



SERVICE BRIEF

Brownfield and Greenfield Code Development

PLC and SCADA software written to your standard, or ours, and proven before it reaches site.



Control system engineering, integration and analytics since 1997
ISO 9001 certified • Cyber Essentials • CHAS • Constructionline • SafeContractor

www.matrixsystems.co.uk
enquiries@matrixsystems.co.uk
0121 240 7454

THE CHALLENGE

Software your engineers can own

Control software outlives the people who write it. On an existing plant it is often patched by several hands, lightly documented and risky to touch. On a new one, every supplier brings its own style, and you inherit the mix. Either way, the code has to run your process safely and stay supportable for decades.

Code nobody fully understands

Years of changes by different engineers leave logic that works but cannot be explained, so every modification feels like a gamble with production.

Your standard, not theirs

You have a coding standard and a block library for good reason. Suppliers who ignore them hand over software your team cannot maintain.

Faults found on site

Logic errors found at commissioning cost the most: idle crews, delayed production and pressure to fix things live.

Documents that drift

Specifications, I/O lists and alarm lists fall out of step with the code, and no longer describe what actually runs.

AT A GLANCE

LANGUAGES

Ladder, FBD and structured text

Statement list where an existing program uses it, and safety programs on safety PLCs.

PLATFORMS

Multi-vendor

Siemens TIA Portal and S7, Rockwell, Schneider, Mitsubishi, Beckhoff and others.

STANDARDS

Designed to recognised practice

IEC 61131-3 languages, ISA-88 structure, IEC 62682 alarm management, IEC 62443 security.

OUR ANSWER

Structured, standards-led software, tested before site and documented as it runs.

We write to the standard you choose: yours, ours, or one we build with you. Existing code is analysed and documented before anything changes. New code is built from your specification. Both are version-controlled from the first line, tested against a simulated plant, and handed over with the documents and support your team needs.

WHAT WE OFFER

Six ways we develop your software

Whether we extend a running plant or start from a blank page, you choose the standard the code is written to and how it is produced.

OPTION 01

Your standard, your blocks

We write to your company coding standard, naming rules and block library, so the result looks like your own engineers wrote it. Your library blocks are reused, not copied and altered.

OPTION 02

Our standard

Where you have no standard, we use ours: a consistent structure of equipment modules, state-based sequences, alarms and interlocks, laid out the same way on every project.

OPTION 03

A standard built with you

We create a standard with you: structure, naming, block library, templates and coding rules, documented so every future supplier and engineer follows it too.

OPTION 04

Generated from your examples

Give us one well-written example of a motor, valve or sequence. We generate the rest of the project in the same form, so code is consistent and quicker to review.

OPTION 05

Brownfield: understand, then change

We analyse your existing code first: structure, sequences, I/O, alarms and interlocks. Missing documents are recovered from it, and changes are made under control.

OPTION 06

Greenfield: specification to code

New systems start from an agreed functional specification. The software is built from it in a clear structure, so every function traces back to a requirement.

MIX AND MATCH

Most projects combine options

An extension to a running line usually keeps your blocks and the structure already there. A new line beside it can be generated from your examples, so the two look and behave alike. We agree the mix at specification stage and record it in the software design specification.

BROWNFIELD AND GREENFIELD

Two starting points, one discipline

Most projects mix the two: new equipment on an existing plant, or a new line that must match the old one. The approach changes with the starting point; the standard of evidence does not.

Brownfield: an existing plant

The software is running and production depends on it. We change it without disturbing what works.

- A restorable backup of what runs today, taken before anything changes
- Code analysed and I/O, alarm and cause-and-effect lists recovered from it
- Changes written in the style and structure already there
- Each change tested in simulation against today's behaviour
- Cut-over planned around your outages, with a rollback plan

Greenfield: a new system

Nothing exists yet. We get the structure right from the start, so the system stays supportable for its whole life.

- Functional design specification agreed before code is written
- Standard chosen up front: yours, ours or a new one
- Consistent code generated from the plant's equipment list
- Software proven against a simulated plant from the first build
- Full document set produced alongside the code, not after it

What we develop

Area	What we develop	How it is proven
PLC software	IEC 61131-3 code in ladder, function block diagram and structured text; statement list where the existing program uses it	Compiled clean, reviewed, sequence and interlock tests
Safety programs	Safety functions on safety PLCs and relays, written to your safety requirements specification	Safety requirements tests, independently reviewed
HMI and SCADA	Screens, alarms, trends, reports and operator commands, consistent with the PLC structure	Tested against the PLC running in simulation
Drives and devices	Drive, instrument, vision and robot interfaces, with parameters archived with the project	Interface tests and parameter records
Communications and data	PROFINET, EtherNet/IP, Modbus and OPC UA links, historians, reports and MES or ERP interfaces	Point-by-point communications tests
Version control	Every change committed with its reason, reviewed and traceable from the first line to handover	Change history handed over with the source

HOW WE DELIVER

From requirement to supported system

Development is one part of the job. Testing, commissioning, documentation and support come with it as standard.

01 Specify

Requirements, standards and risks agreed with you; on existing plant, the code analysed and documented first.

02 Standardise

The coding standard, block library and templates fixed before development starts.

03 Develop

Code written or generated to that standard, version-controlled and peer reviewed as it grows.

04 Test

Sequences, interlocks, alarms and faults tested against a simulated plant, then factory acceptance testing.

05 Commission

Installation, commissioning and site acceptance testing, planned around your operation.

06 Support

Documentation, training and long-term support, with 24/7 cover under a support agreement.

OUR ENGINEERING TOOLCHAIN

Software that checks and builds software

Code analysis

Maps the blocks, sequences, loops, I/O and alarms of a TIA Portal project before we change it.

Document recovery

Produces I/O, alarm and cause-and-effect lists and a functional description from the code.

Code generation

Builds I/O mapping, state machines, alarms and loops from an equipment list.

Compile checks

Each change is imported and compiled automatically before an engineer reviews it.

Simulation

Programs run in PLCSIM Advanced with simulated inputs and process models.

Version control

Every PLC has its own repository; every change is reviewed before merging.

PROOF, NOT PROMISES

How we prove it

- Every function traced from the specification to its code and its test
- Code reviewed by an engineer who did not write it
- Sequence, interlock and alarm tests run against a simulated plant before factory acceptance
- Deviations logged with their fix and retest; nothing closed silently
- On running plant, every change staged from offline to simulation to live, each step signed off

WHERE THE LINE IS

Simulation proves logic against a model of your plant, within that model. It never replaces tests on the real equipment, and it is never safety acceptance. Process limits and safety targets come from your process design and hazard studies; we implement, test and record them.

WHAT YOU RECEIVE

Software and the documents that explain it

✓ Functional design specification

✓ Software design specification

✓ Coding standard and block library

✓ I/O list and cause-and-effect

✓ Alarm list and HMI interface

✓ FAT and SAT protocols and records

✓ Source code with its change history

✓ As-built drawings and evidence index

✓ Training and fault method statements

QUESTIONS CUSTOMERS ASK

Will you use our standard?

Yes. If you have a coding standard or block library, we write to it. If you do not, we offer ours or build one with you.

Our code is undocumented. Can you still work on it?

Yes. We analyse it first and recover the I/O list, alarm list, cause-and-effect and a functional description.

Who owns the software?

You do. You receive the source, its history and the documents, so you can support it yourself or with any supplier.

ABOUT MATRIX

Standards-led, fully documented control systems that you own and can support.

Matrix International (UK) Ltd has designed, built and supported industrial control systems since 1997. We take projects from specification and electrical design, through PLC, DCS and SCADA software, to installation, commissioning and long-term support, and we build the analysis software that shows how a plant is really performing. We are ISO 9001 certified and hold Cyber Essentials, CHAS, Constructionline and SafeContractor accreditation.

Siemens TIA Portal, S7 and PCS 7 • Rockwell • ABB • Schneider • Mitsubishi • Beckhoff • Emerson DeltaV • Honeywell • GE • WinCC, Ignition, AVEVA (Wonderware), FactoryTalk, iFIX and Citect SCADA

Talk to us about your software

enquiries@matrixsystems.co.uk

0121 240 7454

www.matrixsystems.co.uk