CHT EGT Aviation Instrumentation System

Introduction

This project is designed to display all Cylinder Head Temperature (CHT) and Exhaust Gas Temperature (EGT) for a four-cylinder engine on a QVGA TFT display.

It is designed so as to allow the addition of other transducers, such as manifold pressure, oil temperature, airspeed etc, to be displayed.

The project was designed specifically for aviation engines. It is experimental, so will require work to get it going.

The files in this package:

Arduino code to display data
Data acquisition shield schematic (Eagle)
Data acquisition shield board
Photo of assembled system
Photo of screen display

It uses an Arduino UNO compatible processor board together with a custom Arduino shield, and a cheap Arduino screw shield to break out the Arduino pins for connection to the TFT display and perhaps other transducers.

CHT and EGT measurement require the use of K type thermocouples that produce a small voltage that is linearly related to the temperature. Temperatures in excess of 750C can be measured.

Suitable cheap CHT and EGT K type thermocouples can be purchased on Ebay.

However, the voltage produced by these thermocouples is very small, so has to be scaled up before being converted to a digital value. The custom shield uses a pair of chips designed to select one of 8 inputs, (a multiplexer) and a k type thermocouple amplifier. The shield also contains rugged switch mode power supplies to provide both 5V and 9V, enabling the system to easily power a range of other transducers. These power supplies are overkill if all that is required is just the measurement of CHT and EGT, but the system has a huge amount of additional capacity, designed to power a range of possible transducers and an additional screen, and the cost of the extra capacity is very little. There are scores of data acquisition boards and sample code for use with Arduino systems.

Further, the shield allows the selection of two addressing systems, so two of the shields can be stacked and all CHTs and EGTs for a 6 or 8 cylinder engine can be measured. The power supply sections of the second shield do not have to be installed.

The System

The basic system consists of 3 circuit boards stacked one on top of each other, with signals carried through from one board to the next, following the Arduino design.

The bottom board is the custom CHT EGT board. The next board is the Arduino board, and the top board is an Arduino screw shield.

It is possible to stack more boards, but care must be taken to ensure there are no conflicts between the Arduino pins that are used.

The screen is an Adafruit 2.8 inch TFT screen, using an SPI interface. There are cheap similar Chinese clones, but they do not appear to work with the Adafruit Arduino library, and have no native library, even though they use the same processor. So, stay with the Adafruit screen. The resolution is QVGA, that is, quarter of the old VGA computer screen standard, or 320*240 pixels. The screen is powered by the CHT EGT board.

It is possible to use 2 or more of these screens with this system.

The CHT EGT board

This board conforms to the Arduino UNO form factor. IT will accept power input from either a 12V or 24V system. The input is via the screw terminal. The on-board voltage regulators supply 5V and 9V, and this power is made available through the 5V and 9V pins on the Arduino connectors. Note: do not use the Vin on the Arduino board for input voltage, only input voltage from your aircraft bus through the screw terminals.

The board is designed using Eagle PCB. If you wish to get the board manufactured, most manufacturers will accept Eagle PCB board files. The board has a ground plane on the bottom layer. This sometimes disappears so you need to specify to a manufacturer to regenerate the ground plane, (ratsnest command) and check that it has been done before giving the OK to manufacture.

The schematic is included in the package. It is worth your while studying this to understand how it works.

The board uses a combination of surface-mount and through-hole components. When building up the board, place the surface-mount components first. I use the electric frypan method. There is information about this on the internet. Basically, you put just a tiny bit of solder on each of the surface-mount pads, and a tiny bit of residue-free flux, and then carefully sit the components on top. Place the board on an electric frypan turned onto high, and wait until you can see the solder go shiny, indicating it has melted. Turn the frypan off and wait until it cools until you try and move the board. The surface tension in the melted solder will cause the components to settle into place.

Then mount the through-hole components in the normal way.

The components are widely available, most through Ebay. Make sure that the capacitors you use exceed the voltage of your aircraft system by at least 10V.

Solder a link for the addressing mode. This is close to pin 13. You will see that there are three holes. Connecting the two closest to the connector will give the addressing mode used in the program. Connecting the centre pin to the one closest to the screw terminal will provide the alternate addressing which you might use if you are using a second board for a 6 or 8 cylinder engine. Consult the schematic.

This particular set of addresses was used so as to leave the Tx and Rx lines free to connect to a serial peripheral such as a GPS.

You need to be familiar with what pins have been assigned in this configuration before adding additional boards.

Warning: The Arduino UNO uses 5V. If you wish to use one of the new ARM-based Arduinos that run at 3.3V, change the 5V regulator for a 3.3V version.

The Arduino board

The system has been tested on one of the cheap Chinese Arduino UNO clones and it works well. However, they use a different USB chip and you will have to install a driver before you can connect to the board.

You will need to unsolder the connectors running down each side of the board and replace them with stackable ones. These are widely available on line.

This is so you can plug the board into the thermocouple board, and stack the screw terminal board on top.

Take care not to lift the tracks on the board when desoldering. Be gentle.

These Arduino UNO boards are now pretty minimal compared with what is available, and should you require either more power or more IO capacity, look at alternative boards such as the Arduino mega.

The screw terminal board

These are widely available on Ebay and cheap. They allow you to use screw terminals to connect to any of the Arduino I/O lines. This is particularly useful to connect securely to the screen.

The Adafruit TFT screen

This screen provides a bright, clear colour display. It connects via the SPI interface, a clocked serial system. This interface also has a select pin. This means that you can use more than one screen, with mostly common connections to the SPI pins, changing only the select pin.

The Adafruit library provides all the functionality you could possibly want.

The library allows you to declare an instance of the TFT class and specify the select pin you are using. You can then specify another instance of the class and specify a different select pin. This should be clear from the included code.

Note the comments on using the cheap Chinese clones of this TFT board. I can't make them work, and also feel some moral responsibility to use the Adafruit board as they have gone to the trouble of writing the library.

The code

I hope I have commented this sufficiently to make it clear. In order to avoid flicker I have updated the screen only when the data changes. The hottest cylinder is shown in red. The temperature is in degrees centigrade, but could readily be changed to degrees Fahrenheit. I have scaled the length of the bars to show the temperatures for a Jabiru engine, which runs relatively cool CHTs. Rescaling for hotter CHTs only requires changing the constant at the top of the program.

Note that in order to allow for decreasing temperatures it is necessary to overwrite the temperature bars with black bars. If the temperatures exceed the maximum allowed for in the program, you will end up with bits of bar overwriting the text area at the top of the screen. Up to you to fix that problem.