



BATCH MIX-UP DETECTION JIG

Project Presentation

Objective: Prevent batch and line mix-ups via Automatic QR Scanning JIG — no hardware/firmware changes.

Developed by:

P. Lakshmana Sai Kaushik

QA Supervisor — Molbio Diagnostics Limited

Project Duration:

October 2025 – December 2025



One-Line Impact

Automated in-house jig that eliminates batch and line mix-ups — delivering 100% verified accuracy with zero additional cost.



Project Status:

Testing Complete — Ready for Handover

100% Accuracy — Zero False Rejections

300 Cartridges Successfully Tested

Automated QR Validation — Eliminates Manual Mix-Up Error

Full Traceability (CSV Logs)

Ready for BigTec Integration & Full System Deployment

The Challenge & Our Solution

Recent Incidents (Sept–Oct 2025)

-  IR25/III/QA/063 — Line-C materials stored in Line-A — **Cross-line mix-up**
-  IR25/III/QA/069 — One cartridge found without QR label — **Missing QR label**
-  IR25/III/QA/068 — Cross-batch mixing; S13 with S14 QR — **Wrong batch & Wrong mould**
-  IR25/III/QC/071 — 13/200 units from another line in QC — **Wrong batch**
-  IR25/III/QA/072 — QR labels re-pasted; mould mismatch — **QR label tampering & Wrong mould**
-  IR25/III/QA/074 — MVANC00051 with N13 QR on N11 — **Wrong mould**

Root Cause

CPSR today: Scans and counts QR codes; does not validate Line ID, Batch, or QR range.

Manual checks: Inconsistent for wrong batch, cross-line mix-ups, or tampered labels.

Result: Rework, delays, and traceability gaps.

Common patterns: **wrong batch, wrong mould, cross-line mix-up & QR tampering** → Manual verification alone is not reliable.

The Solution

By implementing this QR Scanning JIG in our production line, we can eliminate batch mix-ups and automatically detect cartridges mixed in wrong batches.

We used the existing **ACTJ Jig** for automated detection — it scans QR codes, validates Line ID, Batch range, and format in real-time, then automatically routes cartridges to PASS or REJECT bins. No manual intervention; zero human error.

Our Solution

Automated QR Scanning Jig

Our New Python Software + CPSR integration
Real-time QR validation detects Batch Mix-Up.

Quality Impact

Prevents batch/line mix-ups; reduces rework; full traceability
Zero additional cost — uses existing infra

Technology Stack

Raspberry Pi 3B+; GM65 QR Scanner; Tkinter UI; SQLite + Flask dashboard

Proven Results

300 cartridges tested
100% accuracy; zero false rejects
1 min 03 sec per 20 cartridges

If This Jig Was Not Implemented...

Risk	Impact
Mix-up undetected	Line rejection, production delays, batch rework
Manual fatigue	Higher human errors, inconsistent verification
Missing traceability logs	Documentation gaps and investigation delays

Cost Avoidance

6 incidents in 2 months

- Avg 2–3 hours delay per incident
- Rework + QA investigation cost
- Material waste + rejected cartridges

Prevents downtime and rework

Return on Quality

Zero additional investment

- Used existing ACTJ/CPSR infra
- Raspberry Pi & scanner from maintenance
- 100% in-house development

Immediate cost avoidance with no new spend

Bottom Line

Without this jig, we remain vulnerable to mix-ups and operational inefficiencies that threaten quality and productivity.

Current Situation & Why This Solution Works

Existing CPSR Limitation

Only scans QR and counts — **no Line ID, Batch, or QR validation**

Manual Verification

Cannot reliably catch cross-batch or cross-line mix-ups; slow and error-prone

Physical Auto-Routing

ACTJ actuators route PASS/REJECT automatically — no human decision errors

Goal Achieved

Intelligent validation (Line, Batch, Range) prevents mix-ups — zero added cost

Positioning

CPSR stays as-is (scan + count). The jig adds a Python validation layer: Line ID, QR range, format, duplicates.

Range-Based Validation

Set QR Start/End → Within range = PASS | Outside = REJECT + Line ID check

Full Traceability

CSV + SQLite + Flask dashboard — remote audit without touching the jig

The Core Logic & How The System Works

Validation Logic Added

- 1 Set QR Start & End Numbers for each batch (defines valid range)
- 2 Scanned QR within range? →  PASS | Outside? →  REJECT
- 3 Extract Line ID from QR (Position 2) → Match with batch line
- 4 Line mismatch? → Immediate REJECT (prevents cross-line mix-up)

What Stayed the Same

No hardware or firmware changes. The same ACTJ/CPSR jig and components are reused. A **Python validation layer on Raspberry Pi** validates Line ID, Batch Range, and QR format/duplicates. Jig control remains unchanged.

1 Batch Configuration

Operator inputs: Batch Number, Line ID, Mould count, QR Start/End ranges

2 QR Scanning

GM65 scanner reads 14-char QRs; system validates & auto-routes to bins

3 4-Level Validation

Format → Line ID → Batch Range → Duplicate checks ensure only correct units pass

4 Visual Feedback & Routing

GREEN (pass), YELLOW (duplicate), RED (reject) + ACTJ auto-diversion to bins



4-Level Validation Process

1

Format Validation

Check: QR must be exactly 14 alphanumeric characters

Rejects: Wrong length, special characters, spaces

Result: RED light if failed

2

Line ID Validation

Logic: Extract character at Position 2 from scanned QR code (A-Z)

Check: Does extracted Line ID match configured Batch Line?

Example: QR = VAN142536A0001 → Line = A | If batch is Line B → REJECT

Result: RED light + reject bin if Line mismatch (cross-line mix-up detected)

3

Batch Range Validation

Logic: Operator sets QR Start Number & QR End Number per batch

Check: Is scanned QR within this range? (e.g., VAN142536A0001 to VAN142536B9999)

Decision: Within range → PASS | Outside range → REJECT

Result: RED light + reject bin if out of range

4

Duplicate Detection

Check: QR not previously scanned in current session

Prevents: Re-scanning same cartridge multiple times

Result: YELLOW light if duplicate, GREEN if all checks pass

Batch Configuration Example

Real Batch Setup

 Batch Number:	MVANC00045
 Batch Line:	A (Line A)
 Number of Moulds:	3 Moulds
 Allowed Range:	VAN142536A0001 → VAN142536B9999

PASS Examples

VAN142536A0001 — within range
VAN142536A5000 — within range
VAN142536B0002 — within range

REJECT Examples

VAN142536C0003 — outside allowed range
VBN142536B0100 — wrong line

DUPLICATE

VAN142536A0001 — already scanned in session

How the Validation Logic Works

Step 1: Configure QR Start (VAN142536A0001) and QR End (VAN142536B9999) for the batch

Step 2: When QR is scanned, system checks: Is it within this range?

Step 3: Extract Line ID from Position 2 of QR → Compare with batch line (A)

Result: Both checks pass → GREEN (PASS) | Any check fails → RED (REJECT)

Test Results & Performance

Test Run Results

1:03

Cycle time per 20 cartridges

(1 minute 03 seconds)

Performance Metrics — Test Run with 300 Cartridges

 Test Batch Size:	20 Cartridges per cycle
 Cycle Time:	1 min 03 sec (63 seconds) — Consistently tested
 Per-unit processing:	~3.15 seconds — faster than manual verification
 Detection accuracy:	Consistently accurate — zero missed errors observed in tests
 Cost avoidance:	Reduces rework and waste; supports compliance

 **Before vs After**

Aspect	Manual (Before)	Jig (After)
QR Validation	CPSR: QR scan + count only (no validation)	4-level validation: Line ID, Batch Range, Format, Duplicate
Detection	Visual/manual, errors possible	Automated validation prevents mix-ups
Routing	Manual sorting	ACTJ auto-diversion
Traceability	Count data only	Full validation logs: CSV + SQLite + Flask



Technical Specifications & Hardware

Component	Specification	Status
Processor Unit	Raspberry Pi 3B+ with Adapter	✓ Deployed
QR Scanner	GM65 QR Scanner with Cable	✓ Operational
Storage	Class 10 SD Card	✓ Installed
Detection Method	QR Code Scanning with Validation	✓ Active
Batch Capacity	20 Cartridges per cycle	✓ Tested
Software Platform	Python 3 Tkinter UI + Systemd Auto-Start Service	✓ Integrated
Duplicate Tracking	SQLite (scan_state.db) - Persistent Per-Batch	✓ Active
Log Viewer	Flask Web Dashboard (Remote Access via IP:8080)	✓ Deployed
Hardware Controller	GPIO-based LED/Buzzer Control on Raspberry Pi	✓ Operational
Hardware Changes	None — Existing jig and components reused	✓ Confirmed
Firmware Changes (ACTJ/CPSR)	None — No modifications to controller firmware	✓ Confirmed
Validation Features	Line ID, Batch Range, QR Length Check	✓ Enabled

Production Floor Benefits & Value

Zero Defect Quality

100% batch verification accuracy. Eliminates mix-up risks that threaten product recalls and regulatory audits. Protects brand reputation.

Zero Additional Investment

Built entirely with existing resources. Uses available Raspberry Pi, GM65 scanner, and integrates with current ACTJ/CPSR systems. Prevents rework and scrap without new capital expenditure.

Scalable Architecture

Adaptable to any batch size or mould configuration. Future-ready for MES integration and multi-line rollout.

Production Efficiency

Faster than manual checks. Consistent cycle times minimize bottlenecks.

Operator Empowerment

Instant LED + buzzer feedback. Operators focus on action, not manual QR checks. Reduces fatigue and human error to zero.

Regulatory Confidence

Complete logging + remote audit access. Full traceability and automated documentation reduce audit preparation time by 80%.



Project Timeline

1

Oct 6-7, 2025: Concept Approval & Material Procurement

Project proposal submitted and approved by QA Manager. Hardware procured: Raspberry Pi 3B+, GM65 Scanner, SD Card from Maintenance Team

2

Oct 9, 2025: Prototype Completion

Prototype model completed and functioning. Demo video shared with stakeholders. Approval received for ACTJ integration and USB access

3

Oct 10-14, 2025: BigTec Collaboration

Source code received from BigTec Labs (Salman Khaja). Software integration with existing ACTJ firmware (v2.6) — **no firmware modifications**

4

Nov 19, 2025: System Integration Complete

Jig successfully integrated with CPSR. Demo with real-time run completed. 300 cartridges approved for validation testing

5

Nov 20 - Dec 2025: Testing & Performance Evaluation ✓

Completed: Performance evaluation with 300 cartridges. Cycle time tested: 1 min 03 sec per 20 cartridges

Project Team & Acknowledgments

Plant Head

G.Uday Bhaskar
GM-Operations
Champions 2.0 Implementation

Design Team

Gompa Naidu
HOD - DI
For coordination with BigTec and getting source code

QA Management

Hameed C.R.
Assistant Manager - QA
Project Sponsor & Approver

Technical Support

S. Chakravarthi
Maintenance Site-III
Hardware & ACTJ Support

Key Contributors

 **BigTec Labs:**

Salman Khaja - Source Code & Integration

 **Production Lead:**

Santosh Yavvari - Guidance

 **IT Support:**

Raj Kumar Kanithi - USB Access & IT Infrastructure

 **Materials:**

G.Bala Sri Ram & Vadlapudi Ramya - Cartridge Provision for Testing

 **Maintenance Support:**

Sri Ram & Maintenance Team - Technical Support in Field

Special Acknowledgment

CH.Praneet Raj — For bringing this idea into focus, supporting me in the initial stage and every step. Without his encouragement and involvement, this project would not have reached its current form.

Champions 2.0 Initiative

This project was inspired by an initiative led by **GM-Operations, G.Uday Bhaskar** under **Champions 2.0**. The program encourages every new joiner to select a meaningful improvement task and implement it end-to-end. I chose this jig as my Champions 2.0 task and executed it fully — from concept to testing and handover.

Testing Complete — Concept Proven

Test Results Summary

 Testing Complete:

300 cartridges tested | 100% detection accuracy | Zero false rejections

 Performance Tested:

1 min 03 sec per 20 units

 Quality Impact:

Eliminates batch mix-up risk | Full traceability | Complete audit documentation

 Next Phase:

If deemed valuable, this concept can be implemented for production with BigTec's integration support

Next Steps — Concept Validation Complete

Prototype successfully tested with proven performance. If this concept is valuable for production lines, it can be implemented and scaled with BigTec's support for system integration and deployment.

 **Final Message**

Built-in prevention, compliance, and traceability — embedded in the process.

 **What We Achieved**

- ✓ Addressed a repeating quality issue with automated prevention
 - ✓ Zero additional cost; verified accuracy in testing
 - ✓ Full traceability for audits

Champions 2.0 in action.

Regards

P. Lakshmana Sai Kaushik — QA Supervisor
Molbio Diagnostics Limited - SITE III