

***Design of DC Motor Control System for Conveyor Based on Master Slave  
(Raspberry Pi–ESP32) for Cocoa Fruit Detection Device***  
Nuzula Afianah, S.Kom., M.Cs. (Chief Counselor)

**Aldhi Achmad Muzaki**  
*Study Program of Mechatronics Engineering Technology  
Majoring of Engineering*

***ABSTRACT***

*The manual sorting process of cocoa fruit has drawbacks including inconsistent results and limited work capacity, necessitating an automated system capable of improving process efficiency and consistency. This study aims to design and implement a DC motor control system for a conveyor based on a master-slave architecture using Raspberry Pi 5 and ESP32, and to analyze system performance including conveyor speed regulation through Pulse Width Modulation (PWM) signal variation, the effect of load on DC motor performance, RS-485 communication reliability, and the performance of the Graphical User Interface (GUI). The methods used include integrated panel box-based hardware design, software implementation on Raspberry Pi 5 using Python with a YOLO model for fruit detection, and an ESP32 program using C++ for real-time actuator control. Test results show that increasing PWM values from 20% to 100% produces motor speeds from 66.5 RPM to 487.6 RPM proportionally. Load effects cause the largest RPM drop at PWM 20% of 52.48% and the smallest at PWM 60% of 11.49%. RS-485 communication showed a 100% success rate with average response times of 31.35 ms for the START command, 26.39 ms for the STOP command, and 23.19 ms for the SORT command, with no data loss across all PWM variations. The servo-based sorting mechanism achieved an 85% success rate from 20 trials, with failures attributed to mechanical factors related to variations in cocoa fruit shape. All GUI components functioned successfully with a 100% success rate based on black-box testing. This system has been proven capable of working in an integrated manner as a reliable prototype of an automatic cocoa fruit sorting device.*

**Keywords:** *conveyor, ESP32, DC motor, Raspberry Pi, RS-485*