System Software Design

JCE Digital is involved in many different types of 'System Software Design'. With the diversity of equipment in which we deal, comes a diversity of applicable software engineering environments.

● Procedural Framework

Irrespective of its nature or application, all software produced in JCE Digital is subject to the same procedural framework. The extent to which the framework is applicable to any particular project is dependent on its scale, but this approach does provide a formalised interface for all parties.

Functional Specification Documents

- System Functional Description
- State Diagrams
- Flow/Structure Charts
- Cause & Effect Diagrams
- Logic Diagrams
- System I/O Lists
- Communications Interface

Definition

- HMI Definition Sheets

Systems Analysis

- Direct Client Conference
- Site Survey
- Scope of Work Assessment

Software Implementation

This phase is obviously dependent on the software engineering discipline involved (see below). However, in all but the simplest of projects, there will be continuous client consultation in order to resolve problems as they arise.

Software Testing

This crucial phase of any software project is always based on a complete and accurate functional specification. In larger more complex systems this may also involve client verification procedures. The software testing is normally carried out in addition to, and separately from, the overall system (including hardware) tests.

Software Support

All our software is fully supported, and due do a policy of software modularity, it generally accommodates future design modifications easily.
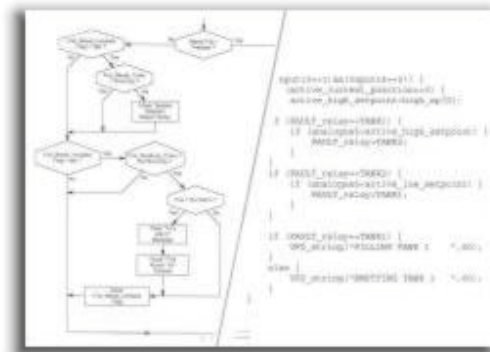
● Fields of Work

The following briefly highlights the type of software engineering disciplines we are involved in. Please refer to our products section for details of the type of application platforms we use.

PLC and HMI Programming

We have considerable experience with several IEC 1131-3 compliant PLC programming environments, and various associated systems (see PLC Systems Integration).

SCADA

We use a variety of applications for the development of SCADA systems, including WinCC and Wonderware.

ANSI 'C'

We use ANSI Standard 'C' to produce application specific software for our embedded controller systems. A long history of development in this area has produced numerous low-level device driver software modules, which we use to speed the customisation of our PCBs to a specific task.

Microsoft Visual Studio .NET

We have previously used Microsoft's Visual Studio 6.0 and now use the next generation of programming tools incorporated in Microsoft Visual Studio 2008. We use either Visual Basic, C# or the C++ programming languages to produce PC based user interaction utilities for communication with our embedded controller PCBs, PLC's or any other applicable hardware.
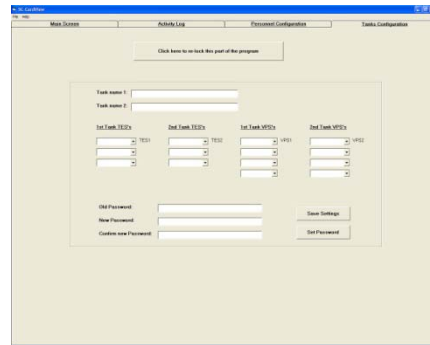
Relational Databases

We have been involved in relational database programming for Management Information Systems (MIS) for a number of years now. The application of this field of software engineering to industrial process monitoring is now common place. The combination of a powerful SCADA implementation and good links to an MIS, allows detailed process information to be accessed in a very efficient manner.