesifikasi 60 dB tidak terpenuhi dengan selisih 0,91 dB. Galat kuantisasi 59 koefisien menambahkan respons galat dengan puncak $-77,3$ dB di pita henti (nilai rms teoretisnya $-83,4$ dB). Komponen kecil ini cukup untuk menaikkan puncak *ripple* equiripple yang semula tepat -60 dB. IIR Q15 dengan penskalaan mempertahankan atenuasi 60,00 dB, sedangkan pergeseran kutub akibat kuantisasi hanya $4,3 \times 10^{-6}$. Format Q31 dan float32 tidak mengubah respons secara berarti.

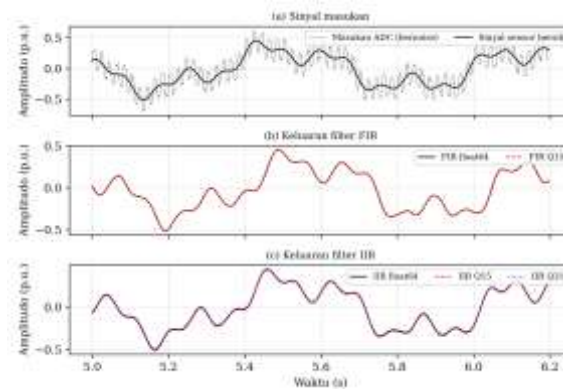
Tabel 2. Pengaruh Kuantisasi Koefisien terhadap Respons Frekuensi

Filter / format	Riak pita lolos (dB)	Atenuasi pita henti (dB)	Jari-jari kutub maks.	Status
FIR float64	0,493	60,04	—	Memenuhi
FIR float32	0,493	60,04	—	Memenuhi
FIR Q31	0,493	60,04	—	Memenuhi
FIR Q15	0,492	59,09	—	Kurang 0,91 dB
IIR SOS float64	0,500	60,00	0,977495	Memenuhi
IIR SOS Q31	0,500	60,00	0,977495	Memenuhi
IIR SOS Q15 (diskalakan)	0,502	60,00	0,977500	Memenuhi
IIR bentuk langsung Q15	—	—	1,000000	Tidak stabil

Baris terakhir Tabel 2 menunjukkan alasan IIR tidak boleh direalisasikan sebagai satu persamaan beda orde tinggi. Koefisien penyebut bentuk langsung orde 5 mencapai 8,65, sehingga harus diskalakan 2^{-4} agar muat dalam Q15 dan resolusi efektifnya turun menjadi 2^{-11} . Dengan resolusi tersebut, salah satu kutub bergeser dari jari-jari 0,9775 ke lingkaran satuan sehingga filter berada pada batas stabil dan dapat beresilasi. Realisasi SOS dengan koefisien Q15 justru hanya menggeser kutub pada digit keenam. Temuan ini sejalan dengan penggunaan kaskade orde-dua sebagai praktik baku untuk mengurangi kepekaan kuantisasi.

3.2 Kinerja Pemfilteran dan Galat Kuantisasi

Gambar 6 memperlihatkan potongan sinyal masukan dan keluaran pada detik ke-5 sampai ke-6,2. Kedua filter menghilangkan interferensi 50 Hz dan derau frekuensi tinggi, dan bentuk gelombang keluaran Q15 berimpit dengan referensi float64. Kurva FIR tertunda konstan 29 sampel, sedangkan kurva IIR memiliki tunda yang berbeda untuk setiap komponen frekuensi.



Gambar 6. Sinyal Masukan dan Keluaran: (a) Masukan, (b) Keluaran FIR, (c) Keluaran IIR

Tabel 3 memuat metrik kuantitatif seluruh konfigurasi. Galat maksimum dinyatakan dalam satuan ternormalisasi (p.u.) terhadap rentang penuh ± 1 .

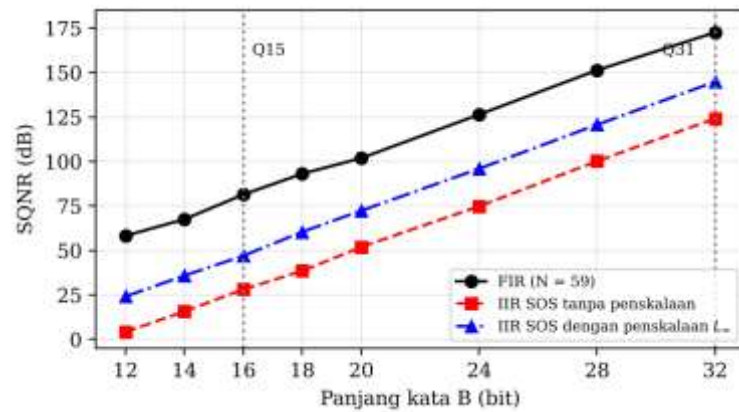
Tabel 3. Kinerja Keluaran Filter pada Setiap Format Bilangan

Filter / format	SQNR (dB)	SNR keluaran (dB)	Galat maks. (p.u.)	Atenuasi 50 Hz (dB)
FIR float64	—	31,65	—	69,0
FIR float32	146,0	31,65	$6,3 \times 10^{-8}$	69,0
FIR Q31	172,6	31,65	$1,4 \times 10^{-9}$	69,0
FIR Q15	81,4	31,65	$5,7 \times 10^{-5}$	70,3
IIR float64	—	32,38	—	67,5
IIR float32	114,9	32,38	$2,3 \times 10^{-6}$	67,5
IIR Q31 tanpa penskalaan	124,2	32,38	$2,7 \times 10^{-7}$	67,5
IIR Q31 diskalakan	144,9	32,38	$2,5 \times 10^{-8}$	67,5
IIR Q15 tanpa penskalaan	28,1	26,68	$1,5 \times 10^{-2}$	67,4
IIR Q15 diskalakan	47,0	32,20	$2,6 \times 10^{-3}$	67,7

Dengan SNR masukan 4,40 dB, kedua filter referensi menaikkan SNR menjadi sekitar 32 dB, atau perbaikan sekitar 27–28 dB. FIR tidak sensitif terhadap format bilangan: SQNR terendah 81,4 dB pada Q15 berarti galat kuantisasi sekitar 50 dB di bawah derau sisa yang memang lolos di pita lolos, sehingga SNR keluaran tidak berubah sampai dua desimal. Keunggulan ini berasal dari dua hal, yaitu FIR tidak memiliki umpan balik sehingga galat tidak terakumulasi, dan kernel CMSIS-DSP menjumlahkan seluruh 59 hasil kali dalam akumulator 64-bit sehingga pembulatan hanya terjadi sekali per sampel keluaran.

IIR Q15 tanpa penskalaan menunjukkan degradasi yang nyata. SQNR hanya 28,1 dB dan SNR keluaran turun 5,7 dB dibandingkan referensi. Penyebab utamanya adalah distribusi penguatan antarseksi: penguatan DC seksi pertama hanya 0,021, sehingga sinyal antara yang disimpan dalam 16 bit menempati sekitar 2 % rentang penuh dan kehilangan sekitar 5,5 bit resolusi. Galat pembulatan yang terjadi di keluaran setiap seksi kemudian diperkuat oleh kutub-kutub seksi berikutnya yang berjari-jari 0,930–0,977. Penguatan derau total dari ketiga titik pembulatan ke keluaran, yang dihitung dari jumlah kuadrat respons impulsnya, mencapai 43,5 dB. Setelah penskalaan L_{∞} sesuai persamaan (13), SQNR naik 18,9 dB menjadi 47,0 dB dan SNR keluaran pulih menjadi 32,20 dB, hanya 0,18 dB di bawah referensi. Atenuasi 50 Hz seluruh konfigurasi tetap di atas 67 dB, sehingga perbedaan format tidak memengaruhi penekanan interferensi, tetapi memengaruhi kebersihan sinyal di pita lolos.

Gambar 7 memperlihatkan SQNR sebagai fungsi panjang kata B . Ketiga kurva naik sekitar 6 dB per bit, sesuai prediksi persamaan (11). Selisih antarkurva menyatakan penguatan derau struktur: IIR tanpa penskalaan berada sekitar 50 dB di bawah FIR, dan penskalaan memperkecil selisih tersebut menjadi sekitar 30 dB. Untuk mencapai SQNR IIR setara FIR Q15 (≈ 81 dB), IIR tanpa penskalaan membutuhkan panjang kata sekitar 25 bit, sedangkan IIR yang diskalakan membutuhkan sekitar 22 bit. Pada mikrokontroler Cortex-M4 kebutuhan ini dipenuhi dengan format Q31 atau float32.



Gambar 7. SQNR terhadap Panjang Kata untuk FIR dan IIR

3.3 Tunda Grup, Kebutuhan Komputasi, dan Memori

FIR 59 tap berfase linear sehingga semua komponen frekuensi tertunda sama, yaitu 29 sampel atau 58 ms sesuai persamaan (16). IIR eliptik memiliki tunda grup 14,5 sampel (29 ms) pada 1,5 Hz, 13,5 sampel (27 ms) pada 5 Hz, 18,8 sampel (37,5 ms) pada 12 Hz, dan 51,2 sampel (102 ms) pada tepi pita lolos 20 Hz. Untuk komponen frekuensi rendah, IIR memberi tunda sekitar separuh FIR, tetapi variasi tunda di dalam pita lolos mendistorsi bentuk gelombang sinyal yang memiliki banyak komponen. Pada aplikasi yang menuntut bentuk gelombang tetap, misalnya deteksi puncak atau pengukuran durasi pulsa, sifat ini menjadi pertimbangan utama.

Tabel 4. Estimasi Kebutuhan Komputasi dan Memori per Kanal ($f_{clk} = 168$ MHz, $f_s = 500$ Hz)

Filter / format	MAC/ sampel	Memori koefisien (B)	Memori state (B)	Siklus/ sampel*	Waktu (μ s)*	Beban CPU (%)*
FIR Q15	59	118	180	79	0,47	0,024
FIR Q31	59	236	360	138	0,82	0,041
FIR float32	59	236	360	138	0,82	0,041
IIR Q15	15	36	24	53	0,32	0,016
IIR Q31	15	60	48	68	0,40	0,020
IIR float32	15	60	24	68	0,40	0,020

Estimasi model orde pertama persamaan (17); perlu divalidasi dengan pengukuran pada perangkat keras

Tabel 4 menunjukkan bahwa IIR membutuhkan 15 MAC per sampel, kira-kira seperempat dari 59 MAC pada FIR. Memori IIR Q15 hanya 60 byte, dibandingkan 298 byte pada FIR Q15 dan 596 byte pada FIR Q31. Memori koefisien biquad Q15 dihitung enam koefisien per seksi karena kernel CMSIS-DSP menyisipkan satu nilai nol untuk keperluan SIMD, sedangkan memori state FIR mencakup $N + 31$ sampel untuk blok 32 sampel. Pada laju sampling 500 Hz, seluruh konfigurasi hanya membebani CPU kurang dari 0,05 %. Artinya, untuk satu kanal sensor berkecepatan rendah, beban komputasi bukan pembeda utama antara FIR dan IIR. Perbedaan ini baru penting ketika jumlah kanal atau frekuensi sampling naik, misalnya akuisisi multikanal pada orde kHz, atau ketika CPU dijalankan pada frekuensi rendah untuk menghemat energi. Dalam kondisi demikian, IIR Q15 yang diskalakan memberi kombinasi biaya terendah dan SNR keluaran yang hampir sama dengan referensi.

3.4 Pedoman Pemilihan dan Perbandingan dengan Penelitian Sebelumnya

Hasil simulasi menghasilkan pedoman praktis berikut. FIR Q15 cocok ketika fase linear dibutuhkan dan memori cukup, dengan catatan bahwa atenuasi pita henti perlu dirancang dengan margin sekitar 1 dB untuk mengompensasi kuantisasi koefisien. IIR Q15 hanya layak bila direalisasikan sebagai kaskade SOS dan penguatan antarseksi diskalakan; tanpa penskalaan, SNR keluaran turun sekitar 5,7 dB. IIR Q31 atau float32 aman tanpa penanganan khusus dengan biaya memori sedikit lebih besar. Realisasi IIR bentuk langsung orde tinggi dalam fixed-point harus dihindari. Perbandingan penelitian ini dengan penelitian sebelumnya disajikan pada Tabel 6.

Tabel 5. Perbandingan dengan Penelitian Sebelumnya

Penelitian	Platform	Filter / format	Fokus
Sim dkk. [8]	STM32 Nucleo-64 (perangkat keras)	FIR dan IIR, kode manual fixed-point dan kode otomatis	Orde filter dan frekuensi sampling maksimum tiap metode implementasi

Penelitian	Platform	Filter / format	Fokus
Żyliński dkk. [1]	STM32WB55 Cortex-M4	CMSIS-DSP, klasifikasi SVM dan Naïve Bayes	Deteksi fibrilasi atrium dan waktu eksekusi di perangkat
Pilipović dkk. [13]	ASIC (sintesis 45 nm)	IIR pembobot-A orde 6 dengan pengali aproksimasi	Energi dan deviasi respons akibat kuantisasi dan urutan seksi
Septian dkk. [7]	Arduino Uno + MATLAB	Filter analog dan digital di komputer	Peningkatan SNR sinyal PPG
Penelitian ini	STM32F407 Cortex-M4 (simulasi bit-true)	FIR equiripple dan IIR eliptik SOS; Q15, Q31, float32	SQNR, SNR keluaran, kestabilan, penskalaan antarseksi, MAC, memori

Sim dkk. Mengukur kinerja FIR dan IIR langsung pada perangkat keras STM32, tetapi tidak memisahkan kontribusi kuantisasi koefisien, pembulatan, dan penskalaan. Penelitian ini melengkapinya dengan memisahkan ketiga sumber galat tersebut melalui simulasi bit-true sehingga penyebab degradasi dapat diidentifikasi sebelum pengujian perangkat keras. Pilipović dkk. [13] menunjukkan pentingnya urutan seksi pada ASIC. Pada filter yang diuji di sini, kuantisasi koefisien Q15 hanya menggeser kutub sebesar $4,3 \times 10^{-6}$ dan atenuasi pita henti sebesar 0,06 dB, sedangkan penskalaan antarseksi mengubah SQNR sebesar 18,9 dB. Dengan kata lain, pada format Q15 distribusi penguatan lebih menentukan mutu keluaran daripada presisi koefisien.

Penelitian ini memiliki beberapa keterbatasan. Sinyal uji bersifat sintesis dan stasioner, sehingga perilaku terhadap sinyal sensor nyata dengan lonjakan atau pergeseran garis dasar belum diuji. Estimasi siklus menggunakan model orde pertama, belum memperhitungkan *wait state* memori Flash, *cache* ART, dan *pipeline* prosesor. Validasi lanjutan dapat dilakukan dengan pengujian *processor-in-the-loop* pada papan STM32F4 menggunakan pencacah siklus DWT.

4. KESIMPULAN

Penelitian ini membandingkan filter lolos-rendah FIR equiripple 59 tap dan IIR eliptik orde 5 untuk mikrokontroler STM32 Cortex-M4 melalui simulasi bit-true kernel CMSIS-DSP dalam format Q15, Q31, dan float32. FIR tidak sensitif terhadap format bilangan: pada Q15, SQNR mencapai 81,4 dB dan SNR keluaran 31,65 dB, sama dengan referensi float64, meskipun atenuasi pita henti turun 0,91 dB di bawah spesifikasi akibat kuantisasi koefisien. IIR membutuhkan seperempat jumlah MAC dan seperlima memori FIR, tetapi pada Q15 hanya layak bila direalisasikan sebagai kaskade seksi orde-dua dengan penskalaan antarseksi. Penskalaan L_∞ menaikkan SQNR IIR Q15 dari 28,1 dB menjadi 47,0 dB dan memulihkan SNR keluaran dari 26,68 dB menjadi 32,20 dB, sedangkan realisasi bentuk langsung orde tinggi menempatkan kutub pada lingkaran satuan. Pada laju sampling 500 Hz beban CPU seluruh konfigurasi di bawah 0,05 %, sehingga pemilihan filter untuk satu kanal lebih ditentukan oleh kebutuhan fase linear dan akurasi dibandingkan beban komputasi. Penelitian selanjutnya akan memvalidasi hasil ini pada perangkat keras STM32F4 melalui pengujian *processor-in-the-loop* dengan sinyal sensor nyata.

REFERENCES

- [1] M. H. H. Ichsan, "Analisis Kinerja Jaringan Sensor Nirkabel untuk Edge Computing Menggunakan LORA SX1278," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 8, no. 5, p. 887, 2021, doi: 10.25126/jtiik.2021853631.
- [2] E. M. Mase, F. W. Witi, and B. Y. Bhae, "Optimalisasi Pemantauan Level Air Dalam Bak Penampungan Air Menggunakan Internet Of Things (IOT) di Universitas Flores," *J. Ilm. Sist. Inf.*, vol. 4, no. 1, pp. 163–179, 2025, doi: 10.51903/4xg62v92.
- [3] M. Ilham Sabilal Muna, G. Wahyu Nyipto Wibowo, and Nadia Annisa Maori, "Implementasi Pemilah Sampah Otomatis Menggunakan ESP32-CAM dengan Metode *Gaussian blur*," *J. Pengemb. Teknol. Inf. dan Komun.*, vol. 4, no. 1, pp. 1–8, 2026, doi: 10.52060/juipik.v4i1.4186.
- [4] S. R. DAMARDHI, I. S. FARADISA, and S. SOTYOHADI, "Evaluating a Preprocessing Pipeline for Fetal Phonocardiography Using FIR Filtering," *ELKOMIKA J. Tek. Energi Elektr. Tek. Telekomun. Tek. Elektron.*, vol. 13, no. 2, p. 155, Apr. 2025, doi: 10.26760/elkomika.v13i2.155.
- [5] L. Rahmawati, W. T. Atmojo, E. P. Cynthia, M. M. Cynthia, and D. N. Cynthia, "Pembandingan Arsitektur Transformer dan CNN untuk Pengolahan Data Non-Visual," *J. Ilmu Komput. dan Tek. Inform.*, vol. 2, no. 1, pp. 8–14, 2026, doi: 10.64803/juikti.v2i1.80.
- [6] S. Furqon, I. Santoso, and Y. A. Adi Soetrisno, "Perancangan Sistem Pengontrolan Lampu Lalu Lintas Berdasarkan Tingkat Kepadatan Kendaraan Menggunakan Metode Fuzzy," *Transient J. Ilm. Tek. Elektro*, vol. 9, no. 1, pp. 88–96, 2020, doi: 10.14710/transient.v9i1.88-96.
- [7] D. Zulfikar and H. Hermanto, "Hamming Code in JPEG Image Steganography within the Discrete Cosine

- Transform Domain,” *J. Appl. Informatics Comput.*, vol. 9, no. 3, pp. 868–875, 2025, doi: 10.30871/jaic.v9i3.9387.
- [8] C. Caroline, N. H. Shabrina, M. R. Ao, N. Laurency, and V. Lee, “Analisis Aplikasi Filter FIR dan Filter IIR dalam Pra-pemrosesan Sinyal Elektroensefalografi,” *Ultim. Comput. J. Sist. Komput.*, vol. 12, no. 1, pp. 40–48, 2020, doi: 10.31937/sk.v12i1.1621.
- [9] M. R. A. SEPTIAN, S. HARIMURTI, W. PRIHARTI, I. HIDAYAT, and M. RAMDHANI, “Optimization System on Photoplethysmography (PPG) Signal using Analog and Digital Filters,” *ELKOMIKA J. Tek. Energi Elektr. Tek. Telekomun. Tek. Elektron.*, vol. 14, no. 1, p. 121, 2026, doi: 10.26760/elkomika.v14i1.121.
- [10] R. Pilipović, V. Risojević, and P. Bulić, “On the design of an energy efficient digital IIR A-weighting filter using approximate multiplication,” *Sensors (Switzerland)*, vol. 21, no. 3, pp. 1–22, 2021, doi: 10.3390/s21030732.
- [11] S. Santoso *et al.*, “Penerapan Filter Digital untuk Menghilangkan Gangguan pada Sinyal Elektrokardiogram,” *JREEC J. Renew. Energy, Electron. Control*, vol. 4, no. 2, pp. 36–42, 2024, doi: 10.31284/j.jreec.2024.v4i2.6711.
- [12] B. Syihabuddin, D. A. Nurmantris, and A. D. Prasetyo, “Perancangan Bandpass Filter Pita Sempit pada Frekuensi L-Band untuk Aplikasi Synthetic Aperture Radar (SAR),” *J. Infotel*, vol. 9, no. 2, p. 198, 2017, doi: 10.20895/infotel.v9i2.230.
- [13] O. P. Ayeoribe, “Instruction Set Optimization for FM-Type Digital Signal Processor (DSP) Architectures with Integration of DVB-T2 TV Systems,” *Appl. Inf. Technol. Comput. Sci.*, vol. 5, no. 1, pp. 21–28, 2026, doi: 10.58466/35jafg24.
- [14] K. Riyanto and I. G. Ngurah Suryantara, “Implementasi Model Lstm Pada Mikrokontroler Untuk Prediksi Iklim Mikro Jakarta Utara Secara Real-Time Menggunakan Tinyml,” *JATI (Jurnal Mhs. Tek. Inform.)*, vol. 10, no. 1, pp. 970–977, 2026, doi: 10.36040/jati.v10i1.16914.
- [15] A. Purnomo and R. Hartono, “Analisa Waktu Respon pada Metode Pengukuran Jarak pada Sistem Informasi Geografi,” *J. Teknol. Inf. dan Ilmu Komput.*, vol. 12, no. 1, pp. 39–46, 2025, doi: 10.25126/jtiik.2025128690.