

THE CHALLENGE



Rewriting a Complex System

A complex system without a standard code increased downtime and created bottlenecks.



To run smoother, the system needed a complete standardized rewrite, but this can be a long, daunting, and risky process.



System Without Standard Code



The surveying system was difficult to maintain and support due to the system being over a decade old. Code that controlled the entire system had



been changed multiple times and was not standardized causing productivity to suffer.

Cracking the Code: Standardization

Solution: Debug and Startup

Handled correctly, a full code rewrite does not need to be disruptive, and the ROI can be huge. To duplicate preexisting logic in a standardized format, the code standard needed to be identified, developed, and implemented to meet the need of the manufacturer. Then, the system went through a series of tests to get it up and running properly. The code rewrite exceeded the manufacturer's goals and led to higher throughput, greater reliability, decreased downtime, and faster speeds.



EXECUTIVE SUMMARY

▶ Consumer Product Manufacturer Faces Standardization Question

Client: Global Consumer Product Manufacturer

Challenge

- Old and complex conveying system was riddled with code issues. Troubleshooting the system necessitated downtime that hurt productivity.
- A complete re-write would streamline and standardize the code, but would also require shutting down the plant, inviting significant risk.

Solution

- Create a defensible plan that makes the case for the risk and cost of the project and allows time for resolving unanticipated problems.
- Completely rewrite code
- Resolve unknown – but planned for – hardware issues

Results

- Proving that a full code re-write does not have to be disruptive, thorough planning for debug and startup reduces both downtime and upgrade cost.
- System up and running in just a few days
- Throughput increased

Cracking the Code

A complex system that represents millions of dollars in capital investment was running slowly, creating bottlenecks, and racking up downtime.

Code that is controlling the entire system has been changed multiple times by many different people, and nothing is standard. Even hardware problems are difficult to correct because they're masked by code issues.

A complete rewrite that includes standardization could make everything run smoother, but it could be costly and includes risks of its own.

The plant manager for a global consumer products manufacturer in the Midwest didn't know what to do. His plant's complex conveying system ran 24/7 to carry finished product to palletizers. It was worth tens of millions of dollars, but most of it was 10-15 years old meaning it had become increasingly difficult to maintain and support. Compounding the issues, it was creating downtime on the upstream production systems.

A Complex System Without a Standard

The system itself was anything but simple. It had over 20 outputs from packaging going to seven palletizers, including six PLCs, around 400 motors, two dozen barcode-scan points – plus numerous merges, sorters, switches, diverts, and more, all along miles of conveyors.

To complicate things further, the plant's wide variety of product formats meant that small, light, poly-wrapped product was running on the same line as large, heavy-cased product.

Productivity was suffering, and troubleshooting the system required excessive downtime. Taking the equipment down for standardization would increase the chance that changes to the code would create unexpected, new problems during startup. Ultimately, how long would the entire plant be disabled?

The plant manager needed to accurately assess his options and construct a plan but was unsure where to start.

Time for a Full Code Rewrite

Anytime a manufacturer takes on a complete rewrite of code, it takes on multiple risks:

- Job complexity can become a minefield of delays and cost overruns (dozens of complex apps, each requiring thousands of lines of code, in a variety of languages)
- Rearchitecting, database work, and the risks of unknown interdependencies compound the risk
- Older equipment may no longer be supported by vendors

Weighing these risks, the manufacturer's team along with Polytron, considered the feasibility of several options. They determined that the only way to completely resolve the conveying system's issues was a full code rewrite. This would also require new programming standards and naming conventions.

An undertaking of this magnitude would raise many questions from corporate capital planners, so the plant's operations, engineering, and maintenance teams set to work with Polytron to develop an airtight plan.

After systematically working through all tough issues with the Polytron engineers, the plant manager was confident that standardization was worth the risk and the cost – and that he could make that argument to upper management.

Seven Steps to a Robust New Solution

Handled correctly, a full code rewrite does not need to be disruptive, and the ROI can be enormous. To duplicate pre-existing logic in a standardized format, the Polytron team took the following steps:

1. Identify and Create the Standard

The team customized Polytron's code standard to meet the manufacturer's specific needs, outlining details of the programming structure, nomenclature, tags and standard routines. A functional specification was developed to communicate the standard in a written form easy for anyone to use.

2. Develop a Logic Narrative for the System

Next, existing code was reverse-engineered and listed out as a logic narrative describe how the equipment works with the conveyor. The team confirmed each operation or made new determinations of how the PLCs would interact with equipment, naming conventions, and other factors.

3. Develop the Code

The Polytron team developed the code in an entirely new program file. The code was object-based to ensure strict adherence to the developed standard.

4. Test the System Using an Emulation Model

The manufacturer's actual factory situations were then emulated, creating a computer-based model of the conveying systems controlled entirely by the new PLC and HMI code to fully test the new controls system – without risk to production.

5. Conduct in-house Factory Acceptance Test (FAT)

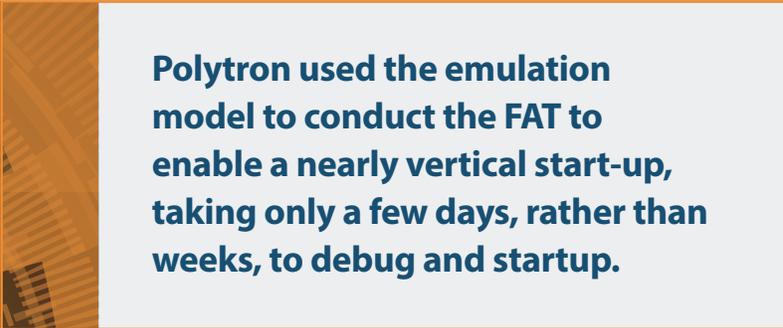
The system was pushed to its limits, essentially trying to "break the system," so that the customer would be confident in the ability of the new code to manage their product.

6. Field Installation and Testing

The same scenarios were validated in the field using the emulation model. The code was solid, and in the very first week, the team recognized a significant improvement to system performance in reduction of partials and downtime. Polytron's emulation model to conduct the FAT enabled a nearly vertical start-up.

7. Formal Training and Documentation

Since success depends on the ability of plant personnel to support the system, Polytron conducted detailed, code-specific training. Originally, this training had been scheduled to last four hours. However, the training was so helpful that the company extended it to include two full days.



Polytron used the emulation model to conduct the FAT to enable a nearly vertical start-up, taking only a few days, rather than weeks, to debug and startup.

Results: Debug and Startup Reduces Downtime and Cost of Upgrade

The code rewrite exceeded the manufacturer's goals:



- Newly standardized system delivers higher throughput, greater reliability, and faster speeds
- Partials are now less than 15% - and falling
- Downtime has been cut in half
- Significant improvement in servicing, manufacturing and packaging - their upstream customer
- Notable decrease in disruption to warehouse workforce
- Standardized system problems are now much easier for the manufacturer to resolve
- Code rewrite enables programmers and technicians to use the code to identify root cause of productivity problems (factors such as hardware issues or product handling issues)



This success prompted the manufacturer to prioritize deploying the solution throughout the remainder of their systems. Currently engaged in improvement processes throughout the plant, Polytron has completed all four phases of rewriting code and optimizing their remaining systems.

About Polytron, Inc.

Since 1983, Polytron has been an industry leading system integration and engineering consulting firm delivering a broad spectrum of innovative manufacturing solutions. Polytron serves manufacturers in the food, beverage, consumer packaged goods, chemical, and life sciences industries across North America.

To learn more about Polytron, visit us online (www.polytron.com) or contact us (www.polytron.com/contact-us) to talk to a specialist today.