

**Week 34: Monday 19/03/2001 to Friday 23/03/2001**

Populated the final prototype Tel Panel PCB board, getting more soldering experience. After completion of the population of all PCB components, I was involved in the testing of the PCB board this included, testing the communications (RS232 to RS422) using my does CI test program which I design 6 months ago.

After the communication passed, the auto-dialer was tested. Then using GenAccess software and Access 4000 controller, the Tel panel was tested in detail: -

- ◆ Using Modem communications at baud's 9600bps, 19200bps, 32768bps.
- ◆ Direct link RS232 to RS422 at baud's 9600bps, 19200bps, 32768bps.
- ◆ Direct link RS232 to RS485 at baud's 9600bps, 19200bps, 32768bps.

All tests were successful; hence design modification made for VT noise resistance did not affect the operation of the Tel Panel. Also proves that I soldered all components in the correct place, using only the schematic diagram as a guild.

The reset of the week was spent playing around with a view new concepts in visual C++, including custom controls. I designed a sample customs analogue meter for use for my GenModbus application. Created a copy of my GenModbus application and modified the copy, the reason is that visual effects are time consuming and I will properly not have time to complete. This is really just an experiment to gain more knowledge on how to create custom graph oriented controls, screen shot below: -



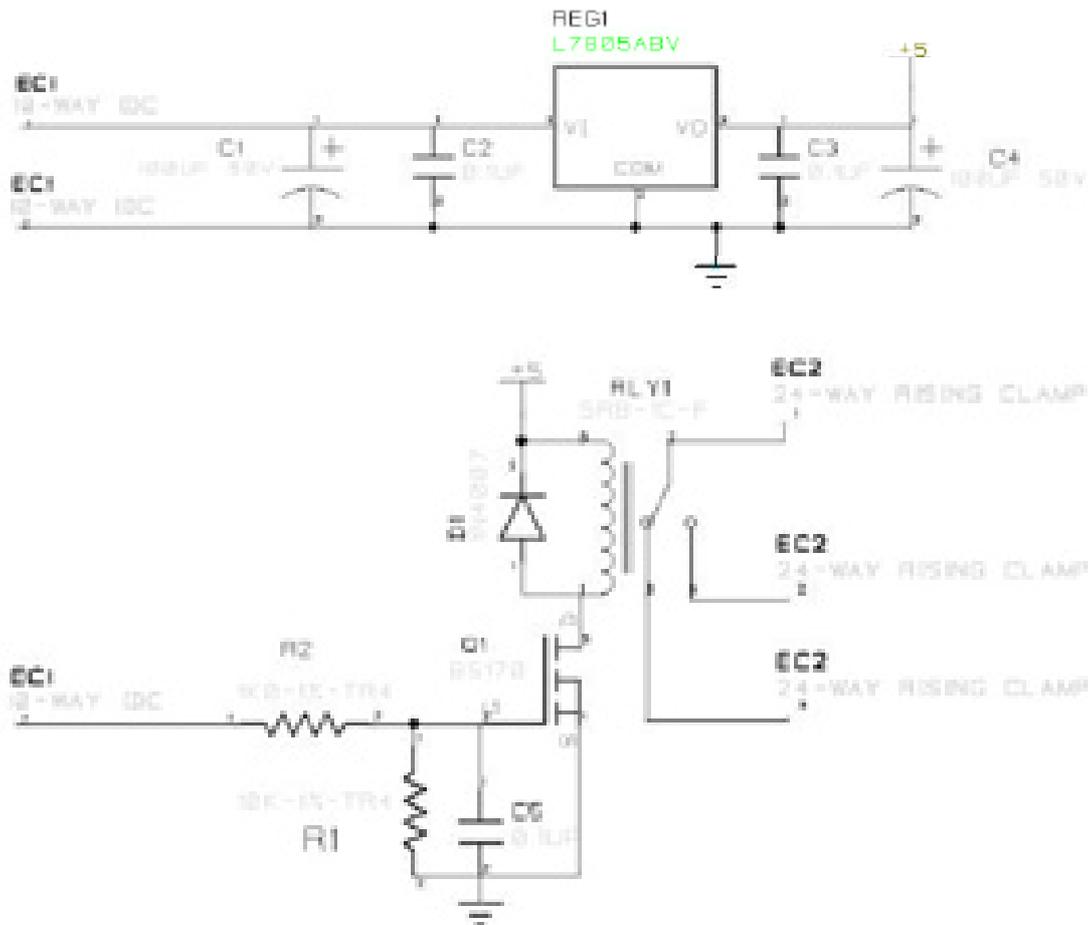
Other Experiments include: -

- ◆ LED displays
- ◆ Resizing of controls
- ◆ Pie charts.

**Week 35: Monday 26/03/2001 to Friday 30/03/2001**

Design of relay board PCB for Access 4000/3000 remote Annunciator, each relay board will have 8 relays. The remote Annunciator has two ribbon cable sockets each can control 8 relays, the relay board must be designed so that the same relay board can be used for faults 1 - 8 and 9 - 16.

5 Volt relay are used, using a 5V regulator (L7805ABV) the voltage operating range of the relay board is 7.5 to 30 Volts. Logic signals from the Annunciator are fed into a transistor (BS170) which in turn switches on the relay: -  
hh



Constructed and tested the design in the electronics lab, the design work well so the circuit diagram was now finalized and design of the PCB can start. There was a small fault discovered in the software of the Annunciator, normally when an alarm is active the appropriate LED flashes on and off. But this is no good for use with the relay board, as the relay will also switch Off and On. The solution was to modify the PIC software to read one of the spare digital switches on the Annunciator PCB. If Switch = 1 for solid LEDs hence solid relay Switch = 0 for flashing LEDs.

Both the schematic and PCB will be design, using CADstar for windows 3.0. Two days were spent learning and getting familiar with this powerful application.

Some time was spend in coming with some why of keeping the relay and annunciator active during cranking, originally a small circuit was design for the relay board with would allow a 9V pp3 battery to power the relay board and Annunciator. But this idea was abandoned because the

current consumed by the each relay is 50mA; there are 8 relays on each board (two board can be used), hence  $8 \times 50\text{mA} = 400\text{mA} \times 2 = 800\text{mA}$ , Annunciator takes 200mA. Hence 1 A is required from the battery, for the worst case scenario. Good quality PP3 battery can deliver 500mA, so it would not be up to task. The Annunciator case is compact and a larger battery would be two large.

It was decided to leave it, if the customer experiences problem with the relays dropping out during cranking they can use a separated power supply to power the Annunciator. This problem will not occur in most cases, it may occur when the Annunciator is a long distance from the Generator (1000s Meters) hence there will be a large voltage drop across the line and the Annunciator will power down (At About 6 Volts). Hence no signal to relays and relay will drop out even though their drop out voltage is 5% of there rated voltage.

Also this week I generated 50 more serial numbers for GenAccess 1.1 software application using my serial generation program I wrote in January. 150 disk labels were then printed using the mail merge technique.

### **Week 36: Monday 02/04/2001 to Friday 06/04/2001**

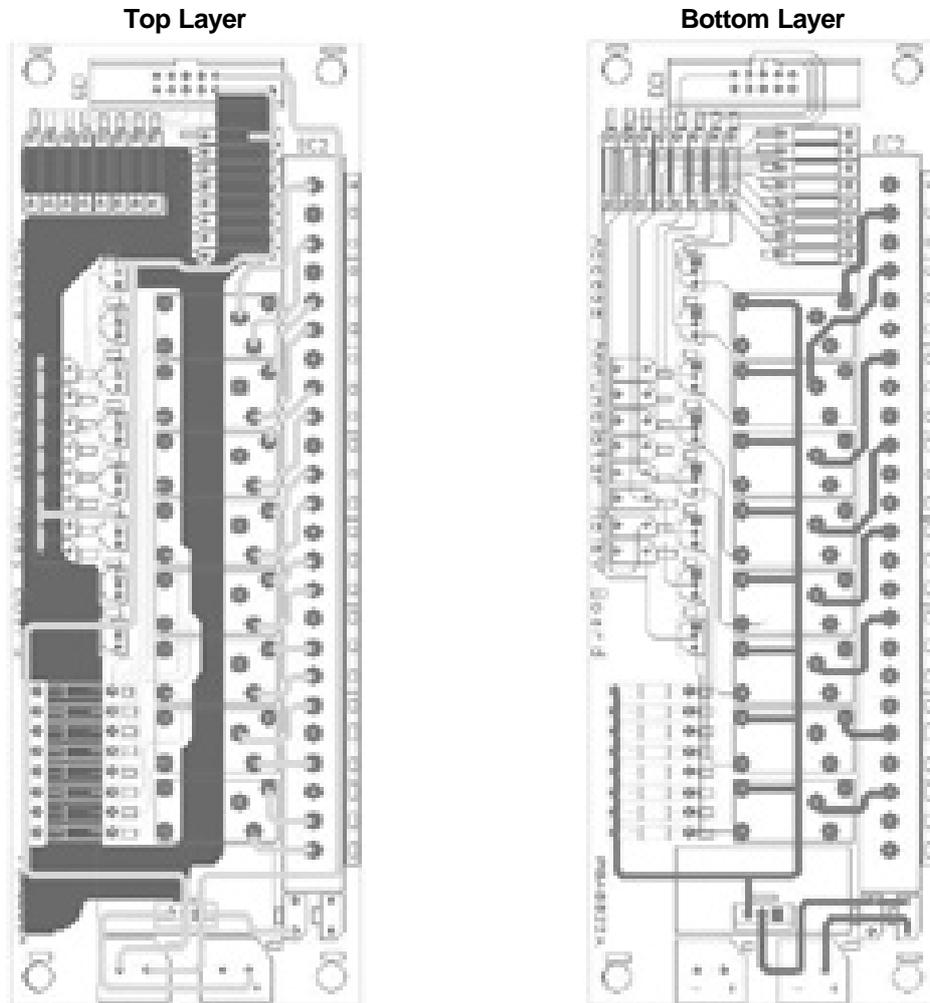
Started design off the PCB, the first thing was to set the size of the board. It was decided that the board will be 165mm by 60mm, and it would have two electrical layers for routes. Then spent some time positioning the components on the board, marking sure that there are no physical problems. Originally I had the 5 V regulator at the top of the board but later move it to the bottom as I did not like it to be close to the ribbon cable incase the ribbon cable started to melt.

After I had the components placed were I wanted them I started to manually route the tracks, using thick 50/25 tracks for GND, VCC, Relay terminals and thin 15/10 tracks for signals. Once all routes were completed using "Design Rules Check" all routes were checked for errors: -

Route to Route	Comp Copper to Route
Pad to Pad	Comp Copper to Pad
Route to Pad	Comp Copper to Via
Pad to Via	Comp Copper to Copper
Route to Via	Comp Copper to Comp Copper
Via to Via	Text to Comp Copper
Copper to Copper	Text to Copper
Neck/Unneck Seg Too Short	Text to Pad
Route to Copper	Text to Via
Pad to Copper	Text to Route
Thin Route Segment	Teardrop to Comp Copper
Via to Copper	Teardrop to Copper
Isolated Copper	Teardrop to Pad
Route Offset	Teardrop to Via
Drill Holes too close	Teardrop to Route
Route to Board Outline	Teardrop to Text
Pad to Board Outline	Teardrop to Teardrop
Via to Board Outline	Text to Text
Copper to Board Outlin	Incomplete Routing
Comp Copper to Board Outline	Text to Board Outline
Teardrop to Board Outline	

Any errors found were fixed and error check compiled again to confirm. Other automatic checks include "Electrical Rules Check", "Connection Check" and "Design Comparison". Once all routes were routed without errors, copper ground plane was added to the top electrical layer. Automatic

check carried again after ground plane was added to make sure it no to close to routes other than GND.



Notice that I've tried to keep the components grouped together, every component in a group of components is the same value.

The spool file used for manufacture of the PCB board will be sent away next well. It will take at least a week for the boards to arrive (5 prototype board will be ordered). When they arrive I will populate them with components and will be involved in the testing producer.

I have gained a new skill, creation of schematic and PCB drawings using CADstar for windows 3.0. A nice break from software, enjoyed every minute.

The following report were generated in CADstar: -

```
-----  
Board Status Report  
  
Cadstar Design Editor Version 3.0  
Design:      D:\Work\COLINM~1\Projects\SCIANN~1\RELAYB~1\RELAY41.pcb  
Design Title:
```

Cadstar PCB Design - 16 Layer Defaults

Date: Friday, April 06, 2001  
 Time: 3:25 PM

	Min	Max	Total
Number of components	59		59
Number of Testpoints			
Area of components (sq.cm)	57.902		57.902
Component/board density	58		58
Number of equivalent 16 pin ICs	11		11
Area per equivalent IC (sq.cm)	9.109		9.109

	Area (sq.cm)	X length (cm)	Y length (cm)
Board outline	100.200	6.000	16.700

	All	Min	Max	Total
Number of pads	177			177
Number of Testpoint pads				

Layer pair codes	Number of vias
(Through Hole)	1
Total	1

	Nets	Mixed width nets	Connections
Total number	51		124

	Unrouted Conns	Unrouted Segments	Stapled Conns	Routed Necked Segments	Routed Unnecked Segments
Widths					
Signal 50/25					161
Signal 15/10					294
VCC 50/25					61
GND 50/25					105
Total					621

Percentage of routed connections to total connections 100

Layer	Component Copper	Copper Teardrops	Unrouted Segments	Route Segments	Route Length (cm)
Top Elec		2		308	132.579
Top Assembly					
Top Glue Spot					
Top Paste					
Top Placement					
Top silk					
Top Solder Resist					
No tracks					
No vias					
Universal ARD					
GND					
Sig 2					
Sig 3					
Sig 4					
Sig 5					
Sig 6					
Sig 7					
Sig 8					
Sig 9					
Sig 10					
Sig 11					
Sig 12					
Sig 13					
VCC					
Bottom Elec				313	161.085
Bottom Assembly					
Bottom Glue Spot					

Bottom Paste  
 Bottom Placement  
 Bottom silk  
 Bottom Solder Resist  
 Doc 1  
 Doc 2  
 Doc 3  
 Doc 4  
 Doc 5  
 Doc 6  
 Doc 7  
 Doc 8  
 Doc 9  
 Doc 10  
 Doc 11  
 Doc 12  
 Doc 13  
 Doc 14  
 Doc 15  
 Doc 16  
 Unassigned  
 Total

2 621 293.664

Layer	Text	Errors	Figures	Areas	Templates
Top Elec					
Top Assembly					
Top Glue Spot					
Top Paste					
Top Placement					
Top silk					
Top Solder Resist					
No tracks					
No vias					
Universal ARD					
GND					
Sig 2					
Sig 3					
Sig 4					
Sig 5					
Sig 6					
Sig 7					
Sig 8					
Sig 9					
Sig 10					
Sig 11					
Sig 12					
Sig 13					
VCC					
Bottom Elec					
Bottom Assembly					
Bottom Glue Spot					
Bottom Paste					
Bottom Placement					
Bottom silk					
Bottom Solder Resist					
Doc 1	26		13		
Doc 2	26		13		
Doc 3	26		13		
Doc 4	26		13		
Doc 5	26		13		
Doc 6	26		13		
Doc 7	26		13		
Doc 8	26		13		
Doc 9	26		13		
Doc 10	26		13		
Doc 11	26		13		
Doc 12	26		13		
Doc 13	26		13		
Doc 14	26		13		
Doc 15	26		13		
Doc 16	26		13		
Unassigned	26		13		
Total	26		13		

	All	Min	Max	Total
Number of test lands				

End of report

**Week 37: Monday 09/04/2001 to Friday 13/04/2001**

Given the task to design a small dos program to allow the test engineers to easily test the communications of every Access Annunciator made. The program needed to be simple to operated; hence there are only 3 items on the main menu.

Screen shot of main menu below: -

```
Access Annunciator Test Program
-----

Start -- hit 1
Help  -- hit 2
Quit  -- hit Q
```

Screen shot of help screen shown below: -

```
Help
-----

Usage: -
1) Connect COM1 to RS232 side of the ADAM converter.
2) Connect Access Annunciator to RS485 side of the ADAM converter
   eg RX+ (Annunciator) to TX+ (ADAM), RX- (Annunciator) to TX- (ADAM).
3) Connect power supply to both ADAM converter and Access Annunciator.
4) Power up the power supply.
5) Run AnnTest.exe from PC.
6) Hit 1 to start testing procedure.

Three tests are carried out: -
1) All LEDs Solid On.
2) All LEDs Quick Flash - Shutdown fault.
3) All LEDs Slow Flash - Alarms.

Program Information: -
Ver:          1.05
By:           Colin K McCord
Date:         09/04/2001
Department:   Electronic Control
Copyright:    (C)2001 FG Wilson (Engineering) Ltd

Hit [Anykey] to return to main menu
```

The communication protocol used in the Access Annunciator is extremely simple when compared to such protocols like Modbus and GenAccess. The Annunciator only receives data it does not transmit; hence only 2 of the 4 wires are used from the RS485 communications.

When the user hits [1] the first of three tests are carried out: -

```
Test 1 - All LEDs Solid On (Switch On)
-----

* Check all LEDs are Solid On

Hit [AnyKey] to continue...
```

If Annunciator communications are working correctly all LEDs should now be on.

```
Test 2 - All LEDs Quick Flash (Shutdown Fault)
-----

* Check all LEDs are flashing Quickly
* If buzzer sounds reset using switch.

Hit [AnyKey] to continue...
```

If Annunciator communications are working correctly all LEDs will be flashing quickly.

```
Test 3 - All LEDs Slow Flash (Alarms)
```

```
-----  
* Check all LEDs are flashing slowly
```

```
Hit [AnyKey] to Finish
```

If Annunciator communications are working correctly all LEDs will be flashing slowly.

When user hits [Anykey] to finish a communication block is sent to switch off all LEDs and main menu appears. The program is design to be small, easy to use, and reliable.

Order components from RS required for 5-prototype relay boards, after Easter I will build and test the relay boards.

Carried out detail costing of Access Annunciator including: -

- ◆ Cost of PCB Components
- ◆ Cost of PCB
- ◆ Cost of populating PCB with components
- ◆ Cost of case
- ◆ Cost of Screws, bolts, nuts and washers.
- ◆ Cost of optional relay board, including; components, PCB, Bolts, nuts, washers, and 10-way ribbon cable.

### **Week 38: Monday 23/04/2001 to Friday 27/04/2001**

5 PCB relay boards arrived, along with the components I ordered from RS. Checked every relay board PCB with multimeter, making sure there were no broken tracks or design errors. Populated all 5 relay boards with components, getting some soldering practice.

After two relay board prototypes had been constructed, they were connected to a prototype Access annunciator so operational tests could be carried out. Each relay was turned on one by one using my Access annunciator test program, and normally closed and normally open pins were checked using a multimeter in buzzer mode.

The relay functioned properly, and the other three boards were completed. Next was reliably testing, this involved leaving all relays on and checking that they where still work after a long period of time.

Using a hand drill I drilled 8 holes on the back of the Access annunciator box, these holes were for mounting of two relay boards. Two relay boards were screwed into place and connected to the Access annunciator via 10-way ribbon cables. The Access annunciator, which is mounted on the lid of the box, was connected to a power supply with communications connect to my PC via RS232 to RS485 converter.

The power was switched on and using my Access annunciator test program all 16 relays were switched on. Power supply voltage was then increased to 28 volts (worst case condition for the 5-Volt voltage regulator 7805).

After some time the voltage regulators and relays had become very hot, but seemed to be stable as the temperature didn't get any hotter as time went on (Heat sink working well). After 8 hours the annunciator and relay boards (still hot) will still operating correctly. This test was carried out

twice and will properly be left running all of next week during office hours, making sure the design is reliable.

### **Week 39: Monday 30/04/2001 to Friday 04/05/2001**

Carried out detailed functional tests on Access Annunciator relay board I designed. These tests included reliably testing, which involved turning on all relays and left running for long periods of time.

The Access Annunciator (with two relay boards fitted) were tested beyond their normal operating conditions. When connected to access 4000 controller it is unlikely that more than 5 faults at anyone time would occur, I carried out reliability tests with all 20 faults active, hence all 16 relays were active.

It's extremely difficult to get all 20 faults to occur using the Access 4000 controller, hence the Access Annunciator was connected to my PC using RS232 to RS485 converter. My Access Annunciator test software (Advanced version) I developed a couple of months ago, was used to active all alarms.

Voltage was set at 29 volts, and the Annunciator with all 16 relays active, was left to run constantly for up to 8 hours. This was repeated several times, afterwards the relay boards and Annunciator were still working as normal. But it did get extremely hot, hence temperature reading where taken; result table shown below: -

Voltage	Relay B1	Relay B2	Current	Hours	Room Temp	Temp. Reading
29.40	All 8 ON	All 8 ON	580 mA	8	26.5°C	63.6°C
29.40	4 ON	4 ON	410 mA	4	26.3°C	51.4°C
29.40	4 ON	All 8 OFF	280 mA	2	26.3°C	39.2°C

The results above were taken with no air holes on plastic box, air holes were added and the following results were obtained: -

Voltage	Relay B1	Relay B2	Current	Hours	Room Temp	Temp. Reading
29.40	All 8 ON	All 8 ON	580 mA	8	26.8°C	53.0°C
29.40	4 ON	4 ON	410 mA	4	26.3°C	39.4°C
29.40	4 ON	All 8 OFF	280 mA	2	26.3°C	32.2°C

The relay boards were fully functional after all of the reliably tests.

Order PCB components from RS for Access Annunciator PCB, the final prototype PCB board is due next week. I will build and test the new board sometime next week.

### **Week 40: Monday 07/05/2001 to Friday 11/05/2001**

Populated and tested the final prototype PCB board for the Access Annunciator, there were no problems and Access Annunciator is due to be released soon.

Started working on the design of software for the new Access 3000 controller (Access 3000 configuration).

**Requirements: -**

Configuration is conducted during bootstrap. There are four main functions: -

- ◆ Export main code to flash memory.
- ◆ Export configuration file to serial EEPROM.
- ◆ Import configuration to PC.
- ◆ Export TEXT file to flash memory.

On the PC side each function is performed using batch mode communication, i.e. each function can be started by selecting Export Main Code, Export Configuration File, Import Configuration file etc.

During bootstrap if 'M' or 'C' or 'U' or 'T' is received 5 times, the software goes to one of the defined functions. When the function is completed, the software enters Configuration State that allows the user to perform another function.

The configuration software should be well structured for future modification. Since remote configuration using modem will be included, the corresponding program block should be inserted as a dummy function.

Calibration program and database function may be required to be included in the future, so the software structure should be flexible to accommodate these applications.

Spend most of the week thinking about the program and how to make the operation of the program as easy as possible (Easy to use user interface, hence no training required for users). Jotted down a couple of rough idea on paper, spending a lot of time thinking about the program structure and flexibility.

### **Week 41: Monday 14/05/2001 to Friday 18/05/2001**

Continued working on the design of the configuration program for the new Access 3000 controller. At this early stage of development I concentrated on program structure and layout. All the communications will be stored in the same dialog class, and can be inserted directly into any program.

I played around with dialogs trying to design and simple easy to use program. Screen dump of the man dialog shown below (Note: at this early stage dialog may change): -

