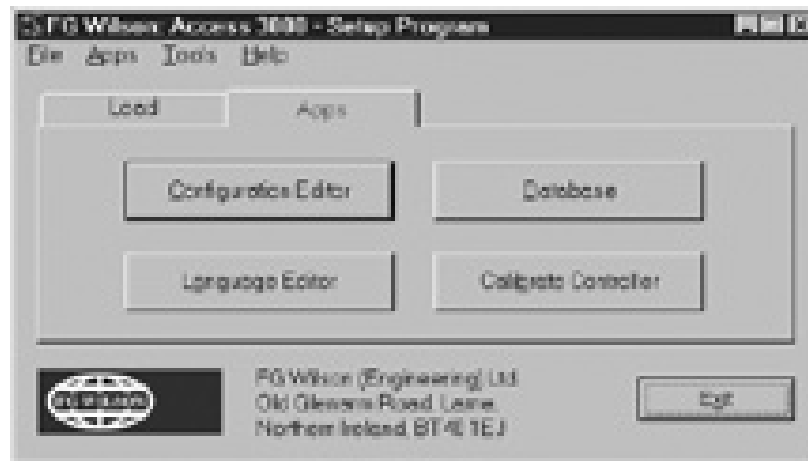


The first and main tab is the "Load" tab this section is simpler to the Load program I designed for Access 4000. I felt that this was a good idea, as my load program is used widely throughout the world, hence when the user sees this program for the first time he/she is already familiar with the operation.

The other Tab "Apps" contains applications, which experience operators can make use off. Screen dump below: -



All the communications will be stored in one dialog and will be called from a number of different locations. Screen shot shown below: -



The program will have direct access to a database, which I will design and program. Program structure must be flexible and readable. I spent some time thinking on the design of the program structure.

This project is a large one and will take some time to complete, the Access 3000 controller is in the early stages of development, hence I will also need to design a test program to test the communications running on laptop connect to PC via RS232.

## Week 42: Monday 21/05/2001 to Friday 25/05/2001

Continue working on my configuration program for the new Access 3000 controller.

I developed a function that could encrypt or decrypt a string up to 500 characters long. It uses a 128-bit key and without this key it is almost impossible to decrypt and encrypted string. This function can be used in any program and will be very us

```
CString CA3KApp::Cryptor(unsigned long *const Pass, CString str, bool Encipher)
```

unsigned long \*const Pass - pointer to 128-bit password

CString str - string to encrypt/decrypt.

bool Encipher - true for encrypt, false for decrypt

Encrypted or decrypted string is returned.

It uses the tiny encryption algorithm; this algorithm is one of the fastest and most efficient cryptographic algorithms in existence. David Wheeler and Roger Needham developed it at the Computer Laboratory of Cambridge University. It is a feistel cipher, which uses operations from mixed (orthogonal) algebraic groups – XORs and additions in this case. It encrypts 64 data bits at a time (8 characters) using a 128-bit key.

It is highly resistant to differential cryptanalysis, and achieves complete diffusion (where a one-bit difference in the plain text will cause approximately 32 bit differences in the ciphertext) after only six rounds (I use 32 rounds). Performance on a modern desktop computer or workstation is very impressive.

For example: -

```
unsigned long pass[4];
pass[0] = 0xFF1287EA; /* 128 bit key - cannot decode with this key */
pass[1] = 0x12345678;
pass[2] = 0x64944953;
pass[3] = 0x37552745;
```

```
CString temp;
temp = Cryptor(pass,"This is a test","",true);
```

```
temp will equal: 0odz0x1819wi8lu3bfln0kln7g4
```

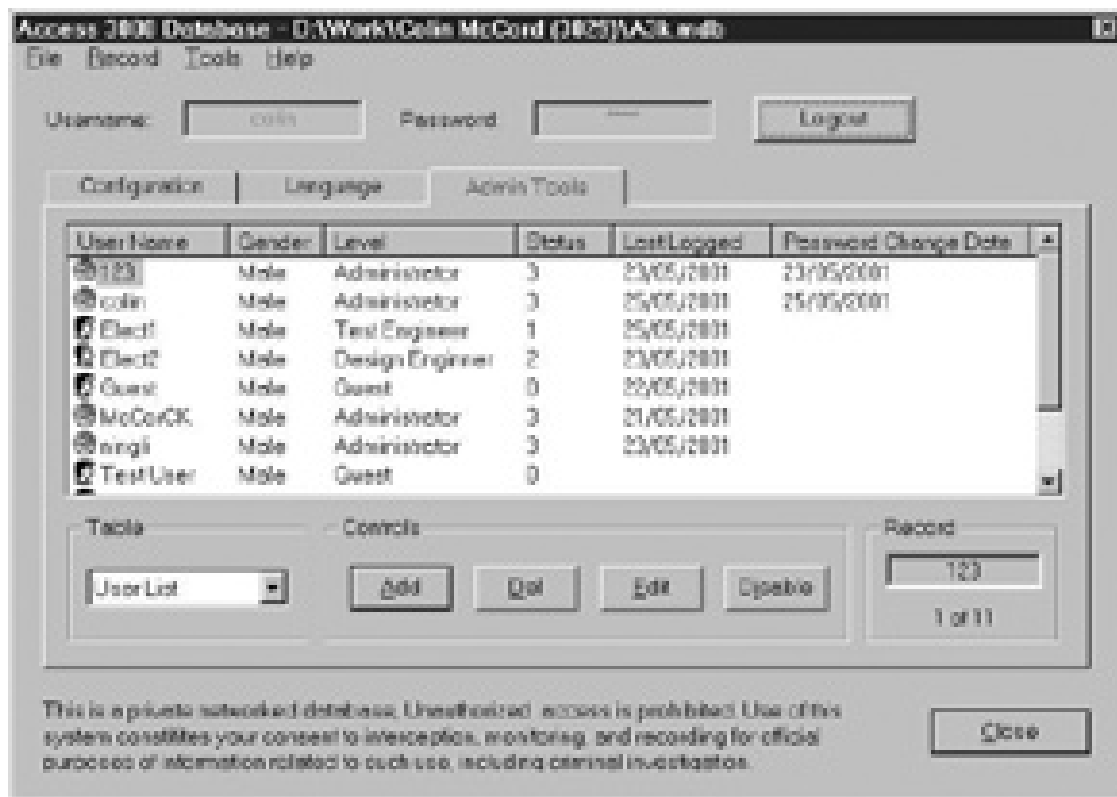
This function will be used to for passwords, serial numbers, etc...and will be used in my programs.

Also worked on the layout and program structure of the configuration program, trying to make it as flexible as possible. Created a view dialogues to view a possible structure. Added three level password control system to the program along with all the code to make it work (using my new encryption function to store passwords in the window registry).





Carried out some work on the database section, creating a dialogue layout and implementing code for user profiles and passwords including adding, editing, and deleting users. Some code was reused from the Access 4000 database program, hence saying time.



Database section will be the last to be completed, but it's important to think about the structure and make sure the program can cope.

Screen dump of Add User Dialogue: -

The 'Add User' dialog box includes the following elements:

- Username:** A text input field.
- Level:** A dropdown menu.
- Gender:** Radio buttons for 'Male' and 'Female'.
- Password:** A text input field.
- Confirm Password:** A text input field.
- Comments:** A multi-line text area.
- Disable User Profile:** A checkbox.
- Buttons:** 'OK' and 'Cancel' buttons at the bottom.

Screen dump of configuration section on database dialogue: -

The 'Access 2000 Database' window features the following configuration options:

- Menu Bar:** File, Record, Tools, Help.
- Login:** Username: colin, Password: [redacted], Logout button.
- Tabs:** Configuration, Language, Admin Tools.
- Current Record:**
  - Dropdown menu.
  - Labels: Please Select Works Order / SAP Number, Date the record was added, User who added the record, Last Date, Last User.
- Communications:** Send from Controller, Write to Controller buttons.
- File:** Import, Export buttons.
- Record:** New Record, Del Record, Editor buttons.
- Move:** Home, Left, Right, End buttons.
- Disclaimer:** This is a private networked database. Unauthorized access is prohibited. Use of this system constitutes your consent to interception, monitoring, and recording for official purposes of information related to such use, including criminal investigation.
- Close:** Close button at the bottom right.

**Week 43: Monday 28/05/2001 to Friday 01/06/2001**

Continue working on my configuration program for the new Access 3000 controller.

Wrote the communication protocol for uploading the main code: -

```
switch(state)
{
    case WaitForResponse:

    case BldPkt: /* read a line from file */

    case SendPkt: /* send out the packet */

    case WtPktAk: /* wait for ACK */

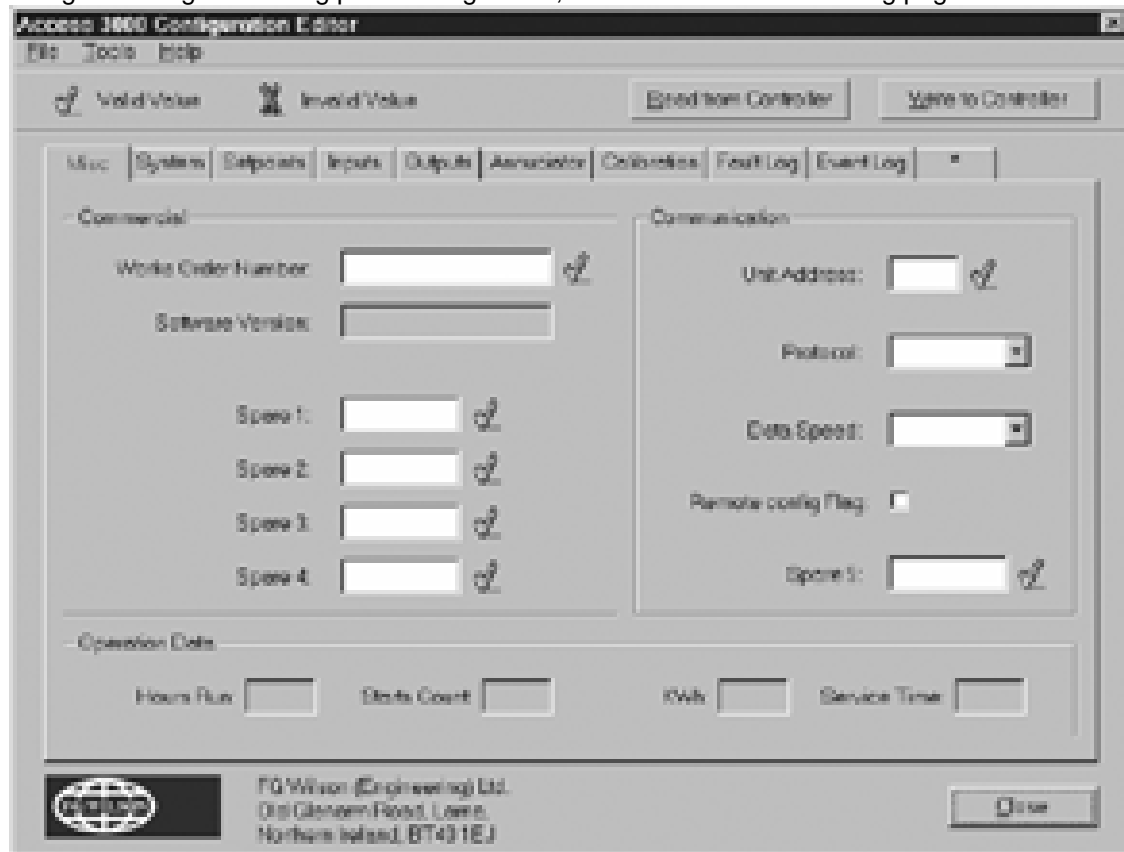
    case PktAkTmo: /* packet timeout */

    case NkPkt: /* receive NAK */

    case ProcEnd

    case Cancel: /* error, abort operation */
}
```

Designed dialogs for editing panel configuration, screens shot in the following pages: -




Little code for has been added yet, I'm just playing around with ideas and trying to design a flexible structure. Screen shot of the 'Misc' tab shown above.

**Access 3000 Configuration Editor**  
File Tools Help

Valid/Value     Invalid/Value       

Misc | System | **Setpoints** | Inputs | Outputs | Annunciator | Calibration | Fault Log | Event Log | \*

Rated kVA:	<input type="text"/>	<input type="button" value="↺"/>	Winding Configuration:	<input type="text"/>
Rated kVA:	<input type="text"/>	<input type="button" value="↺"/>	Number of Phases:	<input type="text"/>
System Voltage:	<input type="text"/>	<input type="button" value="↺"/>	Number of Teeth:	<input type="text"/>
System Frequency:	<input type="text"/>	<input type="button" value="↺"/>	Number of Poles:	<input type="text"/>
PT Ratio A:	<input type="text"/>	<input type="button" value="↺"/>	GT Ratio:	<input type="text"/>
PT Ratio B:	<input type="text"/>	<input type="button" value="↺"/>	Expansion Port:	<input type="text"/>
PT Ratio C:	<input type="text"/>	<input type="button" value="↺"/>	Crank Out Out:	<input type="text"/>
Oil Sender:	<input type="text"/>	<input type="button" value="↺"/>	Crank On Time:	<input type="text"/>
AVR Position:	<input type="text"/>	<input type="button" value="↺"/>	Crank Off Time:	<input type="text"/>
LCD Type:	<input type="text"/>	<input type="button" value="↺"/>	Crank Reports:	<input type="text"/>

 FG Wilson (Engineering) Ltd.  
Oil Glenasm Road, Lanes,  
Northern Ireland, BT43 1EJ   

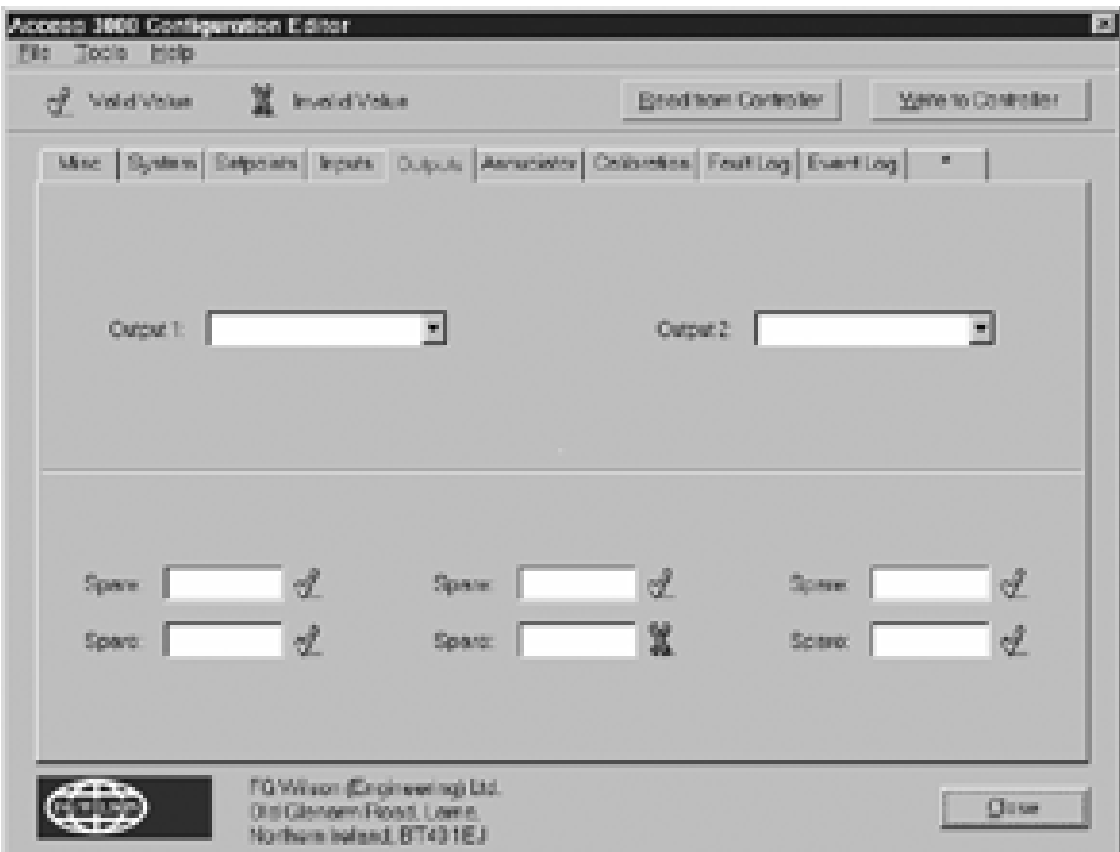
**Access 3000 Configuration Editor**  
File Tools Help

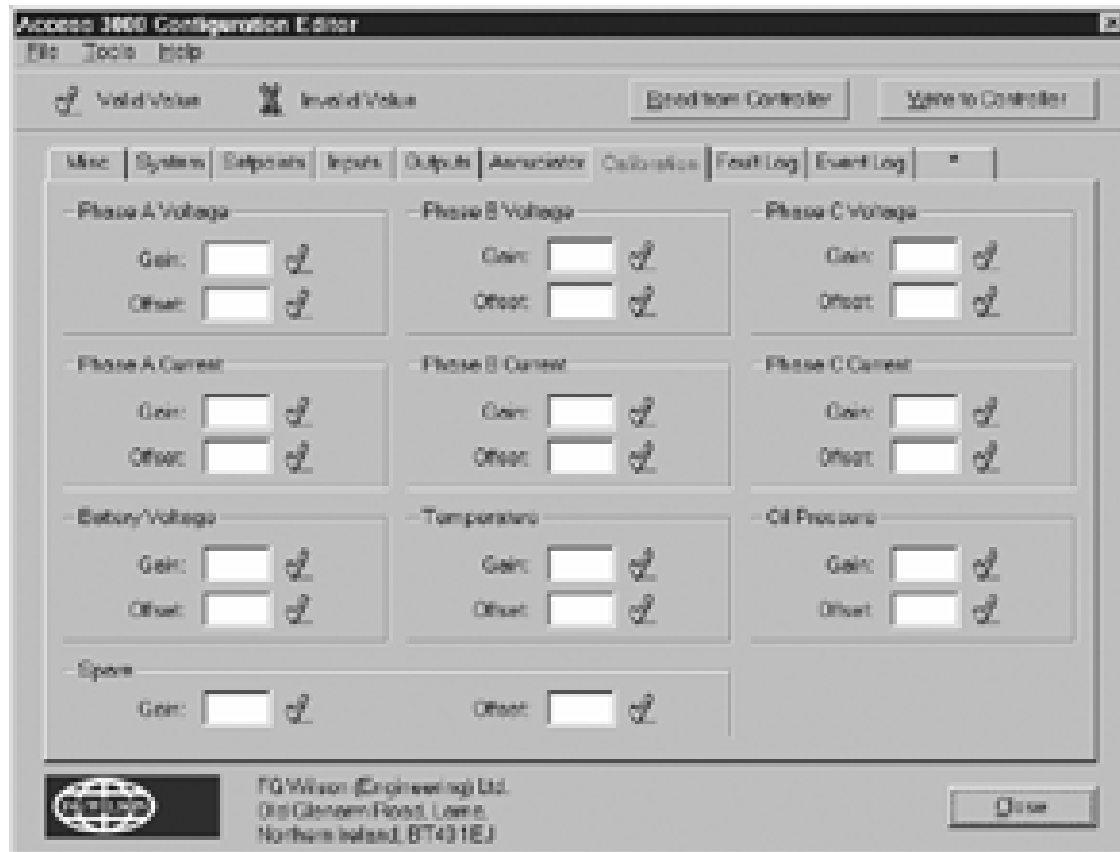
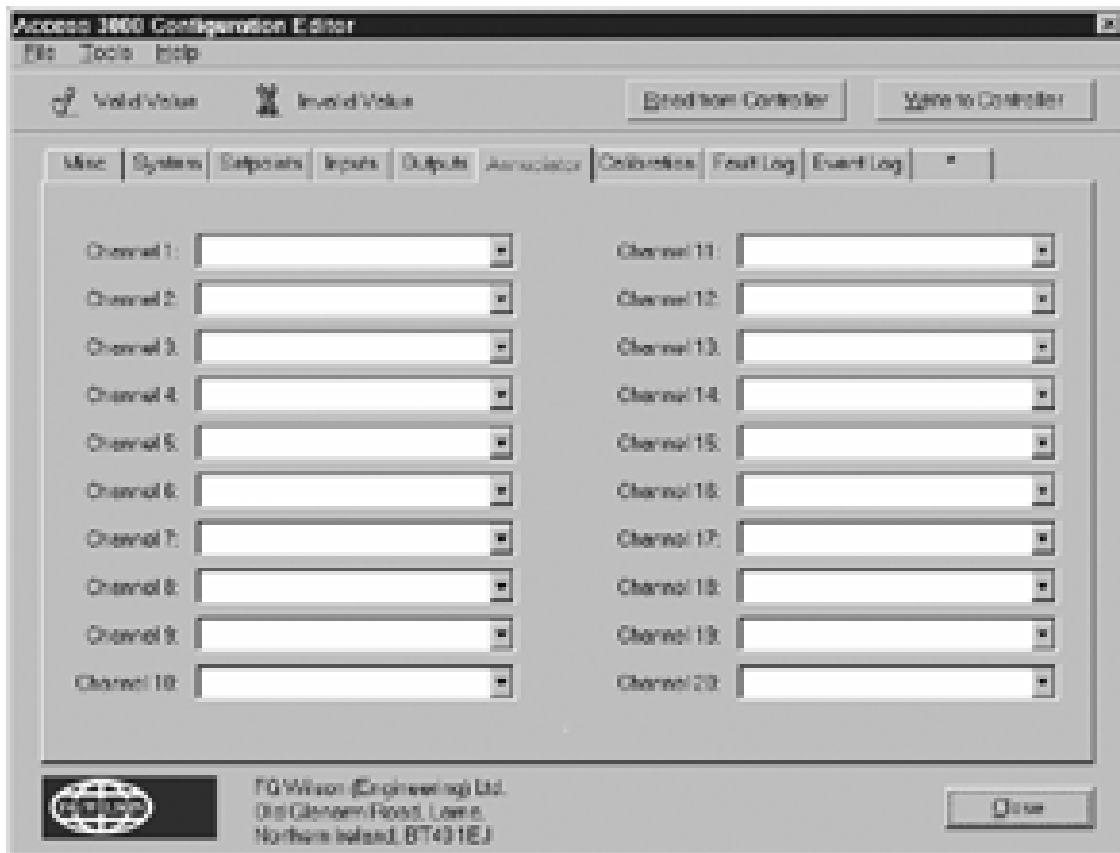
Valid/Value     Invalid/Value       

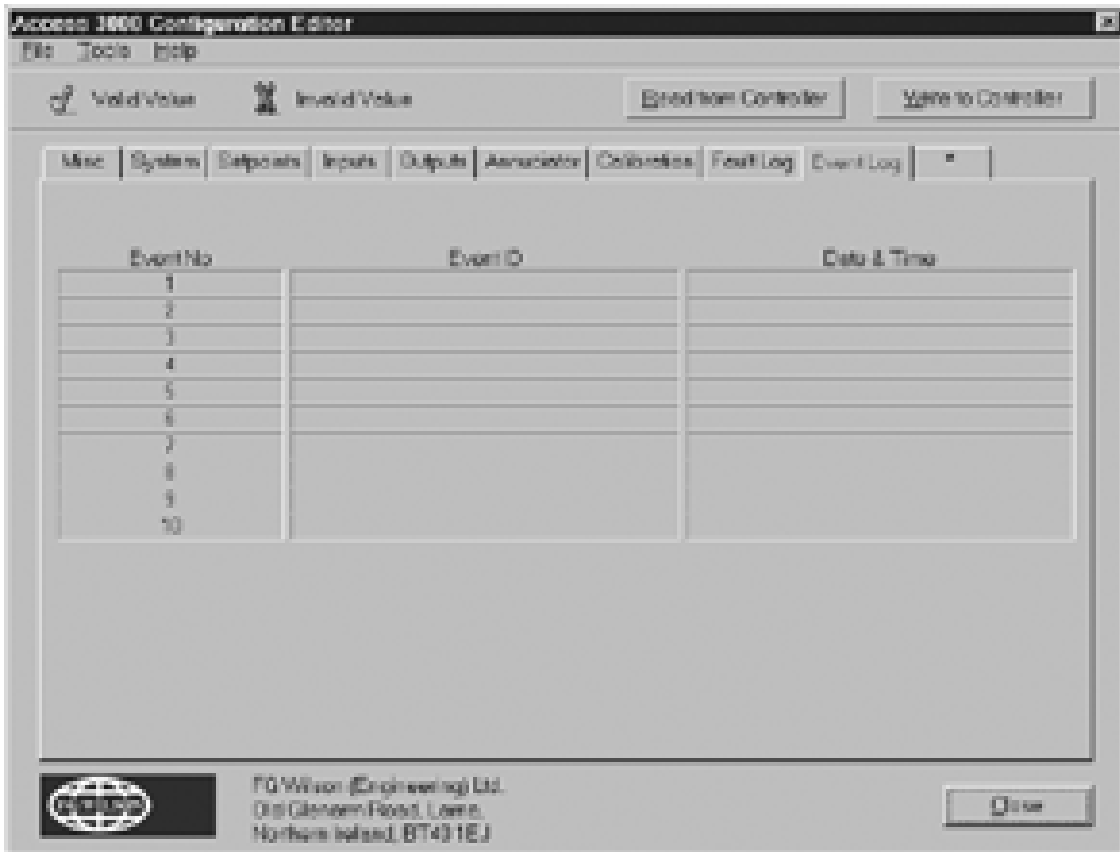
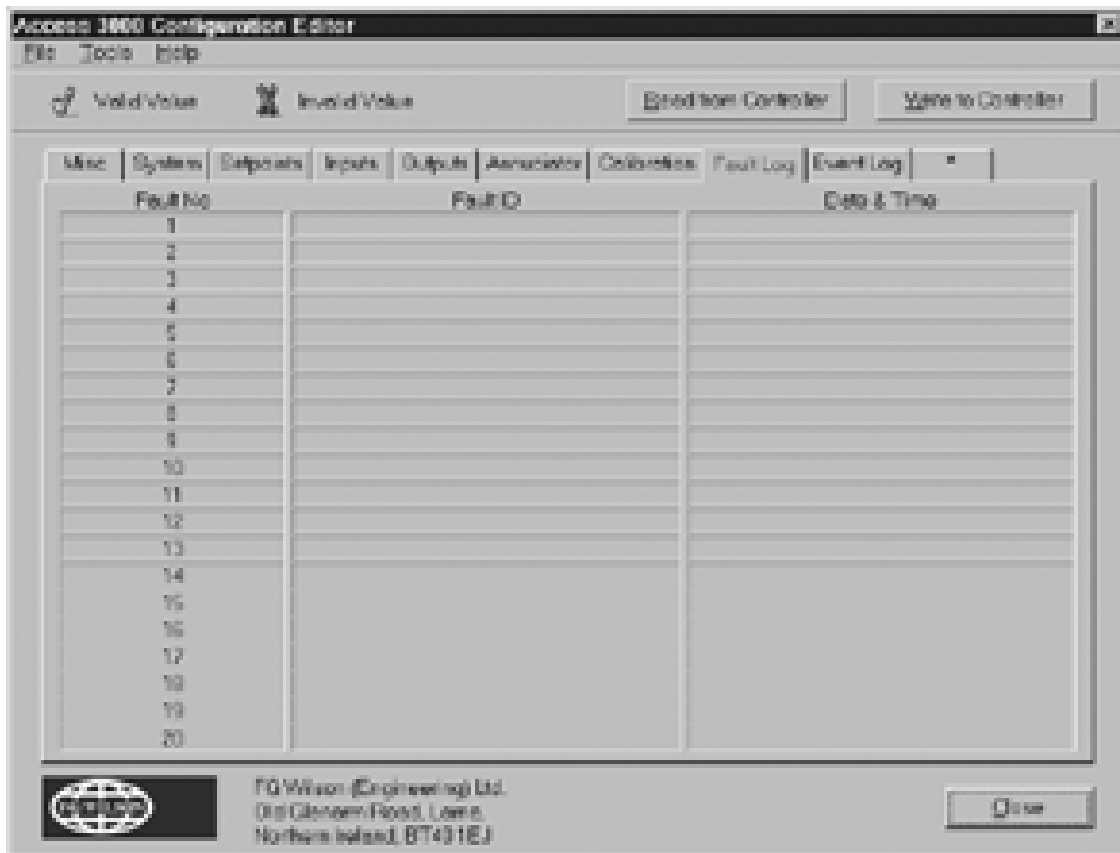
Misc | System | **Setpoints** | Inputs | Outputs | Annunciator | Calibration | Fault Log | Event Log | \*

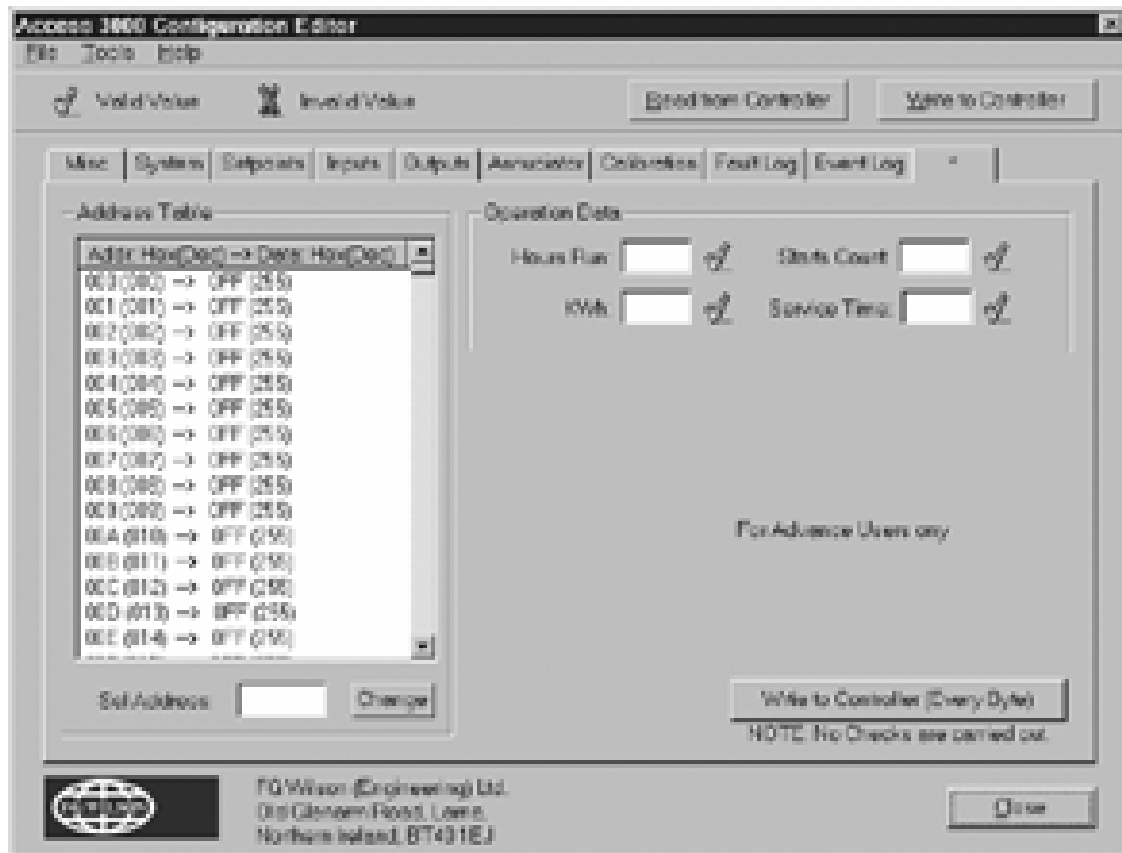
<input type="checkbox"/> Voltage High Setpoint: <input type="text"/> <input type="button" value="↺"/> Delay: <input type="text"/> <input type="button" value="↺"/> Action: <input type="text"/>	<input type="checkbox"/> Voltage Low Setpoint: <input type="text"/> <input type="button" value="↺"/> Delay: <input type="text"/> <input type="button" value="↺"/> Action: <input type="text"/>	<input type="checkbox"/> Freq High Setpoint: <input type="text"/> <input type="button" value="↺"/> Delay: <input type="text"/> <input type="button" value="↺"/> Action: <input type="text"/>	<input type="checkbox"/> Freq Low Setpoint: <input type="text"/> <input type="button" value="↺"/> Delay: <input type="text"/> <input type="button" value="↺"/> Action: <input type="text"/>
<input type="checkbox"/> Bat Volt High Setpoint: <input type="text"/> <input type="button" value="↺"/> Delay: <input type="text"/> <input type="button" value="↺"/> Action: <input type="text"/>	<input type="checkbox"/> Bat Volt Low Setpoint: <input type="text"/> <input type="button" value="↺"/> Delay: <input type="text"/> <input type="button" value="↺"/> Action: <input type="text"/>	<input type="checkbox"/> Bat Charger Fail Setpoint: <input type="text"/> <input type="button" value="↺"/> Delay: <input type="text"/> <input type="button" value="↺"/> Action: <input type="text"/>	<input type="checkbox"/> App Lo Oil Press Setpoint: <input type="text"/> <input type="button" value="↺"/> Delay: <input type="text"/> <input type="button" value="↺"/> Action: <input type="text"/>
<input type="checkbox"/> App Hi Eng Temp Setpoint: <input type="text"/> <input type="button" value="↺"/> Delay: <input type="text"/> <input type="button" value="↺"/> Action: <input type="text"/>	<input type="checkbox"/> Low Eng Temp Setpoint: <input type="text"/> <input type="button" value="↺"/> Delay: <input type="text"/> <input type="button" value="↺"/> Action: <input type="text"/>	<input type="checkbox"/> Overspeed Setpoint: <input type="text"/> <input type="button" value="↺"/> Delay: <input type="text"/> <input type="button" value="↺"/> Action: <input type="text"/>	<input type="checkbox"/> Underspeed Setpoint: <input type="text"/> <input type="button" value="↺"/> Delay: <input type="text"/> <input type="button" value="↺"/> Action: <input type="text"/>

 FG Wilson (Engineering) Ltd.  
Oil Glenasm Road, Lanes,  
Northern Ireland, BT43 1EJ









Started work on an MS dos based test program, which will test the communications of the above program. The reason why I needed a test program is that the Access 3000 is in the early stages of development and it will be at least Christmas before a prototype is operational. The test program needs to have a good program structure, as Dr Ning Li can use it as a guide (reuse sections of code) to write the access 3000 bootstrap code, hence the reason why I used C and not Visual C++ for compatibility and keeping the code simple.

#### **Week 44: Monday 04/06/2001 to Friday 08/06/2001**

Continue working on my configuration program for the new Access 3000 controller.

Design communication protocol for uploading main code to the Access 3000 controller. Designed communication protocol for uploading and download Access 3000 configuration.

Updated program with these new communication protocols. I designed a small dos based test program that will simulate the Access 3000; the reason why I used a dos based test program is that it will allow the communication protocol functions to be directly copied when program on the actual Access 3000 takes place (no advanced C++ functions).

Tested the protocols, with the configuration program running on my PC and my test program running on a laptop with an RS232 connect between them. Any bugs that were found during testing were fixed and re-tested. I even tested both protocols up to 115200 BPS, there were no major problems I successfully managed to get all communication protocols working correctly.