Conveyor Reject System with Consecutive Bad Part Lockout

---

1. Project Summary

This PLC-based simulation models a conveyor system designed for part transportation and automated rejection of defective items. The system runs on Rockwell Studio 5000 using RSLogix Emulate and applies fault logic to stop operations after 10 consecutive rejections. The logic mimics real-world reject cells found in packaging, quality inspection, or pharmaceutical automation lines.

---

2. Functional Overview

- **Start/Stop motor control** with latch logic
- **Part detection** using simulated sensor
- **Bad part signal** from manual toggle (vision simulation)
- **Reject diverter** activates on bad detection
- **Good and bad parts counted** separately
- **Lockout condition** triggered after 10 consecutive bad parts
- **Manual fault reset** with Sys_Reset
- Uses ONS, TON, CTU, RES, JSR instructions

---

3. Routine Structure

Routine Name	Description
MainRoutine	Calls all subroutines using JSR
Map_IO	Maps all simulated input/output tags
Process_Logic	Main control logic for motor and diverter
Faults_and_Reset	Fault detection and system reset logic

---

4. Core Tags

Tag	Type	Purpose
PB_Start	BOOL	Start pushbutton
PB_Stop	BOOL	Stop pushbutton
Part_Sensor	BOOL	Simulates a part passing over sensor
Bad_Part_Sensor	BOOL	Indicates part is bad
Conveyor_Motor	BOOL	Motor output coil
Part_Diverter	BOOL	Activates reject pusher
Consec_Bad_Part	COUNTER	Counts bad parts in a row
Batch_Fault	BOOL	Fault if 10 bad parts occur consecutively
Sys_Reset	BOOL	Resets system fault and bad part counter

---

5. Fault Logic Description

- A CTU increments Consec_Bad_Part on every bad part
- A good part triggers a RES of that counter
- When. DN bit becomes TRUE (i.e. 10 bad parts), Batch_Fault latches
- Conveyor cannot run until Sys_Reset is pressed
- Diverter controlled by TON to ensure pulse rejection

---

6. Conclusion

This project reflects real-world machine logic for quality enforcement. It showcases how to use structured routines, proper tag handling, and safety-oriented logic flow to prevent ongoing production of rejected material. The same pattern can scale to larger batch handling systems or vision-integrated reject cells.
