

## Unjuk Kerja GPIO, PWM, ADC dan Timer pada Mikrokontroler STM32F103, ESP32S dan ATmega328

Fatkhur Rohman<sup>1</sup>, Nurhadi<sup>2</sup>, Mira Esculenta Martawati, ST., MT<sup>3</sup>

e-mail: [fatkhur\\_rohman@polinema.ac.id](mailto:fatkhur_rohman@polinema.ac.id), [nurhadi@polinema.ac.id](mailto:nurhadi@polinema.ac.id), [mmilla20@gmail.com](mailto:mmilla20@gmail.com)

<sup>1, 2, 3</sup>Jurusan Teknik Mesin, Politeknik Negeri Malang, Indonesia

### Informasi Artikel

#### Riwayat Artikel

Diterima 5 Mei 2021

Direvisi 03 Oktober 2021

Diterbitkan 26 Oktober 2021

#### Kata kunci:

STM32

ESP32

FreeRTOS

Arduino

#### Keywords:

STM32

ESP32

FreeRTOS

Arduino

### ABSTRAK

Perangkat embedded system pada masa sekarang memiliki banyak pilihan terhadap jenis mikrokontroler yang sesuai dengan kebutuhan. Hal ini menjadi tantangan tersendiri bagi pengguna ketika diharuskan untuk memilih salah satu jenis mikrokontroler tersebut. Sebagai contoh permasalahan apakah mikrokontroler yang telah dipilih tersebut memiliki sejumlah pin GPIO yang diinginkan, dengan frekuensi switching yang tinggi, berapakah jumlah kanal, resolusi, linieritas dan durasi konversi ADC, bagaimana kemampuan peripheral internal DAC, Timer dan PWM yang bisa dibangkitkan dari mikrokontroler tersebut.

Penelitian ini telah membandingkan setidaknya 4 peripheral internal utama yang dimiliki oleh 3 jenis mikrokontroler. Metode yang dilakukan adalah dengan menguji karakteristik GPIO, PWM, TIMER dan ADC pada 3 jenis mikrokontroler yaitu Arduino ATmega328, STM32F103C8 dan ESP32. Eksperiment dilakukan dengan mengevaluasi frekuensi switching digital output, mengevaluasi resolusi sinyal hasil konversi ADC, mengevaluasi ketepatan hasil instruksi delay berkaitan dengan timer program dan waktu konversi sinyal DAC semuanya dilakukan pada masing-masing mikrokontroler.

Hasil akhir dari penelitian ini menunjukkan, mikrokontroler ESP32 memiliki unjuk kerja GPIO, PWM, TIMER dan ADC terbaik apabila dibandingkan dengan jenis lainnya. Penelitian ini juga membuktikan integrasi FreeRTOS pada Framework Arduino bisa berfungsi dengan optimal meskipun mikrokontroler berjalan pada 2 task yang berbeda di 2 core CPU yang bekerja secara paralel. Frekuensi switching digital output pada ESP32 mampu mencapai 3MHz, waktu konversi ADC hanya 5,7us dan DAC hanya 3,7us.

### ABSTRACT

Today's embedded systems have many choices for the type of microcontroller that suits the needs. This is a challenge in itself for users when required to choose one type of microcontroller. For example, the problem of whether the selected microcontroller has the desired number of GPIO pins, with a high switching frequency, what is the number of channels, resolution, linearity, and duration of the ADC conversion, what is the ability of the internal DAC, Timer and PWM peripherals that can be generated from the microcontroller.

This study has compared at least 4 main internal peripherals owned by 3 types of microcontrollers. The method used is to test the characteristics of the GPIO, PWM, TIMER, and ADC on 3 types of microcontrollers, namely Arduino ATmega328, STM32F103C8, and ESP32. The experiment was carried out by evaluating the digital output switching frequency, evaluating the signal resolution of the

ADC conversion result, evaluating the accuracy of the delay instruction results related to the program timer and DAC signal conversion time, all of which were carried out on each microcontroller. The final results of this study indicate that the ESP32 microcontroller has the best GPIO, PWM, TIMER, and ADC performance when compared to other types. This research also proves that the FreeRTOS integration on the Arduino Framework can function optimally even though the microcontroller runs on 2 different tasks on 2 CPU cores that work in parallel. The digital output switching frequency on the ESP32 is capable of reaching 3MHz, the ADC conversion time is only 5.7us and the DAC is the only 3.7us.

***Penulis Korespondensi:***

Fatkhur Rohman,  
Jurusan Teknik Mesin,  
Politeknik Negeri Malang,  
Jl. Soekarno Hatta No. 9, Malang, Jawa Timur, Indonesia.  
Email: fatkhur\_rohman@polinema.ac.id

## **1. PENDAHULUAN**

Perkembangan teknologi mikrokontroler sudah lama mengadopsi arsitektur prosesor 32 bit. Adopsi internal power management yang efisien membuat arsitektur prosesor tersebut membutuhkan daya yang sangat rendah apabila dibandingkan dengan prosesor 8-bit yang sampai sekarang masih lebih banyak digunakan. Selain itu, keunggulan fitur internal peripheral yang dimiliki pun jauh melebihi prosesor 8bit lainnya. Sebagai contoh adalah mikrokontroler STM32F103 dalam modul bluepill yang memiliki kemampuan diatas rata-rata namun dengan harga paling murah.

Permasalahan yang muncul adalah kebanyakan pengembang aplikasi memerlukan waktu dan pemikiran tidak sedikit untuk mengadopsi jenis mikrokontroler baru pada aplikasi yang digunakan. Meskipun vendor STM32 telah menyediakan pustaka program secara daring, sayangnya informasi itu masih terbatas pada dokumentasi pustaka program tanpa ada contoh program yang dibutuhkan dan siap digunakan. Hal ini sangat berbeda dengan adopsi mikrokontroler AVR yang telah lebih dahulu populer, sehingga banyak sekali ditemukan contoh dan panduan aplikasi program yang mempermudah pengembang aplikasi.

Sebagai solusi dari permasalahan tersebut, penelitian ini berusaha untuk menganalisis secara lebih detail kemampuan dari peripheral yang dimiliki oleh mikrokontroler STM32F103. Hal ini perlu dilakukan untuk memberikan deskripsi detail berbanding jenis mikrokontroler STM32F103 dibandingkan mikrokontroler AVR atau yang lain. Tentu saja sesuai dengan datasheet ini telah dideskripsikan dan dijelaskan, namun dalam tahapan praktis pada aplikasi terapan masih belum pernah dilakukan. Dimulai dari peripheral GPIO untuk mengendalikan aplikasi ON/OFF digital, pembangkitan sinyal PWM untuk aplikasi kontrol motor, penggunaan peripheral ADC untuk membaca informasi sensor dan timer untuk aplikasi pewaktuan dan tanggal.

Penelitian sebelumnya tentang penggunaan mikrokontroler STM32 telah dilakukan pada aplikasi akuisisi data secara real time untuk sinyal yang dibangkitkan oleh gerakan mekanis pada mesin industri. Penelitian tersebut mengembangkan sebuah sistem operasi kecil yang ditanam pada otak mikrokontroler STM32F103VET6 untuk membaca sinyal sensor melalui peripheral ADC. Keunggulan mikrokontroler STM32 tersebut sangat berguna karena firmware yang dikembangkan mampu bekerja dengan kecepatan tinggi, namun membutuhkan daya yang sangat rendah [1].

Penelitian lainnya juga telah menggunakan STM32 sebagai kontroler utama untuk mengendalikan sebuah robot kursi lipat yang mampu membantu pasien dari posisi tidur ke posisi duduk. Dalam blok diagram yang disajikan dalam penelitian tersebut, mikrokontroler STM32 digunakan untuk mengendalikan driver motor untuk mekanisme lipat dan berjalan (diatas roda) menggunakan komunikasi CAN bus. GPIO pada STM32 digunakan juga untuk kontrol ON/OFF relay untuk mekanisme kursi terangkat dan mode toilet pasien [2].

Penelitian ini telah membandingkan setidaknya 4 peripheral internal utama yang dimiliki oleh 3 jenis mikrokontroler. Metode yang dilakukan adalah dengan menguji karakteristik GPIO, PWM, TIMER dan ADC pada 3 jenis mikrokontroler yaitu Arduino ATmega328, STM32F103C8 dan ESP32. Eksperiment dilakukan dengan mengevaluasi frekuensi switching digital output, mengevaluasi resolusi sinyal hasil konversi ADC, mengevaluasi ketepatan hasil instruksi delay berkaitan dengan timer program dan waktu konversi sinyal DAC semuanya dilakukan pada masing-masing mikrokontroler.

**2. METODE PENELITIAN**

Penelitian eksperimen dilakukan dengan cara mengontrol, memanipulasi dan melakukan observasi terhadap variabel penelitian., dimana bahan dan alat yang dibutuhkan untuk penelitian ini adalah sebagai berikut:

1. Komputer
2. Modul STM32F103 Bluepill, ESP32 dan ATmega328
3. Portable Digital Scope
4. Sensor Analog – Potensiometer high Granulity

Untuk menentukan platform pemrograman yang lebih efisien, penelitian ini mencoba untuk menggunakan beberapa editor program beserta compiler atau interpreter yang tersedia secara luas dan telah banyak digunakan oleh banyak pengembang. Untuk 3 jenis mikrokontroler yang digunakan sesuai rencana tentu saja menggunakan Arduino IDE untuk percobaan yang pertama. Platform pemrograman berikutnya adalah menggunakan STM32CubeIDE yang telah disediakan oleh ST Electronic untuk pembuatan program STM32F103C8. Untuk pilihan platform program yang ketiga adalah menggunakan libopenm3 yang sepenuhnya dibangun dari komunitas opensource. Terakhir, penelitian ini mencoba menggunakan Visual Studio Code dengan penambahan ekstensi PlatformIO.

Dari beberapa platform pemrograman tersebut, penelitian menitikberatkan pada beberapa unsur utama berikut untuk menentukan keunggulan masing-masing platform pemrograman tersebut:

1. Dukungan komunitas
2. Code completion
3. Ketersedian library program
4. Dukungan terhadap variasi jenis mikrokontroler
5. Kapasitas resources yang dibutuhkan

**3. HASIL DAN PEMBAHASAN**

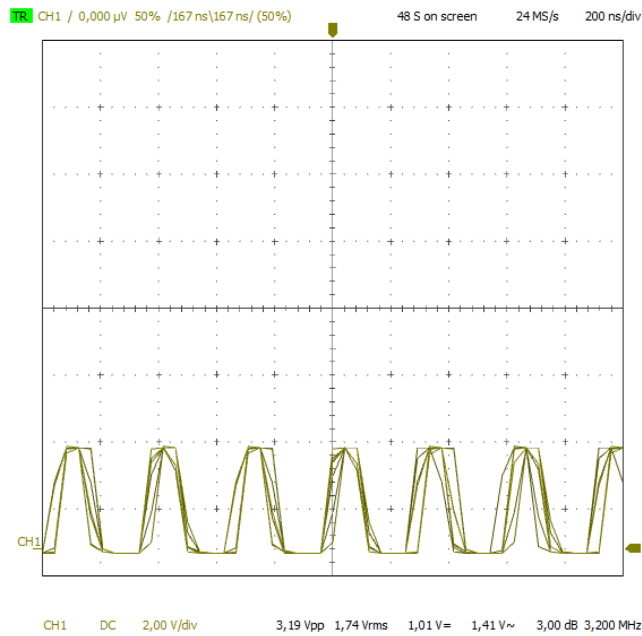
Tabel 1 berikut menunjukkan hasil percobaan yang telah dilakukan untuk masing-masing jenis mikrokontroler untuk nilai F CPU, V peak-peak dan F GPIO.

Tabel 1. Perbandingan frekuensi CPU, tegangan peak dan frekuensi GPIO 3 jenis mikrokontroler

Mikrokontroler	FCPU_Max (MHz)	Vpp (Volt)	F GPIO (kHz)
STM32F103C8 (Bluepill)	72	4,6	410,4
ESP32S(ESP-WROOM-32)	240	3,2	3200
ATmega328p (Arduino Uno)	16	3,35	133

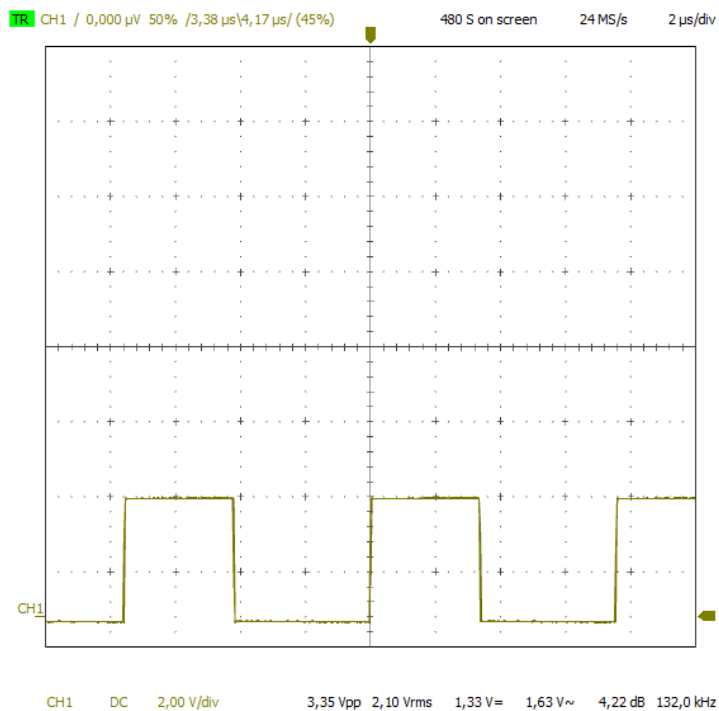
**3.1. Pengujian frekuensi switching output digital sinyal**

Untuk mengetahui unjuk kerja peripheral GPIO, berikut ini adalah menunjukkan tampilan salah satu pin output digital pada masing-masing mikrokontroler.



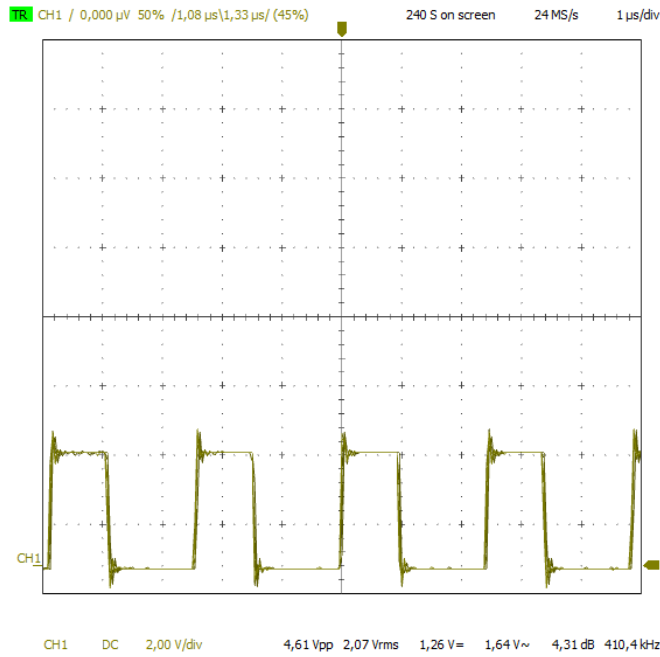
Gambar 1. Frekuensi switching GPIO pada ESP32S

Digambar diatas menunjukkan bahwa frekuensi digital output pada ESP32S mencapai 3 MHz. Sebagaimana ditunjukkan pada program bahwasanya tidak ada delay antara logika low dan high. Sehingga, perubahan logika langsung dilakukan oleh pin tanpa ada jeda instruksi lain. Gambar osiloscope juga menunjukkan bahwa sinyal tidak lagi sempurna berbentuk kotak karena tingginya frekuensi kerja yang dihasilkan. Selain itu tingginya frekuensi kerja dari output digital ini dikarenakan CPU dari mikrokontroler beroperasi pada kecepatan tinggi tanpa adanya beban tambahan apapun. Frekuensi kerja CPU ESP32 adalah sebesar 240MHz tertinggi dibandingkan jenis mikrokontroler lainnya. Selain itu, juga memiliki 2 inti CPU yang mengijinkan kerja paralel antara 2 CPU secara bersamaan. Hal ini juga yang mendasari ESP32S telah mendukung FreeRTOS dalam framework Arduino.



Gambar 2. Frekuensi switching GPIO pada ATmega328

Untuk frekuensi output digital pada ATmega328 cukup rendah sebesar 132kHz saja. Nilai ini tergolong rendah apabila dibandingkan dengan CPU competitor yang mampu mencapai nilai lebih dari 10x lipat. ATmega328 sampai sekarang masih sangat populer karena memang telah lama digunakan oleh komunitas maupun platform Arduino. Banyaknya library program yang telah disediakan menjadi pertimbangan jenis mikrokontroler ini. Sebagai pertimbangan tentu saja, rendahnya frekuensi switching ini dipengaruhi oleh frekuensi CPU ATmega328 yang hanya 16MHz. Disisi yang lain, apabila ATmega tidak memiliki varian lain yang mendukung aplikasi terkini seperti IoT, AI dan lainnya tentu saja akan ditinggalkan oleh penggunaannya.

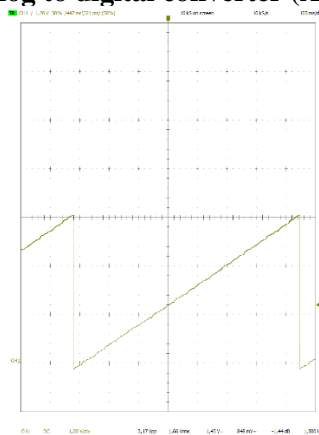


Gambar 3. Frekuensi switching GPIO pada STM32f103c8

Jenis mikrokontroler ini tergolong lebih baru apabila dibandingkan dengan ATmega328 namun lebih lama apabila dibandingkan dengan ESP32. Pengujian frekuensi GPIO sangat menjanjikan dengan menampilkan nilai mencapai hampir 0,5 MHz. Tentu saja dengan CPU 1 inti hal ini sangat menjanjikan karena disamping bisa aktif dalam mode low power, namun masih bisa bekerja diatas nilai rata-rata mikrokontroler jenis ATmega. Frekuensi kerja dari CPU STM3eF103C8 ini bisa mencapai nilai 72MHz.

Salah satu fitur yang sangat menjanjikan adalah kemampuan untuk menjalankan aplikasi RTOS yang didukung penuh oleh ST electronic sebagai vendor IC tersebut. Penelitian ini telah membuktikan bahwa RTOS dapat berjalan dengan lancar pada platform ESP32 dan STM32F103C8, namun tidak pada ATmega328.

**3.2. Pengujian data konversi Analog to digital converter (ADC) dan PWM**



Gambar 4. Sinyal input analog untuk pengujian peripheral ADC

Sinyal diatas merupakan sinyal sumber input ADC yang digunakan untuk acuan konversi sinyal ADC. Sinyal tersebut dibangkitkan menggunakan IC MCP7435 I2C module. Modul DAC ini diaktifkan dari ESP32 menggunakan RTOS dengan task-1 aktif pada core CPU ke 2. Selanjutnya, task ke-2 pada core CPU 1 dijalankan untuk membaca sinyal ADC secara simultan. Durasi pembacaan ADC pertama dan kedua kemudian dihitung durasi nya menggunakan timer internal ESP32 dan hasilnya di tampilkan dalam komputer. Selanjutnya data di tulis pada Tabel berikut:

Tabel 2. Waktu konversi ADC, durasi perintah analogWrite() untuk PWM, dan durasi perintah digitalRead() untuk input digital

Mikrokontroler	Waktu Konversi ADC	PWM(analogWrite)	Input Digital(digitalRead)
STM32F103C8 (Bluepills)	6,7us	4,4us	0,6us
ESP32S(ESP-WROOM-32)	5,7us	3,7us	0,142us
ATMega328p (Arduino Uno)	112us	11,5us	10us

Dari Tabel 2 terlihat bahwa pada semua variabel uji tersebut menghasilkan nilai waktu paling cepat pada ESP32S. Dengan nilai paling lambat didapatkan pada ATMega328. Dukungan dual core CPU dengan kecepatan mencapai 240 MHz benar-benar memberikan keunggulan pada semua aspek pada ESP32S. Arsitektur 32bit pada penelitian ini diperkirakan juga kurang tepat apabila dibandingkan dengan ATMega328 yang masih menggunakan sistem 8 bit. Namun, tentu saja dari sisi harga yang hampir sama, bisa terlihat bahwa ESP32S benar-benar mampu bekerja dengan kecepatan yang sangat tinggi.

Selain itu, dengan integrasi 16 channel ADC pada ESP32 dan STM32F103 memberikan mikrokontroler ini keleluasaan aplikasi apabila diinginkan untuk membangun aplikasi dengan banyak input sensor 12bit resolusi ADC menghasilkan konversi yang lebih akurat dan dapat diterima oleh aplikasi yang lebih kompleks, dimana akurasi dan kepresisian menjadi point penting pada perangkat yang digunakan.

### 3.3. Pengujian Timer Delay

Tabel 3. Data Timer Delay

Mikrokontroler	delay_ms(1)	delay_ms(100)	delay_us(2)	delay_us(5)	delay_us(100)
STM32F103C8 (Bluepills)	1002.9 us	100299.992 us	2.122 us	5.148 us	99.998 us
ESP32S(ESP-WROOM-32)	1000.01 us	100000.001 us	2.237 us	4.999 us	99.999 us
ATMega328p (Arduino Uno)	1007.987 us	100054.546 us	0.707 us	3.782 us	99.378 us

Pengujian timer delay ini sebenarnya bertujuan untuk mengetahui seberapa akurasi nilai delay yang dihasilkan dibandingkan dengan nilai asli yang diinginkan oleh programmer. Berdasar hasil percobaan diatas, masih terlihat dominasi ESP32S yang mampu dengan sangat akurat melaksanakan semua perintah delay dengan durasi waktu yang sangat tepat apabila dibandingkan dengan jenis mikrokontroler yang lain.

## 4. KESIMPULAN

Dari penelitian telah dilakukan dan menganalisis data hasil percobaan, didapatkan beberapa simpulan yaitu kinerja terbaik dari sisi perangkat GPIO, perangkat ADC, Timer dan PWM dimiliki oleh ESP32S yang memiliki nilai tertinggi apabila dibandingkan dengan STM32F103C8 dan ATMega328. Kesimpulan berikutnya adalah platform pemrograman terbaik untuk aplikasi mikrokontroler dimiliki oleh Visual Studio Code dengan penambahan ekstensi PlatformIO yang memberikan hasil maksimal pada semua aspek yang dibutuhkan oleh seorang programmer

## 5. UCAPAN TERIMA KASIH

Terima kasih penulis sampaikan kepada UPT. P2M Politeknik Negeri Malang yang telah memberikan dana penelitian dengan Surat Perjanjian Penugasan Pelaksanaan Program Penelitian Nomor 042.01.2.401004/2020. Selanjutnya, terima kasih kepada Direktur dan Kajur/Kaprodi Teknik Mesin Politeknik Negeri Malang yang telah memfasilitasi kegiatan penelitian ini.

## 6. DAFTAR PUSTAKA

- [1] H. F. Zhang and W. Kang, "Design of the data acquisition system based on STM32," *Procedia Comput. Sci.*, vol. 17, pp. 222–228, 2013, doi: 10.1016/j.procs.2013.05.030.
- [2] L. Sang, M. Yamamura, F. Dong, and Z. Gan, "applied sciences of a Novel Wheelchair-Stretcher Assistive Robot."
- [3] V. K. Abdrakhmanov, R. B. Salikhov, and K. V. Vazhdacv, "Development of a Sound Recognition System Using STM32 Microcontrollers for Monitoring the State of Biological Objects," *2018 14th Int. Sci. Conf. Actual Probl. Electron. Instrum. Eng. APEIE 2018 - Proc.*, pp. 170–173, 2018, doi: 10.1109/APEIE.2018.8545278.
- [4] X. Zhang, P. Zhang, Y. Shi, J. Dang, and L. Yuan, "Hardware Design of Non-contact Voltage Detector Based on STM32 Hardware Design of Non-contact Voltage Detector Based on STM32 Microcontroller," pp. 0–6, 2020, doi: 10.1088/1757-899X/768/6/062036.
- [5] R. Setiawan, M. Rivai, and S. Suwito, "Implementasi Analog Front End pada Sensor Kapasitif Untuk Pengaturan Kelembaban Menggunakan Mikrokontroler STM32," *J. Tek. ITS*, vol. 6, no. 1, 2017, doi: 10.12962/j23373539.v6i1.22153.