An environmental enclosure for an inverted time-lapse microscope

1. Introduction

There are undoubtedly instances when it I required to modify an existing microscopy facility to include time-lapse imaging. The technology which provides automated, time-sequential imaging is relatively mature and numerous software tools are available to provide such repetitive imaging. However, in biological sciences in particular, which deal with observations of changes in cell morphology, cell motility, cell signalling processes etc., or with cell division studies in response to different cell insults or other modifying agents, it is essential to maintain an appropriate environment for cell growth. In short, we need to maintain cell viability. This note describes an approach for constructing a microscope enclosure from readily available components and at relatively low cost, with the aim of maintaining the cell environment at the usual 37^{o}C and 5% CO_2 atmosphere. These conditions are controlled and can be set to any desired value, although the controlled internal enclosure temperature should be in the range of 30^{o}C -45^{o}C for best operational performance.

There are of course a number of commercially available solutions to this ‘problem’. These in general are expensive (costing £7k-£15k, depending on features installed) and need to be tailored to a specific microscope body. Most often such solutions enclose only part of the microscope and thermal gradients across the microscope body are unavoidable. Moreover, such systems are associated with thermal gradients within the enclosure.

We have taken a different approach: we enclose the complete microscope. Although this makes access to various microscope components rather more difficult than the classical partial enclosure, this is not a significant problem, since by the very nature of time-lapse work, constant imaging/illumination conditions are used. The enclosure is designed to house an inverted microscope, the geometry normally used for time-lapse imaging of cells in a growth medium.

2. The enclosure

The environmental enclosure described in this note is shown in Figure 1. It consists of a frame constructed from 30 x 30 mm lengths of extruded aluminium struts with most of the panels made from two layers of black 4 mm thick Correx™ sheet. This material is a reasonably efficient insulator since it traps air within it corrugations; a cross-section is shown in Figure 2. The two corrugated polypropylene sheet layers are arranged with their ‘cells’ at right angles to each other; this increases material strength to deflection and lowers thermal conductance across the walls. The base and the top of the enclosure are made from dense 18 mm laminated composite material (similar to a kitchen worktop or shelving material). Finally two doors are fitted, again made up from 30 x 30 mm aluminium extrusions, but this time they are filled with 6 mm thick semi-transparent, grey acrylic sheets. The doors allow access to the enclosure and some degree of visibility of the inside of the enclosure. Although the thermal conductance of acrylic sheet (Perspex™) is far from ideal, thermal losses through the doors is adequately low. Most of the thermal leakage is through the aluminium sections, but in practice, only a few tens of watts are lost out of the complete enclosure when the internal temperature is 37^{o}C and the outside is at the normal laboratory temperature of 20^{o}C or so.

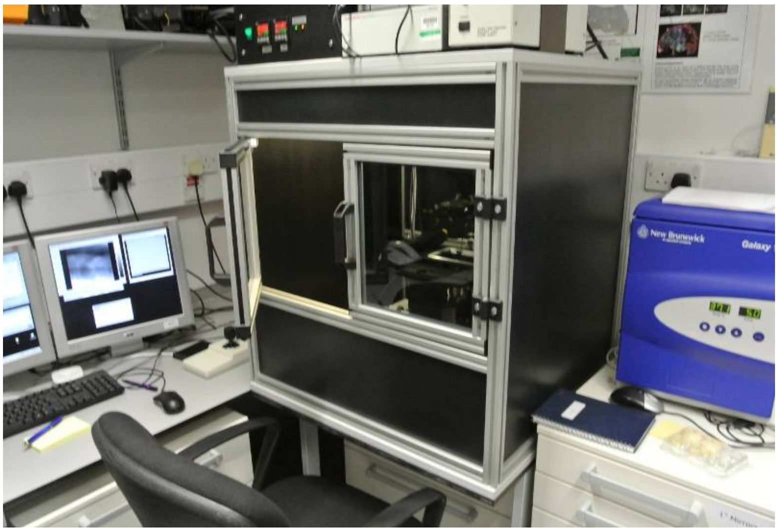


Figure 1: The completed time-lapse microscope enclosure.

Details of the aluminium struts and the insulation are shown in Figure 2. It just happens to be convenient that the 8 mm ‘gap’ in the aluminium strut is just right for two layers of the Correx™ material. In the case of the thinner door panels, the resulting gap is filled with a grey silicone rubber seal, as outlined later in section 6.

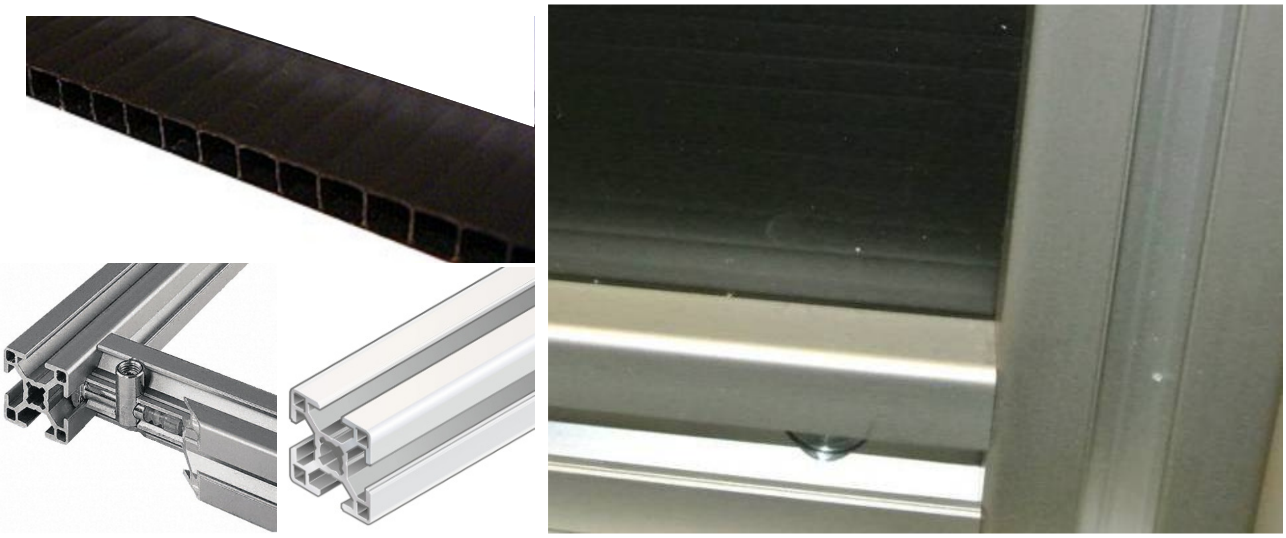


Figure 2: Close up views of the corrugated polypropylene sheet, the 30 x 30 mm strut, one of the enclosure corner joints and one corner of the completed enclosure.

The dimensions of the enclosure are shown in Figure 3. The two enclosure doors are made to be as small as practicable to minimise the loss of CO_2 (and heat) while allowing adequate access and allowing manual use of the microscope system with eyepieces, i.e. without the incubator control system energised.

The enclosure electronic control system is placed on top of the enclosure and is an integral part of it. The various sensor connections are taken through the bottom of the control system chassis and wired directly to the relevant components

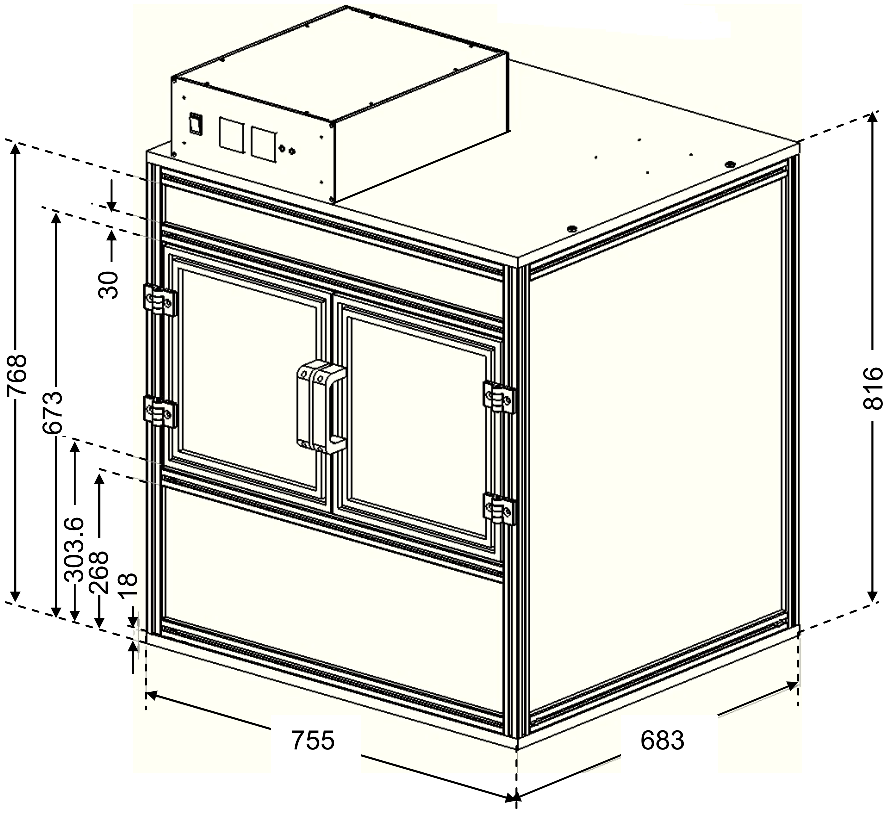


Figure 3: Outline of the environmental enclosure, with the control system on top of the enclosure.

SolidWorks* models of the enclosure are shown in the next figure, Figure 4. Detailed drawings can be obtained on request. It is noted that while all the components used in the construction of the frame can be obtained from individual suppliers, it is significantly more cost-

*http://www.solidsolutions.co.uk

effective to obtain the complete enclosure from a specialised aluminium profile supplier; in our case this was KJN Automation Ltd*, who supplied all the mechanical components for ~£300.

One of the issues with all microscope enclosures is what to do with the 'scope illumination light sources. These can produce significant amounts of heat (e.g. when a Hg arc lamp is used for fluorescence excitation). The TE300 microscope is convenient in the sense that there is a clean separation of the fluorescence excitation lamp, in line with the filament trans-illumination lamp. However, the filament lamp mounting is a little awkward, requiring a somewhat complex cutout shape in the rear panels. It is much simpler to remove the conventional Nikon lamphouse and replace it with an LED illuminator. This is just what we did and this illuminator is described in a separate note. The Nikon lamphouse could be fitted, but we just did not have the patience to figure out the correct cutout shape! Doubtless other microscope bodies will be slightly different, but it is probable that they could be accommodated in a similar manner.

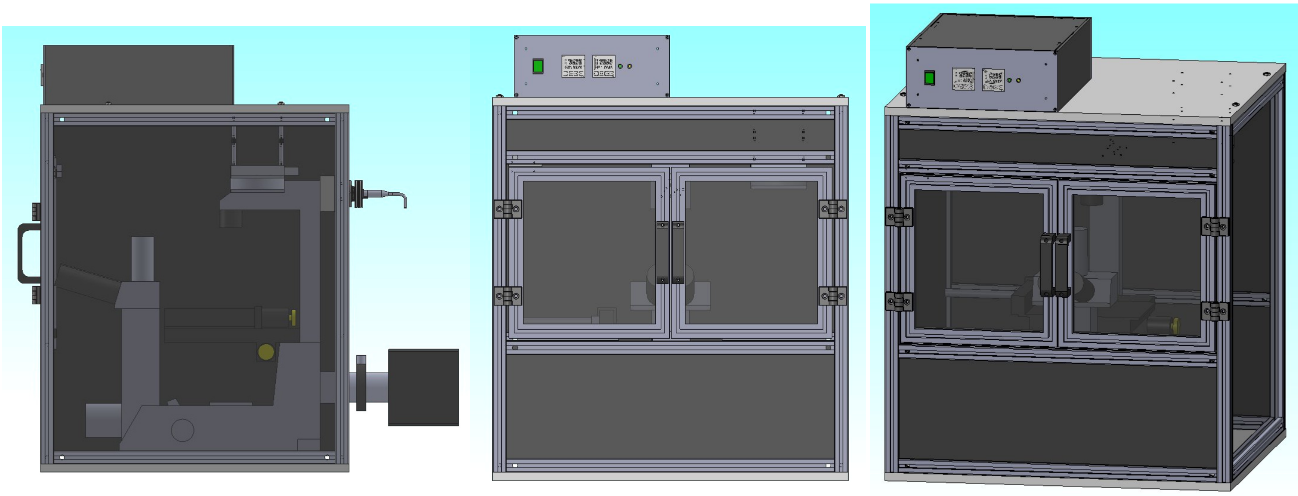


Figure 4: Side, front and rear views of the environmental enclosure with the control electronics. The TE300 microscope chassis is also outlined here, showing that easy access to the eyepieces is possible. The enclosure rear wall is constructed from two sections; the lower section can be modified to allow a range of access holes for cables etc.

3. CO_2, humidity and temperature sensing and heating assembly

It is of course critical to select appropriately reliable and accurate transducers for sensing the enclosure temperature and CO_2 concentration. We have based our system around two convenient and low cost devices: A platinum resistance foil sensor and an infrared CO_2 sensor module. The latter can be obtained from Gas Sensing Solutions Ltd and is particularly convenient because it provides two outputs: a digital serial data stream which carries CO_2 concentration, humidity and temperature, the later derived from a bandgap semiconductor sensor. The second output is an analogue voltage proportional to percentage CO_2, this can be easily used with a standard process controller. Humidity is sensed by a capacitive sensor. The digital (UART) streams are converted to a USB data link using a standard interface module. This provides us with a computer display of the enclosure environment, as described later. It is noted that it is important to specify, at time of ordering, that the analogue output is enabled on the CO_2 sensor. It is not obvious to us why the analogue output is not always enabled, and we fell into that trap; ours is not to reason why a special request is necessary. Other than this small difficulty, the sensor is excellent; we used a model covering the 0-20% range. In the perfect world, a 0-10% model would exist!

* http://www.kjnltd.co.uk
** http://www.gassensing.co.uk/
*** http://www.labfacility.co.uk/sensing-resistors-inserts-platinum.html

The platinum foil sensor (Labfacility DM-333***) is used with a second process controller which controls power to a heater. These devices are shown in Figure 5.

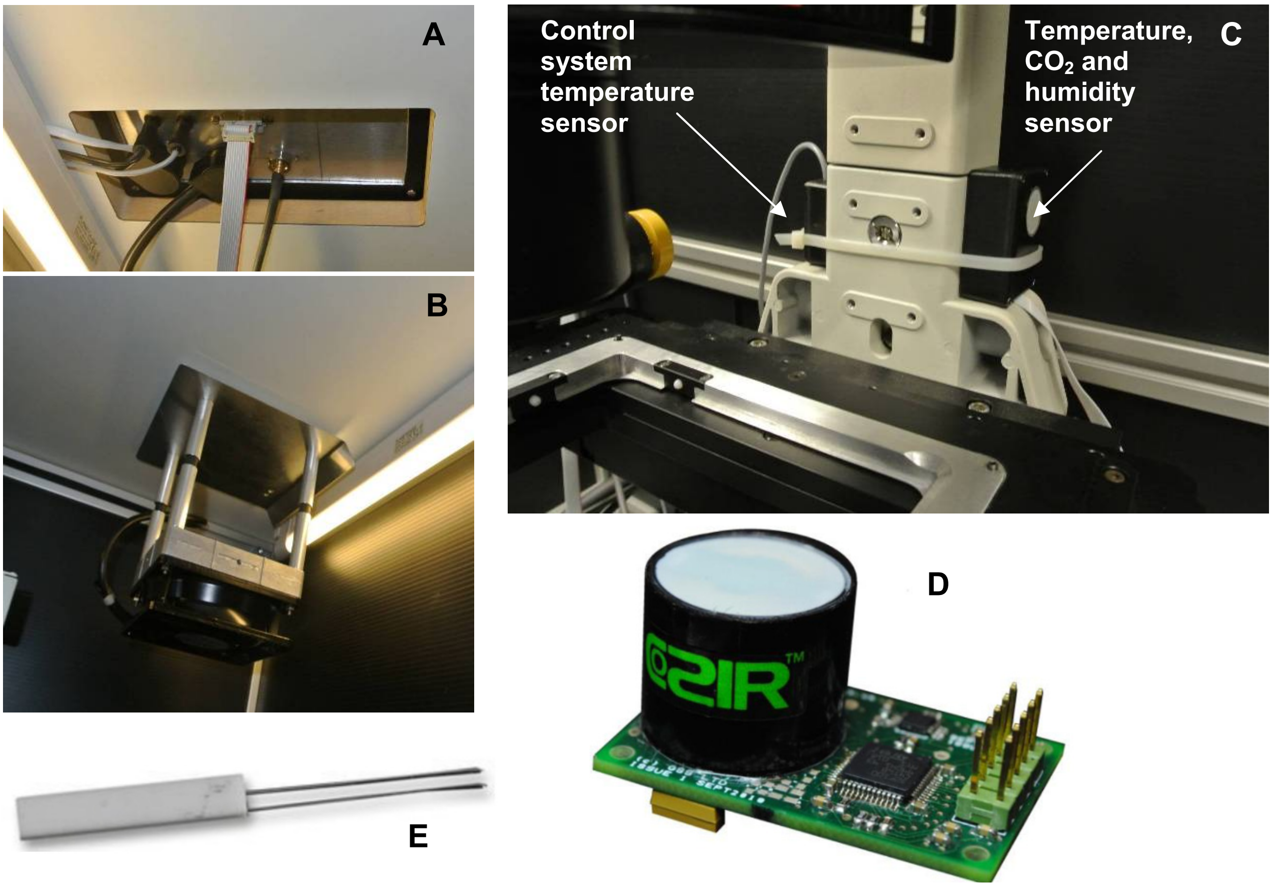


Figure 5: Views of key enclosure sensors and heater/fan assembly. A: Detail of the connections through the roof of the enclosure. B: The heater and fan assembly, again mounted on the enclosure roof on four pillars through anti-vibration mounts, on the rear right side, next to one of the enclosure LED strip illuminators; an aluminium heat shield is also shown. D: The combined CO_2, humidity and temperature sensor and E: The Pt100 temperature sensor used by the control system.

In our laboratory we are fortunate to have access to a piped CO_2 supply and this was used in this project, although of course a standard gas cylinder could also be used. The gas is passed through a solenoid-operated valve and then fed to a small bubbling tower placed inside the enclosure. This humidified the gas and, although the humidity is not independently regulated, it was found that this simple process is quite adequate.

The sensors are enclosed in small plastic boxes, strapped to the rear pillar of the microscope, as shown in Figure 5(C). The Pt100 foil sensor is epoxy-glued to a small aluminium strip, exposed to the environment through a hole in box. The CO_2 senor is similarly exposed, although the temperature and humidity sensors are not. The power consumption of the sensor module is so low (3.5 mW) that it hardly affects local temperature.

Enclosure hating is provided by a fan assisted 400 W heater, as shown in Figure 5(B). Although nowhere near 400 W is required to maintain the enclosure at 37^{o}C, it is desirable to have the high power available when the enclosure is used from cold. The enclosure temperature takes about one hour to settle, although it is close to working temperature within 30 mins or so. The heater is a device manufactured by Stego*and conveniently mount directly to a standard 119 mm fan. Although somewhat expensive, it does include the crucial over-temperature thermostat cutout. This

http://www.stego.de/

is particularly important when such a high power heater is used. The fan re-circulates the enclosure atmosphere 'upwards', forcing it towards the enclosure roof and redistributing it through the enclosure. The addition of baffles would doubtlessly improve performance, but we have found the arrangement depicted in Figure 5(B) quite adequate. The heater/fan combo is mounted through vibration isolation rubber mounts from the top of the enclosure. Without such isolators, the enclosure top acts as a soundboard and produced vibration not conducive to microscopy!

Finally, we light the enclosure using two LED light strips, mounted near the left and right sides of the enclosure top. These lights are switched off with a door-operated switch, much like in a domestic refrigerator. This is if course essential for time-lapse fluorescence work. The lighting circuit is always energised, allowing the enclosure to be used when the environmental control system is switched off.

All these devices are connected to the enclosure electronic control unit, described next.

4. Temperature and CO_2 control

The enclosure controller is shown in Figure 6. It is constructed in a 333 deep '3U' 19" rack box, used sideways, i.e. with the usual 'sides' used as the front and back. Almost any other similar enclosure can be used; the one used here is convenient, as the top can be easily removed for access. The key components are two process controllers used to control temperature and gas concentration. There are probably simpler approaches to perform feedback control, but the adaptive proportional-integral-derivative (PID) control loops in these units are excellent and involve no design work! However, it is essential to have a good magnifying glass available when reading the manuals: the fonts used are so small that both the younger and older authors could not read them without assistance!

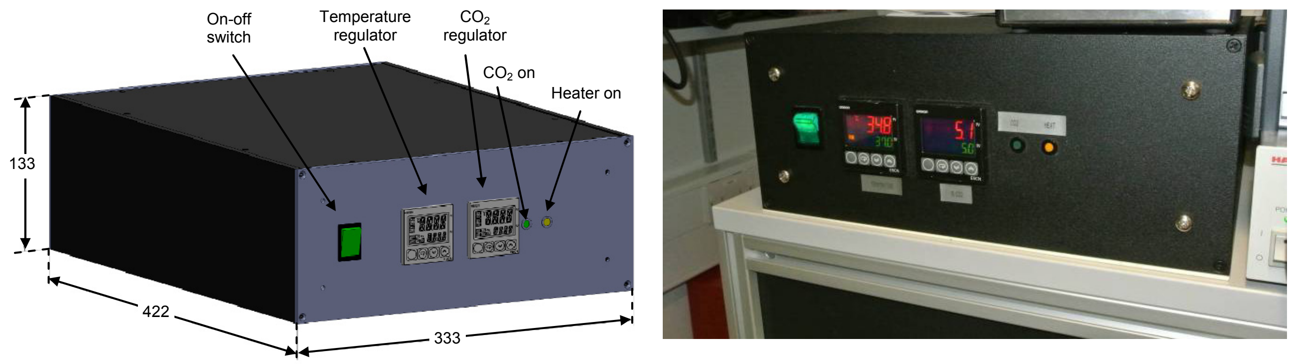


Figure 6: The enclosure control system. A SolidWorks model is shown on the left, and the real thing on the right.

The front panels of the controllers can be configured to make easy set-point changes. Although the process controllers appear identical, they are different: one of them provides a sensor input for the Pt100 temperature sensor, while the other accepts a sensor voltage.

The unit contains a number of DIN-rail mounted components. These include a solid-state relay which turns on and off the heater power, a door switch relay, a 24V dc power supply and a time-delay relay. This time-delay relay is used to override the process controller output when the doors are left open for more than a few minutes: this prevents undue usage of CO_2 when the doors are accidentally left open. All internal connections are made on DIN rail terminals. External connections, into the enclosure, are made through a socket panel, shown in Figure 5(A); this includes the gas outlet bulkhead fitting. Connections to the outside world are made through the rear of the unit: these are the ac mains input, the gas inlet and the USB output. Fuses are also placed on the rear of the enclosure.

The system circuit diagram is shown in Figure 7 and the internal component layout in Figure 8.

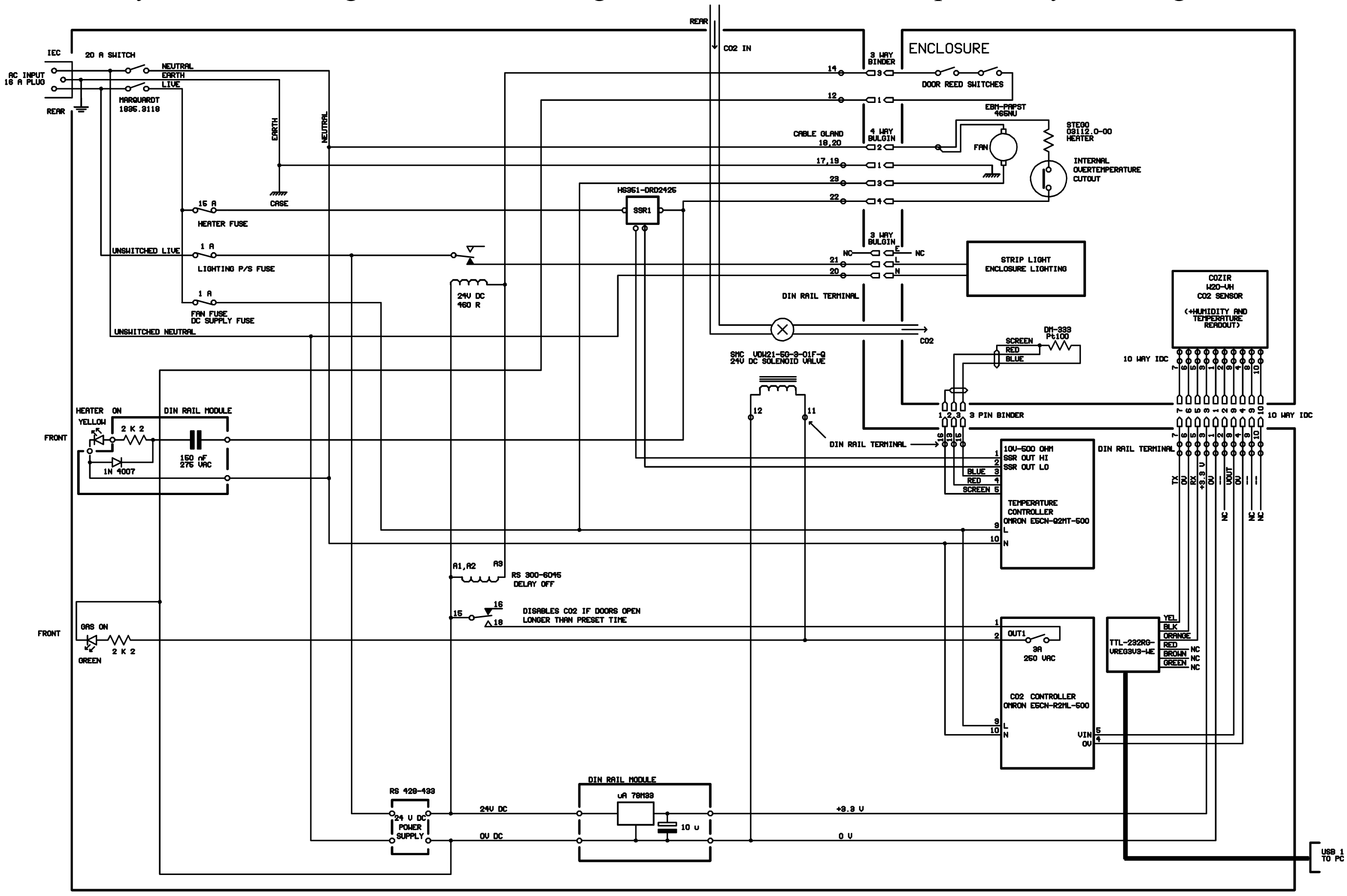


Figure 7: Circuit diagram of the enclosure control system. Connections inside the enclosure are shown in the top right. The 'heater on' LED indicator senses heater ac mains through a capacitor/diode dropper. A 3.3 V regulator supplies dc power to the combo sensor as this needs to be always energised, even when not USB-connected. All subassemblies are DIN-rail mounted .

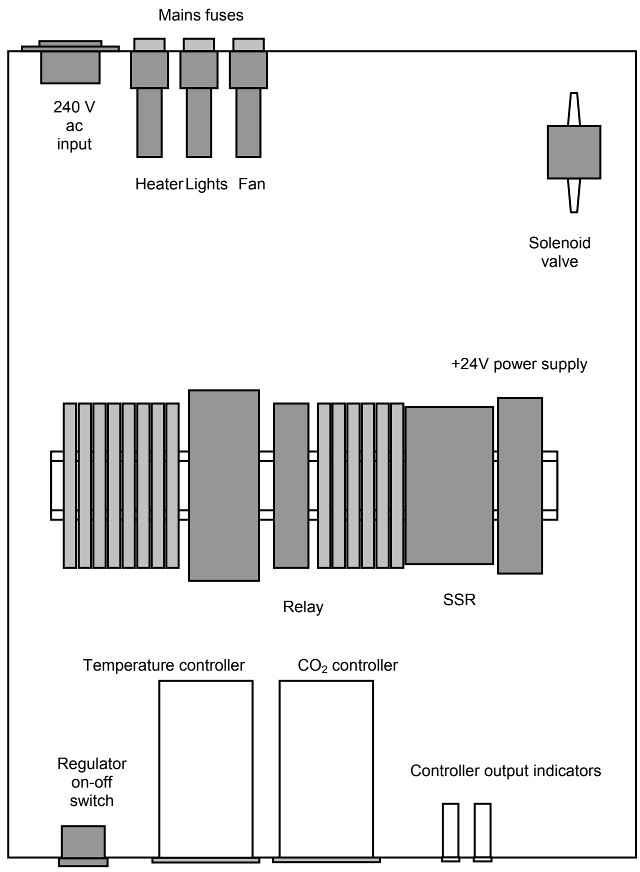
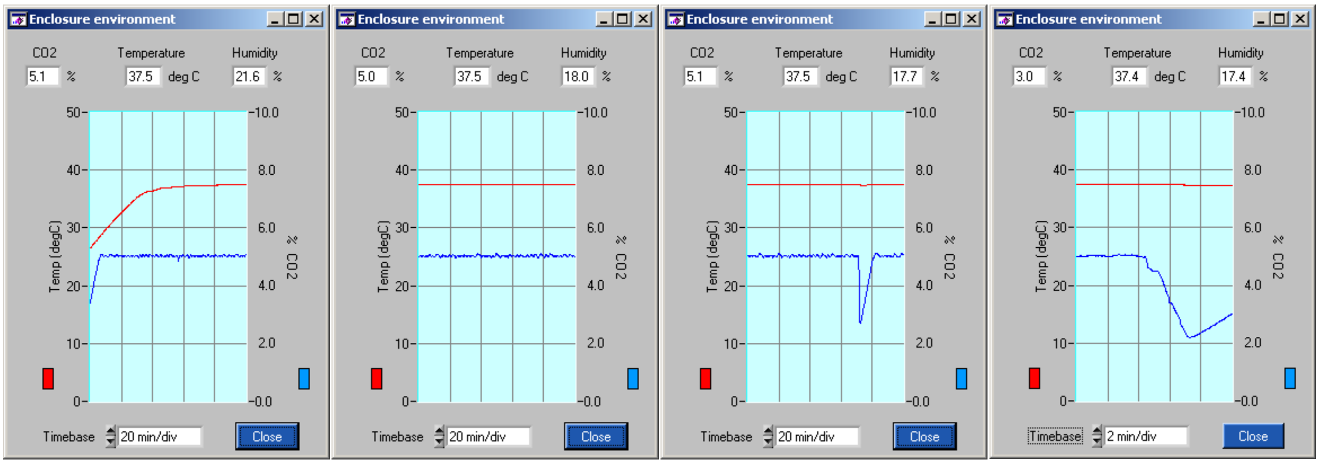


Figure 8: Internal layout of the control system inside the electronics enclosure. Plenty of room for additions!

5. Temperature, humidity and CO_2 display

For convenience and for logging purposes, we use a simple software interface which provides a chart recorder-like display of the enclosure environment. Actually only temperature and CO2 concentration are displayed on a chart, while humidity is displayed on a numerical indicator. The software graphical user interface is shown in Figure 9. This also shows the performance of the environment regulation, considered fairly respectable for a single-door incubator system.



A B C D

Figure 9: The environmental enclosure software graphical user interface which provides a readout of temperature, CO_2 and humidity, as well as chart recorder-like display of the first two variables over the last 10-100 minutes, depending on the timebase setting. A: settling time following switch-on. B: typical long-term stability. C: Kinetics of recovery following opening of the enclosure door for 30 seconds. D: as C, but on 2 min/division timebase and door opening for 2 minutes.

The software has been developed using National Instruments ‘CVI LabWindows’. It is provided for guidance as it is expected that other users may wish to use something more modern; we are old-fashioned and know very well this aging software environment, which allows straightforward ‘C’ programming. For completeness, we provide the listing below.

```
#include "cvixml.h"
#include <rs232.h>
#include <ansi_c.h>
#include <cvirte.h>
#include <userint.h>
#include "utility.h"
#include "formatio.h"
#include  <analysis.h>
#include "DeviceFinder.h"
#include "CO2 sensor readout_ui.h"
#include "usbconverter_v2.h"

#define NUMPOINTS 400
#define CO2PIC    0x16                          //Address of control PIC

int NUMBER_AVERAGES=3;

static int PORT;
static int chartPanel;
static int mode;
static int time_np[1000];
int temp_average=0,np=0;

double temp_data[1000],temp_CO2[1000],temp_humidity[1000];
double total_dtemp=0,total_dco2=0,total_dhumidity=0;

static int initI2Cport(void);
static int sendPosition(int);

void init_CO2_sensor(void);
void PlotGraph(double dhumidity,double dtemp,double dco2);
void TempStore(void);
```

```
static void EnglishDate(char *Date);
void StoreNewPlotRate(void);


int main (int argc, char *argv[])
{
	if (InitCVIRTE (0, argv, 0) == 0)
		return -1;	/* out of memory */
	if ((chartPanel = LoadPanel (0, "CO2 sensor readout_ui.uir", CHARTPANEL)) < 0)
		return -1;

	if(initI2Cport()==-1)return 0;		//Set port number

	DisplayPanel (chartPanel);
	init_CO2_sensor();
	Delay(1.0);
	SetCtrlAttribute (chartPanel, CHARTPANEL_TIMER, ATTR_ENABLED, 1);	//Enable timer
	RunUserInterface ();
	DiscardPanel (chartPanel);
	GCI_closeI2C();
	return 0;
}
static int getFTDIport(int *PORT)
{
char path[MAX_PATHNAME_LEN],ID[20];
int id_panel,pnl,ctrl;

//If we are using an FTDI gizmo Device Finder will give us the port number

	GetProjectDir (path);
	strcat(path,"\\");

	strcat(path, "CO2 sensor readoutID.txt");
	return selectPortForDevice(path, PORT, "Select Port for CO2 sensor");


}
static int initI2Cport()
{
int  err,ans;
char port_string[10];

if (getFTDIport(&PORT) == 0)
		sprintf(port_string, "COM%d",PORT);
	else {
		while(getFTDIport(&PORT) != 0){
			ans=ConfirmPopup ("Comms error", "Try plugging USB cable in or do you want to quit?");
			if(ans==1){	//quit
					return -1;

			}
		}

	}
	sprintf(port_string, "COM%d",PORT);
	err = OpenComConfig (PORT, port_string, 9600, 0, 8, 1, 512, 512);

	SetComTime (PORT, 1.0);			//Set port time-out to 1 sec
	FlushInQ (PORT);
	FlushOutQ (PORT);
	return 0;

}
void init_CO2_sensor(void)
{
int x=0;
char buffer[20],data[20];

	sprintf(buffer,"M 4166\r\n");		//Send output string

	ComWrt (PORT, buffer, 8);		//Send the set up command
	ComRd(PORT, data, 10);		//Read echoed returned bytes

	CallCtrlCallback (chartPanel,CHARTPANEL_TIMEBASE, EVENT_COMMIT, 0, 0, NULL); //Get timebase setting

	for(x=0;x<(NUMPOINTS);x++){			//Generate array for graph plotting
		time_np[x]=x;
		}
}

void PlotGraph(double dhumidity,double dtemp,double dco2)
{
int y;

	total_dtemp = total_dtemp + dtemp;
	total_dco2= total_dco2 + dco2;
	total_dhumidity = total_dhumidity + dhumidity;
	temp_average++;

	if(temp_average >= NUMBER_AVERAGES){
		temp_average = 0;		//Reset counter
```

```
                 for(y=0;y<(NUMPOINTS);y++){       //Shift old values up to make space for new values

                         temp_data[y] = temp_data[y+1];
                         temp_CO2[y] = temp_CO2[y+1];
                         temp_humidity[y] =       temp_humidity[y+1];

                 }

          //Place new values into array
         temp_data[NUMPOINTS] = total_dtemp/NUMBER_AVERAGES;
         temp_CO2[NUMPOINTS] = total_dco2/NUMBER_AVERAGES;
         temp_humidity[NUMPOINTS] = total_dhumidity/NUMBER_AVERAGES;

         DeleteGraphPlot (chartPanel, CHARTPANEL_CO2_TEMP_GRAPH, -1, VAL_DELAYED_DRAW);

     SetCtrlAttribute (chartPanel, CHARTPANEL_CO2_TEMP_GRAPH, ATTR_ACTIVE_YAXIS, VAL_LEFT_YAXIS);
         PlotXY(chartPanel, CHARTPANEL_CO2_TEMP_GRAPH,  time_np, temp_data, NUMPOINTS,
                         VAL_INTEGER, VAL_DOUBLE, VAL_THIN_LINE, VAL_NO_POINT, VAL_SOLID, 1, VAL_RED);

         SetCtrlAttribute (chartPanel, CHARTPANEL_CO2_TEMP_GRAPH, ATTR_ACTIVE_YAXIS, VAL_RIGHT_YAXIS);
         PlotXY(chartPanel, CHARTPANEL_CO2_TEMP_GRAPH, time_np ,temp_CO2, NUMPOINTS, VAL_INTEGER, VAL_DOUBLE,
         VAL_THIN_LINE, VAL_NO_POINT, VAL_SOLID, 1, VAL_BLUE);

         TempStore();
         total_dtemp = 0;    //Reset summations
         total_dco2 = 0;
         total_dhumidity = 0;

         }
}
static void EnglishDate(char *Date)
{
   char *DateAm;
   char Tempstr [4];
   int NoBytes;

   DateAm = DateStr ();
   CopyString (Tempstr, 0, DateAm, 3, 3);
   NoBytes = Fmt(Date,"%s<%s",Tempstr);
   CopyString (Tempstr, 0, DateAm, 0, 3);
   NoBytes = Fmt(Date,"%s[a]<%s",Tempstr);
   CopyString (Tempstr, 0, DateAm, 8, 2);
   NoBytes = Fmt(Date,"%s[a]<%s",Tempstr);
}

void StoreNewPlotRate(void)                           //Makes a note in file when plotting rate is changed
{
FILE *fp;
static fpos_t ptr;
int i, exists, size;
static int lastp;
char dataPath[MAX_PATHNAME_LEN],date[40];
char *time;

         EnglishDate(date);
         time = TimeStr();

         //Store data
         GetProjectDir (dataPath);

         strcat(dataPath,"\\enclosure environment\\");
         strcat(date," enclosure environment.csv");
         strcat(dataPath,date);

         exists = FileExists (dataPath, &size);

  if(!exists)
          return;
  else {                                                                          //Append to existing file
      fp = fopen (dataPath, "a");
         if (fp == NULL) return;
         SetWaitCursor (1);
         fprintf(fp, "Time %s\n",time);
                 fprintf(fp, "Plotting every %.1f seconds\n",(double)NUMBER_AVERAGES/2);
         fclose(fp);
         fp = fopen (dataPath, "r+");
                 fsetpos (fp, &ptr);
         fclose(fp);
  }

  SetWaitCursor (0);
}
void TempStore(void)
{
  FILE *fp;
  static fpos_t ptr;
  int i, exists, size;
  static int lastp;
  char dataPath[MAX_PATHNAME_LEN],date[40];
  char *time;
```

```
          EnglishDate(date);
          time = TimeStr();

        //Store data
         GetProjectDir (dataPath);

         strcat(dataPath,"\\enclosure environment\\");
         exists = FileExists (dataPath, &size);

  if(!exists)
    MakeDir(dataPath);

        strcat(date," enclosure environment.csv");
        strcat(dataPath,date);
    exists = GetFileInfo (dataPath, &size);

    //Start new file
  if (exists != 1) {
    fp = fopen (dataPath, "w");

    if (fp == NULL) return;
    SetWaitCursor (1);
    fgetpos (fp, &ptr);

 //Note where start of file is
    fprintf(fp, "%s\n",date);
    fprintf(fp, "Time %s\n",time);
    fprintf(fp, "Plotting every %.1f seconds\n",(double)NUMBER_AVERAGES/2);
    fprintf(fp," Temp,\t CO2,\t Humidity\n");
    fprintf(fp, "%.1f,\t%.2f,\t%.1f\n",temp_data[NUMPOINTS],temp_CO2[NUMPOINTS],temp_humidity[NUMPOINTS]);
    fclose(fp);
  }
  else {                          //Append to existing file
      fp = fopen (dataPath, "a");
        if (fp == NULL) return;
        SetWaitCursor (1);

        fprintf(fp, "%.1f,\t%.2f,\t%.1f\n",temp_data[NUMPOINTS],temp_CO2[NUMPOINTS],temp_humidity[NUMPOINTS]);
        fclose(fp);
        fp = fopen (dataPath, "r+");
        fsetpos (fp, &ptr);
        fclose(fp);
  }

  SetWaitCursor (0);
}

int CVICALLBACK cb_co2_close (int panel, int control, int event,
                 void *callbackData, int eventData1, int eventData2)
{
char val1[20];
        switch (event)
                {
                case EVENT_COMMIT:
                        SetCtrlAttribute (chartPanel, CHARTPANEL_TIMER, ATTR_ENABLED, 0);     //Disable timer
                        QuitUserInterface (0);
                        break;
                }
        return 0;
}

int CVICALLBACK cbtimer (int panel, int control, int event,
                 void *callbackData, int eventData1, int eventData2)
{
char data[50];
int num_bytes,humidity,temp,co2,uf_co2;
double dhumidity,dtemp,dco2;

        switch (event)
        {
                case EVENT_TIMER_TICK:

                num_bytes=40;
        ComRdTerm (PORT, data, num_bytes,13);                //Read until CR received
        sscanf (data,"%*s %d %*s %d %*s %d %*s %d",&humidity, &temp, &co2, &uf_co2);

        dhumidity=(double)humidity/10;
        dtemp=(double)(temp-1000)/10;
        dco2=(double)co2/1000;

        SetCtrlVal (chartPanel, CHARTPANEL_TEMP, dtemp);
        SetCtrlVal (chartPanel, CHARTPANEL_CO2, dco2);
        SetCtrlVal (chartPanel, CHARTPANEL_HUMIDITY, dhumidity);

        PlotGraph( dhumidity,dtemp,dco2);

        break;

        }
        return 0;
}
```

```
int CVICALLBACK cb_timebase (int panel, int control, int event,
                void *callbackData, int eventData1, int eventData2)
{
int timebase,y;
FILE *fp;
char dataPath[MAX_PATHNAME_LEN];

        switch (event)
        {
                case EVENT_COMMIT:
                        GetCtrlVal (chartPanel, CHARTPANEL_TIMEBASE, &timebase);
                        NUMBER_AVERAGES = timebase;

                for(y=0;y<(NUMPOINTS);y++){                        //Zero arrays
                        temp_data[y] = 0;
                        temp_CO2[y] = 0;
                        temp_humidity[y] =       0;

                }

                        DeleteGraphPlot (chartPanel, CHARTPANEL_CO2_TEMP_GRAPH, -1, VAL_IMMEDIATE_DRAW);
//Delete gragh
                        temp_average = 0;        //Reset counter
                        total_dtemp = 0;         //Reset summations
                        total_dco2 = 0;
                        total_dhumidity = 0;

                        StoreNewPlotRate();

                        break;
        }
        return 0;
}
```

6. Enclosure electronic and mechanical components

Item	Note	Manufacturer / model	Supplier Part no.	Qty	Price
CO_2 sensor	NDIR 0-20% CO_2	Cozir W20-VH * see note	Gas Sensing Solutions Ltd	1 off	£ 140
Temperature sensor	Pt 100 type 2 x 10 mm	Labfacility DM-333	One Call 859-8533	1 off	£ 4.40
PID temperature controller	Pt 100 input	Omron E5CN-R2ML	RS Stock 535-367	1 off	£ 151.00
PID CO_2 Controller	Linear input	Omron E5CN-Q2MT	RS Stock 535-436	1 off	£ 151.00
USB interface module	UART type i/o	FTDI TTL-232RG-VREG3V3-WE	RS Stock 715-8529	1 off	£ 19.01
USB interface module connector	10 way IDC socket	Harting 510 6803	One Call 120-0502	2 off	£ 2.48
Solid State Relay	DIN rail mount 280Vac 20A	Crydom HS351DR-D2425	RS Stock 703-4586	1 off	£ 37.10
Heater fan	119 x 119 x 38mm 160m³/h	EBM-Papst 4650NU	RS Stock 749-2021	1 off	£ 29.22
Fan lead	To connect to fan header	EBM Papst LZ120	One Call 959-9720	1 off	£ 3.18
Heater	230V ac 400W	Stego Elektrotechnik 03112.0-00	RS Stock 282-1275	1 off	£ 136.57
Damper	Male-male bobbin, M4 studs, 10 mm dia.	Fivistop1008VV10-45	One Call 499-5909	4 off	£ 6.68
Solenoid valve	2 Port Solenoid valve 24 V DC 1/8 bsp	SMC VDW21-5G-3-01F-Q	RS Stock 701-3236	1 off	£ 16.67
Front panel on-off switch	20 A peak current	Marquardt 1835.3118	One Call 1831115	1 off	£ 4.25
DC power supply 24V DC - 1A out	85-264 V ac in DIN Mount	RS Essentials	RS Stock 428-433	1 off	£ 34.18
Lighting relay DIN rail panel mount	SPST-NC, SPST-NO 24V dc 460 Ω	Finder 22.23.9.024.4000	RS Stock 511-1183	1 off	£ 14.72
LED Gas on indicator	Green LED indicator12V DC	APEM Q8F3BXXG12E	One Call 200-8535	1 off	£ 3.11
LED Heater on indicator	Yellow LED indicator12V DC	APEM Q8F3BXXY12E	One Call 200-8536	1 off	£ 3.11
LED current limit capacitor	150 nF 275 V ac	Panasonic ECQUAAF154M	RS Stock 739-8626	1 off	£ 0.281
Fuseholders	1 1/4" screw cap; rear panel	Bulgin FX0415/S	One Call FF01327	3 off	£ 5.97
IEC hot condition AC power inlet	10 A current carrying capacity	Bulgin PX0590/63	One Call 151-745	1 off	£ 1.07
IEC hot condition cable socket	10 A current carrying capacity	Bulgin PX0597	One Call 117-2515	1 off	£ 2.25
Enclosure lighting strips	LED 500 mm	RS Essentials	RS Stock 701-2886	2 off	£ 50.86
Illumination on-off door switch	Proximity Switch 500 mA 50V dc NO	RS Essentials	RS Stock 333-192	1 off	£ 1.45
3.3 V regulator, TO-220	UA78M33CKCS	Texas Instruments	RS Stock 661-6737	1 off	£ 0.356
Regulator capacitor	10 µF, 25V	Vishay BC Components MAL203036109E3	One Call CA07171	1 off	£ 0.30
Enclosure	19" case, Black 3U	Hammond RM3U1913SBK	One Call 187-7359	1 off	£ 70.24
DIN rail	0.5 m long 7.5 mm high	RS Essentials	RS Stock 467-406	1 off	£ 2.65
DIN rail enclosure	36 mm wide for regulator, capacitor etc.	Camden CNMB/2ST/2	RS Stock 749-5938	1 off	£ 4.49
DIN rail connector blocks	Double-decker din rail terminal 2.5 m	RS Essentials	RS Stock 501-950	12 off	£ 12.58
DIN rail connector end-plate	End cover for double deck terminal	RS Essentials	RS Stock 815-717	1 off	£ 0.351
Cable grommets	For taking heater power into enclosure	Pro Power 241/69/79B	One Call 143-607	2 off	£ 1.40
USB interface CO_2 connector	10 way header	3M 4610-6050	One Call 525-091	1 off	£ 3.25
Heater/fan interconnector	4 way socket	TE Connectivity 206430-2	One Call 592-857	1 off	£ 2.00
Heater/fan interconnector	4 way plug	TE Connectivity 182651-1	One Call 589-550	1 off	£ 3.00
Door switch interconnection	Socket, 3 way, free	Binder 09 9748 70 03	One Call 112-2804	1 off	£ 2.99
Temperature sensor interconnection	Plug, 3 way, free	Binder 09 9747 70 03	One Call 112-2802	1 off	£ 2.60
Door switch interconnection	Panel plug, 3 way	Binder 09 9749 30 03	One Call 112-2806	1 off	£ 2.18
Temperature sensor interconnection	Panel socket, 3 way	Binder 09 9750 30 03	One Call 112-2808	1 off	£ 2.26
Lighting interconnection	Panel socket, 3 way	Bulgin SA 2404	One Call 314-031	1 off	£ 2.39
Lighting interconnection	Plug, 3 way R/A	Bulgin SA 2403	One Call 314-020	1 off	£ 3.98
Plastic box for temperature sensor	Box, ABS, black, 18x40x28 mm	Camden-Boss RX2006/S	One Call 187-1051	1 off	£ 0.94
Plastic box for CO_2 sensor	Box, ABS, black, 23x54x38 mm	Camden-Boss RX2008/S	One Call 1871053	1 off	£ 1.28
Timer relay to disable CO_2 solenoid	True off delay timer	Broyce Control M1EDF 24VAC/DC/230VAC .5-10MI	RS Stock 300-6045	1 off	£ 36.00
				Total	**£ 959.17**
Enclosure thermal insulator	Corrugated Polypropylene sheet	Black 4mm / 700gsm - pack of 5; 2440mm x 1220mm	The Plastics Shop		
Enclosure handles	2 off on 2 doors	KJN 3842525767		2 off	--
Enclosure hinges	4 off on 2 doors	Bosch Rexroth 3842535687	RS Stock 528-9400	4 off	--
Enclosure window seals	Grey rubber seal	Bosch Rexroth 3842523494	RS Stock 022-7567	As req'd	--
Enclosure door seals	Black rubber seal	Bosch Rexroth 3842516598	RS Stock 437-1312	As req'd	--
Enclosure framework	30 mm / 8 mm slot	Bosch Rexroth3842990720/3000	RS Stock 389-9796	As req'd	--
Enclosure corner joints	Quick-connect	KJN 535459	KJN 535459	10 off	--
Cover strip	PVC cover stip	Bosch Rexroth 3842501962	RS Stock 418-0841	As req'd	--
Door material	Natural Grey see-thru Perspex 6 mm	2 off offcuts from 1000 x 660 x 6 sheet	Bay Plastics Ltd	2 off	--
				Total	**~£300**

* Ensure at time of ordering that analogue output facility is enabled

Most of the items can be obtained from usual component distributors such as RS and One Call/Farnell. Some items are 'special' and can be obtained from suppliers listed above; their contact details are provided below for completeness.

Gas Sensing Solutions Ltd
60 Grayshill Road, Westfield North Courtyard, Cumbernauld, Glasgow G68 9HQ
+44 (0) 1236 781 900
http://www.gassensing.co.uk/product/cozir-wide-range/

KJN Automation Ltd
Unit 5, Peckleton Lane Business Park, Peckleton Common, Leicester, LE9 7RN
Tel: +44 (0)1455 823304
http://www.kjnltd.co.uk

The Plastics Shop
16 Bayton Road, Bayton Road Industrial Estate, Coventry, CV7 9EJ
024 76 588 383
http://www.theplasticshop.co.uk

Bay Plastics Ltd
Unit H1, High Flatworth, Tyne Tunnel Trading Estate, North Shields, Tyne & Wear, NE29 7UZ
(0) 191 258 0777
http://www.plasticstockist.com/

The overall cost is <~£1300 and the unit can be assembled in a 3-4 days, making the effective cost ~£2000, considerably less than equivalent commercial microscope incubators. The performance is superior to most commonly available incubators and the stability and reproducibility of focus is excellent.

This note was prepared in December 2012 by B Vojnovic, IDC Tullis, J Prentice and RG Newman, who also constructed the electronics; J Prentice and G. Shortland constructed the enclosure, from an outline design by B Vojnovic.

The financial support of Cancer Research UK, the MRC and EPSRC is gratefully acknowledged.