IMPLEMENTASI *REAL-TIME OPERATING SYSTEM* (RTOS) PADA SISTEM DETEKSI DINI PENCURIAN BERBASIS ESP32-CAM

SKRIPSI

M. FATHURRAHMAN
H1051211108

PROGRAM STUDI REKAYASA SISTEM KOMPUTER
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS TANJUNGPURA
PONTIANAK
2026

IMPLEMENTASI *REAL-TIME OPERATING SYSTEM* (RTOS) PADA SISTEM DETEKSI DINI PENCURIAN BERBASIS ESP32-CAM

M. FATHURRAHMAN
H1051211108

Skripsi
Sebagai salah satu syarat untuk memperoleh gelar
Sarjana Komputer pada Program Studi Rekayasa Sistem Komputer

PROGRAM STUDI REKAYASA SISTEM KOMPUTER
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS TANJUNGPURA
PONTIANAK
2026

LEMBAR PENGESAHAN

Judul Tugas Akhir	: Implementasi *Real-Time Operating System* (RTOS) Pada Sistem Deteksi Dini Pencurian Berbasis ESP32-CAM
Nama Mahasiswa	: M. Fathurrahman
NIM	: H1051211108
Jurusan/Program Studi	: Rekayasa Sistem Komputer
Tanggal Lulus	: 30 Juli 2026
SK Pembimbing	: 4251/DST/UN22.8/DT.00.09/2026/15 Juli 2026
SK Penguji	: 5077/DST/UN22.8/DT.00.09/2026/31 Juli 2026

Dosen Pembimbing

Pembimbing I	Pembimbing II
Rahmi Hidayati, S.Kom., M.Cs.	Tedy Rismawan, S.Kom., M.Cs.
NIP 198607202015042001	NIP 198609222014041002

Dosen Penguji

Ketua Penguji	Anggota Penguji
Dr. Dedi Triyanto, S.T., M.T.	Irma Nirmala S.T., M.T.
NIP 198112242009121003	NIP 198404052019032015

Pimpinan Sidang (merangkap anggota penguji)	Sekretaris Sidang (merangkap anggota penguji)
Rahmi Hidayati, S.Kom., M.Cs.	Tedy Rismawan, S.Kom., M.Cs.
NIP 198607202015042001	NIP 198609222014041002

Mengesahkan
Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam
Universitas Tanjungpura

Prof. Dr. Gusrizal, S.Si., M.Si.
NIP 197108022000031001

PERNYATAAN INTEGRITAS AKADEMIK

Saya yang bertanda tangan di bawah ini:

Nama	: M.Fathurrahman
NIM	: H1051211108
Program Studi/Jurusan	: Rekayasa Sistem Komputer
Fakultas	: Matematika dan Ilmu Pengetahuan Alam

dengan ini menyatakan bahwa dokumen ilmiah Tugas Akhir yang disajikan ini tidak mengandung unsur pelanggaran integritas akademik sesuai Peraturan Menteri Pendidikan, Kebudayaan, Riset, Dan Teknologi Republik Indonesia Nomor 39 Tahun 2021. Apabila di kemudian hari dokumen ilmiah Tugas Akhir ini mengandung unsur pelanggaran integritas akademik sesuai ketentuan perundangan tersebut, maka saya bersedia menerima sanksi akademik dan/atau sanksi hukum yang berlaku.

Demikian pernyatan ini untuk digunakan sebagaimana mestinya.

Pontianak, 24 Agustus 2026



M.Fathurrahman
H1051211108

IMPLEMENTASI REAL-*TIME* OPERATING SYSTEM (RTOS) PADA SISTEM DETEKSI DINI PENCURIAN BERBASIS ESP32-CAM

Abstrak

Real-Time Operating System (RTOS) diperlukan untuk menjamin kesinambungan deteksi pada sistem keamanan lemari rumah guna mengatasi masalah *blocking delay* pada arsitektur *superloop* saat pengiriman gambar ke Telegram. Penelitian ini bertujuan menguji keandalan arsitektur FreeRTOS dibandingkan *superloop* pada ESP32-CAM. Sistem mengintegrasikan kernel FreeRTOS dengan menggabungkan sensor PIR sebagai pemicu siaga dan sensor getar SW-420 sebagai konfirmasi akhir, serta Telegram Bot API untuk transmisi peringatan. Pengujian dilakukan melalui 55 stimulus fisis dengan parameter uji *capture rate*, *blocking delay*, dan *response time*. Hasil eksperimen membuktikan bahwa arsitektur FreeRTOS mampu mempertahankan *capture rate* 100% karena *blocking delay* berhasil diisolasi pada tugas jaringan, meskipun membutuhkan *response time* sedikit lebih besar. Sebaliknya, arsitektur *superloop* mengalami kegagalan dengan kehilangan hingga 58,18% data kejadian akibat CPU yang terkunci selama transmisi. Dengan demikian, penerapan FreeRTOS terbukti lebih sesuai untuk memperkuat sistem deteksi dini pencurian pada objek penyimpanan tertutup.

Kata kunci: Deteksi Dini, Pencurian*, ESP32-CAM, Real Time Operating System (RTOS), Telegram*.

IMPLEMENTATION OF REAL-TIME OPERATING SYSTEM (RTOS) ON AN ESP32-CAM BASED EARLY THEFT DETECTION SYSTEM

Abstract

A Real-Time Operating System (RTOS) is essential to ensure detection continuity in a home cabinet security system, specifically to overcome the blocking delay issue inherent in the superloop architecture during image transmission to Telegram. This study aims to evaluate the reliability of the FreeRTOS architecture compared to the superloop method on the ESP32-CAM. The system integrates the FreeRTOS kernel with a sensor fusion consisting of a PIR sensor as a standby trigger and an SW-420 vibration sensor for final confirmation, along with the Telegram Bot API for alert transmission. Testing was conducted using 55 physical stimuli, utilizing capture rate, blocking delay, and response time as evaluation parameters. Experimental results demonstrate that the FreeRTOS architecture successfully maintained a 100% capture rate by isolating the blocking delay within the network task, albeit requiring a slightly higher response time. Conversely, the superloop architecture experienced a critical failure, losing up to 58.18% of event data due to CPU locking during transmission. Therefore, the implementation of FreeRTOS is proven to be highly suitable for enhancing early theft detection systems in closed storage objects.

Keyword: *Early Detection, Theft, ESP32-CAM, Real Time Operating System (RTOS), Telegram.*

PRAKATA

Bismillahhirohmaanirrohim, Puji syukur ke hadirat Allah SWT atas segala limpahan rahmat dan karunia-Nya yang memungkinkan terselesaikannya skripsi dengan judul "Implementasi *Real-Time Operating System* (RTOS) Pada Sistem Deteksi Dini Pencurian Berbasis ESP32-CAM" ini dengan baik Shalawat serta salam juga tercurah kepada junjungan kita, Nabi Muhammad Rasulullah SAW, beserta keluarga, sahabat, dan para pengikutnya yang setia hingga akhir zaman. Skripsi ini disusun sebagai salah satu syarat untuk menyelesaikan program pendidikan pada jenjang sarjana dan mendapatkan gelar Sarjana Komputer (S.Kom) di Universitas Tanjungpura.

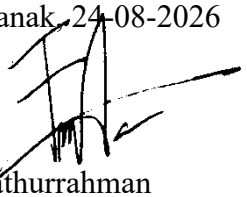
Selama proses penyusunan skripsi, banyak bantuan, dukungan, serta bimbingan yang diberikan oleh berbagai pihak. Atas segala kontribusi tersebut, ucapan terima kasih yang sebesar-besarnya disampaikan kepada:

1. Kedua orang tua beserta keluarga atas segala nasehat, kasih sayang dan kesabaran dalam mendidik penulis, serta senantiasa tiada hentinya memberikan doa, semangat, dan bantuan dalam bentuk apapun sehingga penulis dapat menyelesaikan skripsi ini.
2. Bapak Prof. Dr. Gusrizal, S.Si., M.Si., selaku Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Tanjungpura yang telah mengesahkan skripsi ini.
3. Ibu Rahmi Hidayati, S.Kom., M.Cs., selaku Ketua Jurusan Rekayasa Sistem Komputer FMIPA Universitas Tanjungpura Pontianak, sekaligus dosen pembimbing akademik (PA) serta pembimbing pertama penulis yang telah memberikan arahan dan saran yang sangat bermanfaat dalam penyusunan skripsi ini.
4. Bapak Tedy Rismawan, S.Kom., M.Cs., selaku dosen pembimbing Kedua penulis yang telah memberikan bimbingan, masukan, dan motivasi selama proses penyusunan skripsi ini.
5. Bapak Dr. Dedi Triyanto, S.T., M.T., sebagai dosen penguji pertama yang telah memberikan saran dan masukan dalam penulisan skripsi ini.
6. Ibu Irma Nirmala, S.T., M.T. sebagai dosen penguji kedua yang telah memberikan saran dan masukan dalam penulisan skripsi ini.

7. Ibu Kartika Sari, S.Kom., M.Cs., selaku dosen pembimbing kedua penulis periode sebelumnya yang telah memberikan bimbingan, masukan, dan motivasi selama proses penyusunan skripsi ini.
8. Bapak dan Ibu Dosen serta staf administrasi Jurusan Rekayasa Sistem Komputer yang telah senantiasa memberikan ilmunya selama masa perkuliahan, sehingga penulis dapat menyelesaikan skripsi ini .
9. Bapak Prima selaku Tenaga Administrasi bagian keuangan FMIPA yang telah memberikan kemudahan selama masa pembayaran kuliah.
10. Saudara Devin dan Muhammad Iqbal Nur yang senantiasa menemani, memberi semangat, dukungan, dan bantuan dalam bentuk apapun sehingga penulis dapat menyelesaikan skripsi ini.
11. Saudara Muhammad Faishal Hady, S.Kom., yang senantiasa memberi dukungan berupa bantuan dalam bentuk apapun sehingga penulis dapat menyelesaikan skripsi ini.
12. Teman-teman seperjuangan mahasiswa/i angkatan 2021 yang telah memberikan semangat, dukungan, dan kebersamaan selama proses penyusunan skripsi ini.
13. Semua pihak yang membantu baik secara langsung maupun tidak langsung dalam membantu penulis menyelesaikan skripsi ini.

Penulis menyadari bahwa dalam penyusunan skripsi ini masih jauh dari sempurna, sehingga untuk segala saran dan kritik yang membangun, penulis ucapkan terimakasih. Akhir kata penulis berharap skripsi ini dapat membawa manfaat bagi semua pihak yang menggunakannya.

Pontianak, 24-08-2026



M. Fathurrahman
H1051211108

DAFTAR ISI

LEMBAR PENGESAHAN ii
PERNYATAAN INTEGRITAS AKADEMIK iii
Abstrak iv
Abstract v
PRAKATA vi
DAFTAR ISI viii
DAFTAR GAMBAR xii
DAFTAR TABEL xiii
DAFTAR KODE PROGRAM xiv
BAB 1 PENDAHULUAN 1
1.1 Latar Belakang 1
1.2 Rumusan Masalah 4
1.3 Tujuan Penelitian 4
1.4 Batasan Masalah 4
1.5 Manfaat Penelitian 5
1.6 Sistematika Penulisan 5
BAB 2 LANDASAN TEORI 7
2.1 Perbandingan Dengan Penelitian Terdahulu 7
2.2 ESP32-CAM 9
2.3 Sensor PIR (*Passive Infrared Receiver*) HC-SR501 10
2.4 Sensor Vibrasi SW-420 11
2.5 Telegram 11
2.6 Arduino IDE 12
2.7 Konsep Kejadian (*Event*) dalam Sistem Tertanam Waktu Nyata 12
2.8 Konsep Dasar *Real-Time Operating System* (RTOS) 14
2.9 Manajemen Tugas (*Task*) pada RTOS 15
2.10 Algoritma Penjadwalan *Preemptive Priority-Based* 16
2.11 Antrean Pesan (*Message Queue*) 17
2.12 Perlindungan Data dengan Operasi *Critical Section* 17
2.13 Konsep Multi*task*ing dan Konkurensi 18
2.14 Tingkat Keberhasilan Perekaman (*Capture Rate / Accuracy*) 18

2.15 Waktu Tanggap (*Response Time*) 19
2.16 Waktu Tunda Komputasi (*Blocking Delay*) 20
BAB 3 METODE PENELITIAN 23
3.1 Studi Literatur 23
3.2 Pengumpulan Data 24
3.3 Analisis Kebutuhan 24
3.3.1 Kebutuhan Perangkat Keras 24
3.3.2 Kebutuhan Perangkat Lunak 25
3.4 Perancangan Sistem 25
3.4.1 Perancangan Perangkat Keras 25
3.4.2 Perancangan Perangkat Lunak 25
3.5 Implementasi 26
3.5.1 Implementasi Perangkat Keras 26
3.5.2 Implementasi Perangkat Lunak 26
3.6 Pengujian 27
3.7 Kesimpulan dan Saran 28
BAB 4 PERANCANGAN 29
4.1 Deskripsi Sistem 29
4.2 Perancangan Arsitektur Sistem 34
4.3 Perancangan Perangkat Keras 35
4.4 Perancangan Perangkat Lunak 37
4.5 Perancangan Antarmuka (User Interface) 41
BAB 5 IMPLEMENTASI, PENGUJIAN, DAN PEMBAHASAN 43
5.1 Implementasi Perangkat Keras 43
5.2 Implementasi Perangkat Lunak 46
5.3 Kode Program 49
5.4 Pengujian Sistem 54
5.4.1 Uji Fungsionalitas Logika Sensor 55
5.4.2 Pengujian *Response Time* Pada Arsitektur RTOS 55
5.4.3 Pengujian *Response Time* Pada Arsitektur Non-RTOS 63
5.4.4 Pengujian *Blocking Delay* Pada Arsitektur RTOS 66
5.4.5 Pengujian *Blocking Delay* Pada Arsitektur Non-RTOS 67
5.4.6 Pengujian *Capture Rate* Atau Kesinambungan Deteksi Pada Arsitektur RTOS dan Non-RTOS 68

5.4.7 Pengujian Stabilitas Sistem 69
5.5 Pembahasan 71
BAB 6 PENUTUP 75
6.1 Kesimpulan 75
6.2 Saran 76
DAFTAR PUSTAKA 77

DAFTAR GAMBAR

Gambar 2. 1 ESP32-CAM. 10
Gambar 2. 2 Sensor PIR (Passive Infrared Receiver) HC-SR501 10
Gambar 2. 3 Sensor Vibrasi SW-420 11
Gambar 2. 4 Ilustrasi Percobaan Pencurian (Sumber: WikiHow) 14
Gambar 2. 5 Task State (Sumber : Jatmiko, et al., 2015). 15
Gambar 2. 6 Algoritma Preemptive Priority-based Schedulling (Sumber : Jatmiko, et al., 2015). 16
Gambar 3. 1 Tahapan Metode Penelitian 23
Gambar 4. 1 Diagram Alir Sistem 32
Gambar 4. 2 Perancangan Arsitektur Sistem 34
Gambar 4. 3 Perancangan Perangkat Keras 36
Gambar 4. 4 Diagram Alir RTOS 39
Gambar 4. 5 Perancangan Antarmuka 42
Gambar 5. 1 Implementasi Sensor Gerak (PIR) 43
Gambar 5. 2 Implementasi Sensor Getar (SW-420) 44
Gambar 5. 3 Implementasi ESP32-CAM 44
Gambar 5. 4 Implementasi Perangkat Keras pada Sistem Pengamanan 45
Gambar 5. 5 Pembuatan Bot Melalui Layanan BotFather 47
Gambar 5. 6 Proses Mendapatkan Nomor Chat ID 47
Gambar 5. 7 Contoh Proses Pengiriman Berhasil. 48
Gambar 5. 8 Implementasi Sistem Berdasarkan Output Serial. 49

DAFTAR TABEL

Tabel 2. 1 Perbandingan Dengan Penelitian Terdahulu 7
Tabel 4. 1 Penggunaan Pin ESP32-CAM Untuk Sensor PIR Dan Sensor Vibrasi 36
Tabel 4. 2 Spesifikasi Perangkat Keras 37
Tabel 4. 3 Rancangan Pembagian Task RTOS, Core, dan Memori 40
Tabel 5. 1 Pengujian Fungsionalitas Logika Sensor 55
Tabel 5. 2 Deskripsi Fungsional Sistem 56
Tabel 5. 3 Identifikasi 8 *time*stamp Primer 57
Tabel 5. 4 Data Percobaan ke-1 *Response Time* Keseluruhan *Event* RTOS 59
Tabel 5. 5 Rekapitulasi *Response Time* RTOS Pada Percobaan Ke-1 60
Tabel 5. 6 Data Percobaan ke-2 *Response Time* Keseluruhan *Event* RTOS 61
Tabel 5. 7 Rekapitulasi *Response Time* RTOS Pada Percobaan K-2 63
Tabel 5. 8 Data Percobaan Ke-1 *Response Time* Keseluruhan *Event* Non-RTOS 64
Tabel 5. 9 Rekapitulasi *Response Time* Non-RTOS Pada Percobaan Ke-1 64
Tabel 5. 10 Data Percobaan Ke-2 *Response Time* Keseluruhan *Event* Non-RTOS 65
Tabel 5. 11 Rekapitulasi *Response Time* Non-RTOS Pada Percobaan K-2 66
Tabel 5. 12 Rekapitulasi *Blocking Delay* RTOS Pada Percobaan Ke-1 66
Tabel 5. 13 Rekapitulasi *Blocking Delay* RTOS Pada Pecobaan Ke-2 67
Tabel 5. 14 Rekapitulasi *Blocking Delay* Non-RTOS Pada Pecobaan Ke-1 67
Tabel 5. 15 Rekapitulasi *Blocking Delay* Non-RTOS Pada Pecobaan Ke-2 68
Tabel 5. 16 Perbandingan *Capture Rate* RTOS dan Non-RTOS Pada Percobaan Ke-1 68
Tabel 5. 17 Perbandingan *Capture Rate* RTOS dan Non-RTOS Pada Percobaan Ke-2 69
Tabel 5. 18 Stabilitas Perangkat Lunak 69
Tabel 5. 19 Rekapitulasi Perbandingan Parameter Pengujian (Percobaan 1 - Beban 31 *Event*) 72
Tabel 5. 20 Rekapitulasi Perbandingan Parameter Pengujian (Percobaan 2 - Beban 55 *Event*) 72

DAFTAR KODE PROGRAM

Kode Program 5. 1 Konfigurasi Sistem, Pin Sensor, dan Kamera.....49
Kode Program 5. 2 Struktur *Event* dan *Queue*.....50
Kode Program 5. 3 Sinkronisasi Shared Armed *State* dengan *Critical Section*...51
Kode Program 5. 4 Implementasi Multi-*Task*ing RTOS (PIR, Vibrasi, Action).51
Kode Program 5. 5 Implementasi Arsitektur *Superloop* Konvensional.....52
Kode Program 5. 6 Transmisi Jaringan Telegram dan Logging Data Serial 53

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Tingginya angka kriminalitas, khususnya Pencurian dengan Pemberatan (Curat), masih menjadi ancaman nyata bagi keamanan ruang privat, termasuk ruang penyimpanan tertutup seperti lemari rumah. Berdasarkan publikasi resmi Badan Pusat Statistik dalam Statistik Kriminal 2024/2025, kejahatan terhadap hak milik/barang terus mendominasi profil kriminalitas nasional, di mana kasus Pencurian dengan Pemberatan menyumbang jumlah kejadian yang sangat masif, yakni mencapai 62.872 kejadian pada tahun 2023 dan 51.969 kejadian pada tahun 2024. Meskipun secara keseluruhan di tingkat nasional angka tersebut tampak menurun, tingkat risiko penduduk terkena kejahatan secara umum (*crime rate*) masih berada pada level yang perlu diwaspadai, yaitu 204 orang per 100.000 penduduk. Terlebih lagi, ancaman ini terbukti semakin meluas, di mana persentase desa/kelurahan di Indonesia yang menjadi sasaran tindak pidana pencurian justru mengalami peningkatan dari 26,50% pada tahun 2021 menjadi 29,92% pada tahun 2024, menjadikannya sebagai jenis kejahatan paling dominan di lingkungan masyarakat (Badan Pusat Statistik, 2025). Kondisi ini menunjukkan perlunya sistem keamanan yang responsif, aktif selama pemantauan, dan mampu menyimpan bukti kejadian untuk ditinjau kembali. Sistem pengawasan konvensional seperti CCTV masih memiliki keterbatasan karena bergantung pada kapasitas penyimpanan lokal dan tidak proaktif dalam menelusuri kejadian (Wicaksana et al., 2025).

Pada sisi lain, sistem keamanan berbasis Internet of Things (IoT) sering menghadapi masalah alarm palsu dan penundaan pemrosesan data yang dapat menurunkan tingkat kepercayaan pengguna, sebuah fenomena yang secara psikologis dikenal sebagai efek *cry wolf* atau *alarm fatigue* (Barbierato, 2026; Ndichu et al., 2026). Sistem alarm konvensional sering kali bersifat statis (*latching*), di mana notifikasi cenderung tetap aktif atau terus berbunyi meskipun kondisi gangguan aktual sudah tidak ada. Oleh karena itu, arsitektur sistem peringatan perlu dirancang agar lebih adaptif dan responsif. Pendekatan ini memastikan bahwa notifikasi yang dihasilkan maupun status waspada sistem benar-benar

merepresentasikan kondisi ancaman di lapangan secara akurat dan *real-time*, sehingga sistem terhindar dari pengabaian oleh pengguna dan lebih dapat diandalkan (Roy et al., 2026; Sallam & Barachi, 2026). Dari sisi deteksi, penggunaan satu sensor saja masih berisiko memicu alarm palsu, sedangkan penggabungan sensor PIR dan ultrasonik terbukti dapat meningkatkan akurasi lokalisasi hingga 70,7% (Ciuffreda et al., 2023). Selain itu, analisis getaran pada mikrokontroler juga efektif untuk mendeteksi anomali mekanis, misalnya upaya paksa pada pintu atau jendela (Kolok et al., 2025).

Dalam konteks ruang penyimpanan tertutup, sensor getar SW-420 telah dibuktikan relevan untuk mendeteksi guncangan paksa pada objek tertutup seperti kotak amal (Ma'ruf et al., 2021). Untuk pengawasan area sekitar, sensor PIR juga lazim diintegrasikan dengan ESP32-CAM guna mendeteksi pergerakan dan mengirim notifikasi jarak jauh melalui Telegram (Wicaksana et al., 2025). Di sisi lain, penggunaan ESP32 sebagai unit komputasi pada sistem kritis juga telah menunjukkan skalabilitas dan efisiensi tinggi pada aplikasi berbiaya rendah, misalnya pada sistem kendali orientasi *CubeSat* (Mamani-Saico & Yanyachi, 2023).

Meskipun perangkat keras tersebut banyak diterapkan, sistem keamanan IoT masih menghadapi keterbatasan operasional, terutama latensi pengiriman notifikasi dan kehilangan pesan akibat jaringan yang tidak stabil. Penelitian Cakrayuda et al. (2025) menunjukkan bahwa pengiriman gambar ke Telegram rata-rata membutuhkan 2,7 detik (Cakrayuda et al., 2025), bahkan pada kondisi tertentu dapat mencapai 5,73 detik (Kurniawan et al., 2018). Keterbatasan ini menjadi lebih kritis ketika perangkat berjalan pada jaringan berbagi di lingkungan domestik yang dapat meningkatkan latensi dan kehilangan pesan (Mamani-saico & Yanyachi, 2023). Di sisi lain, sensor PIR juga memiliki keterbatasan sudut deteksi yang dapat menimbulkan *blind spot* (Wicaksana et al., 2025).

Berdasarkan temuan tersebut, penggunaan arsitektur polling sekuensial konvensional (*superloop*) berpotensi menimbulkan *blocking delay* saat ESP32-CAM memproses pengiriman data ke Telegram. Pada kondisi ini, pembacaan sensor berikutnya dapat tertunda karena sistem harus menunggu proses sebelumnya selesai, sehingga muncul celah waktu deteksi yang berada di luar keterbatasan fisik

sudut sensor (Kader & Budi, 2021). Hal ini selaras dengan prinsip arsitektur standar yang menahan proses program utama dan mencegah sistem untuk mengeksekusi pembacaan sensor berikutnya apabila instruksi transmisi jaringan belum selesai secara utuh (Jatmiko et al., 2015).

Dalam konteks sistem tertanam yang kritis, pendekatan penjadwalan konvensional yang memicu variasi penundaan waktu eksekusi ini sangat berbahaya karena terbukti merusak jaminan stabilitas dan fungsionalitas sistem secara keseluruhan (Raj, 2025). Secara empiris, kelumpuhan instruksi saat mikrokontroler (*superloop/bare-metal*) memproses koneksi jaringan ini mengakibatkan sistem gagal mengakuisisi pembacaan sensor pada interval waktu tersebut, yang berujung pada hilangnya data secara permanen atau *data loss* (Dehani et al., 2026).

Untuk meminimalisasi potensi kelumpuhan instruksi tersebut, penerapan *Real-Time Operating System* (RTOS) diusulkan sebagai solusi pendekatan perangkat lunak. Konsep RTOS diperkenalkan pada sistem tertanam untuk mengeksekusi program dalam pola yang teratur sehingga tidak ada tugas yang terlaksana melebihi batas waktu (*deadline*) yang ditentukan. Melalui algoritma *preemptive priority-based scheduling*, RTOS mampu membagi struktur program menjadi sejumlah tugas (*task*) diskrit yang dieksekusi berdasarkan kelas prioritasnya (Marettania et al., 2025; Jatmiko et al., 2015). Implementasi FreeRTOS pada arsitektur *multi-core* ESP32 telah terbukti mampu mengoptimalkan pembagian beban kerja secara efektif guna meminimalisasi latensi (Azhar et al., 2023). Raj (2025) menambahkan bahwa inovasi RTOS modern sangat mendukung sistem kritis yang membutuhkan manajemen sumber daya adaptif untuk menjaga stabilitas sistem. Selain itu, sistem operasi waktu nyata secara umum lebih andal dalam menjaga stabilitas, mengurangi fluktuasi latensi, dan mencegah kehilangan data saat sistem berada pada beban puncak (Ye et al., 2023).

Berdasarkan hal tersebut, penelitian ini difokuskan pada evaluasi komparasi arsitektur perangkat lunak untuk menguji sejauh mana implementasi algoritma penjadwalan tugas berbasis FreeRTOS memengaruhi durasi *blocking delay* pada ESP32-CAM dibandingkan metode *superloop*. Pengujian perbandingan akan dilakukan untuk membandingkan tingkat kesinambungan deteksi sistem antara kedua arsitektur tersebut melalui parameter *response time* dan *capture rate*, guna

menghasilkan cetak biru sistem keamanan yang andal dan tanggap dalam menjaga ketersediaan data.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan diatas, terdapat rumusan masalah sebagai berikut:

1. Bagaimana fungsionalitas dan kesesuaian logika pengabungan sensor PIR dan Vibrasi dalam mendeteksi kejadian pada sistem deteksi dini pencurian berbasis ESP32-CAM?
2. Bagaimana RTOS mengatur prioritas setiap *task* dan penjadwalan pada sistem deteksi dini pencurian berbasis ESP32-CAM?
3. Bagaimana perbandingan kinerja arsitektur FreeRTOS dan *superloop* dalam menjaga kesinambungan deteksi berdasarkan parameter *blocking delay*, *response time*, dan *capture rate*?

1.3 Tujuan Penelitian

Adapun tujuan dari penelitian ini sebagai berikut:

1. Memvalidasi fungsionalitas dan kesesuaian logika penggabungan sensor PIR dan Vibrasi dalam mendeteksi skenario kejadian pada sistem deteksi dini pencurian berbasis ESP32-CAM.
2. Menganalisis arsitektur RTOS untuk mengatur prioritas *task* dan penjadwalan pada ESP32-CAM.
3. Membandingkan kinerja FreeRTOS dan *superloop* dalam menjaga kesinambungan deteksi berdasarkan *blocking delay*, *response time*, dan *capture rate*.

1.4 Batasan Masalah

Adapun batasan masalah dari penelitian ini sebagai berikut:

1. Objek penelitian adalah sistem deteksi pencurian untuk lingkungan dalam ruangan pada lemari.
2. Perangkat keras difokuskan secara spesifik pada penggunaan mikrokontroler ESP32-CAM, sensor PIR HC-SR501, dan sensor getar SW-420.
3. Fokus penelitian ditekankan pada evaluasi komparasi arsitektur perangkat lunak antara metode *polling* sekuensial (non-RTOS) dengan arsitektur *multitasking* menggunakan kernel FreeRTOS.

4. Media pengiriman informasi peringatan jarak jauh dibatasi pada integrasi *Bot Application Programming Interface* (API) Telegram *Messenger*.
5. Pengujian stimulus kejadian fisik dilakukan secara manual sebagai simulasi gangguan pada objek lemari, dengan fokus pada kemampuan sistem dalam mendeteksi *event* beruntun, bukan pada pengukuran besar gaya mekanis guncangan.
6. Parameter yang dianalisis dalam penelitian ini dibatasi pada *blocking delay*, *response time*, dan *capture rate*.

1.5 Manfaat Penelitian

Manfaat yang dapat diperoleh dari penelitian ini adalah:

1. Memberikan rujukan akademis mengenai penerapan RTOS dalam mengatasi latensi dan celah waktu deteksi pada sistem IoT.
2. Menghasilkan purwarupa cetak biru sistem keamanan yang tanggap dan mampu merekam seluruh kejadian tanpa kehilangan data.

1.6 Sistematika Penulisan

Untuk memberikan gambaran yang menyeluruh dan sistematis mengenai penelitian ini, maka sistematika penulisan tugas akhir ini disusun sebagai berikut:

BAB 1 PENDAHULUAN

Bab ini menjelaskan latar belakang masalah, rumusan masalah, tujuan penelitian, batasan masalah, manfaat penelitian, dan sistematika penulisan.

BAB 2 LANDASAN TEORI

Bab ini memaparkan tinjauan penelitian terdahulu dan teori yang menjadi dasar penelitian, meliputi ESP32-CAM, sensor PIR HC-SR501, sensor getar SW-420, Telegram, Arduino IDE, konsep RTOS, *task scheduling*, *message queue*, *critical section*, *capture rate, response time,* dan *blocking delay*.

BAB 3 METODE PENELITIAN

Bab ini menjelaskan tahapan penelitian yang meliputi studi literatur, pengumpulan data, analisis kebutuhan, perancangan sistem, implementasi, dan pengujian sistem.

BAB 4 PERANCANGAN

Bab ini menguraikan deskripsi sistem, perancangan perangkat keras, perancangan perangkat lunak, serta rancangan arsitektur non-RTOS dan arsitektur

RTOS.

BAB 5 IMPLEMENTASI, PENGUJIAN, DAN PEMBAHASAN

Bab ini memuat implementasi perangkat keras dan perangkat lunak, kode program, hasil pengujian, dan pembahasan. Pada bab ini dilakukan analisis perbandingan arsitektur RTOS dan *superloop* berdasarkan parameter *blocking delay*, *response time*, dan *capture rate*.

BAB 6 PENUTUP

Bab ini berisi kesimpulan akhir yang disusun berdasarkan hasil penelitian untuk menjawab rumusan masalah, serta saran untuk pengembangan penelitian selanjutnya.