Introducing PCB111000_CP2102

The purpose of PCB111000_CP2102 is to provide an introduction to C programming for micro controllers. A micro controller is a programmable electronic circuit built on a slither of silicon. It can be used to control many things, in this case an 8 digit display and to detect when switches are being operated.

The idea is that students write a number of fairly short C programs for a particular micro controller. As part of this exercise they look at a few key sections of the micro documentation (its data sheet) which give precise details about how to drive it.

A commentary is programmed into the micro controller which students can print out in small chunks as an aide to the learning process. An excellent text by Jo Pardue is referred to. It includes humour and anecdotes. These ease the often stressful process of grasping exactly what is going on. The book can now be downloaded for free.

In the first photo shown below see

The blue CP2102 module:	This enables the PCB to communicate with a PC
A 28 pin device:	This is an Atmega328 and is the device that will be programmed.
A vertical switch	This resets the device and is known as (k/a) the reset control switch
Three horizontal switches	These are the user switches and enable the user to input data
Two 4 digit display modules	Most of the user programs will be used to drive this display

The second photo hows a 20 pin device (an ATtiny1608). This uses:
16 of its pins to drive the display (each digit has 8 segments)
Two for power (0V and +5V)
One that enables it to be programmed
One that enables it to communicate with the Atmega328.

The PCB comes with a built in program that runs when power is first applied (to check that everything is OK).

Milestones

1 Downloading a number of applications from the web including the Arduino IDE and establishing communications with the PCB. See the User guide for help with this step.

2 Building (compiling) the first program (a freebie) uploading it to the PCB and making it run.

3 Choosing several programs from a number of examples, entering them onto the PC using Arduino, getting them to compile and finally uploading them to the PCB.

4 Taking a look at communications between the PCB and PC, with numbers and text going in both directions.

5 Multi tasking: Using interrupts to enable the micro to perform several jobs at once. For example driving the display and printing out the results of some arithmetic.

6 Tring to think of something useful to build: May be a clock or stop watch and entering data from the user switches.

7 Using some Arduino library functions to help input and output numbers

8 Taking a look at floating point numbers and building a simple calculator.

The idea here is that learning should be more about getting familiar with what works and how things are done, hopefully having some fun at the same time, with as little effort as possible and feeling encouraged to experiment.

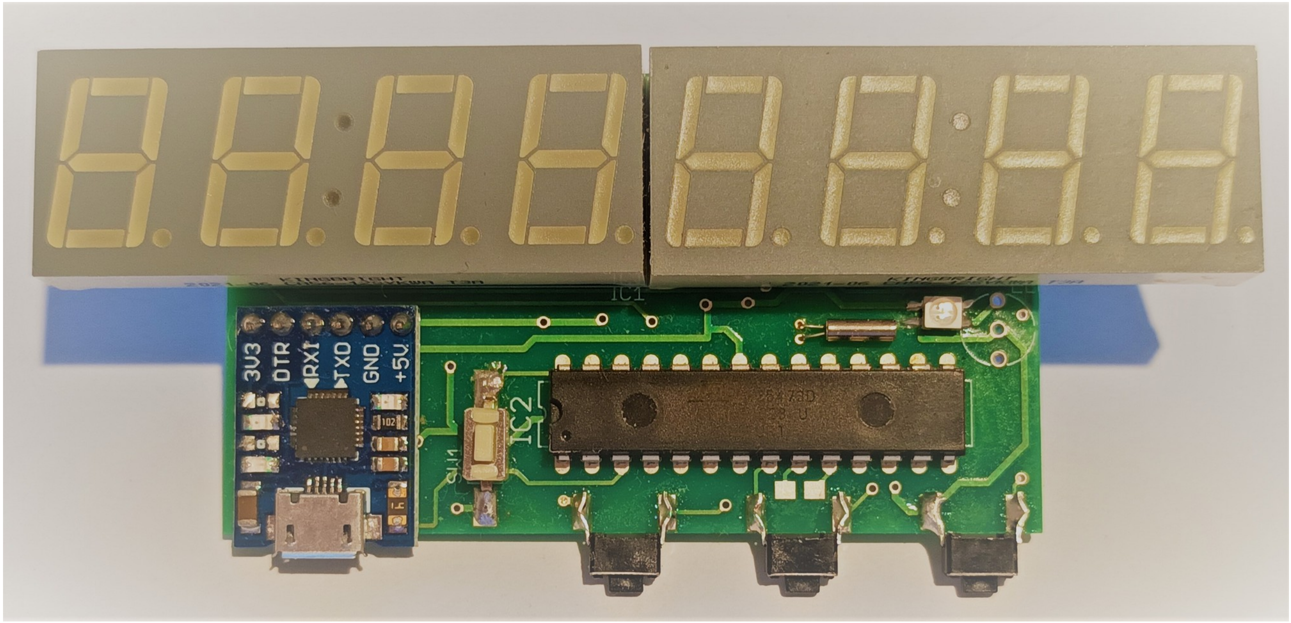

Photo of PCB111000_CP2102 viewed from above

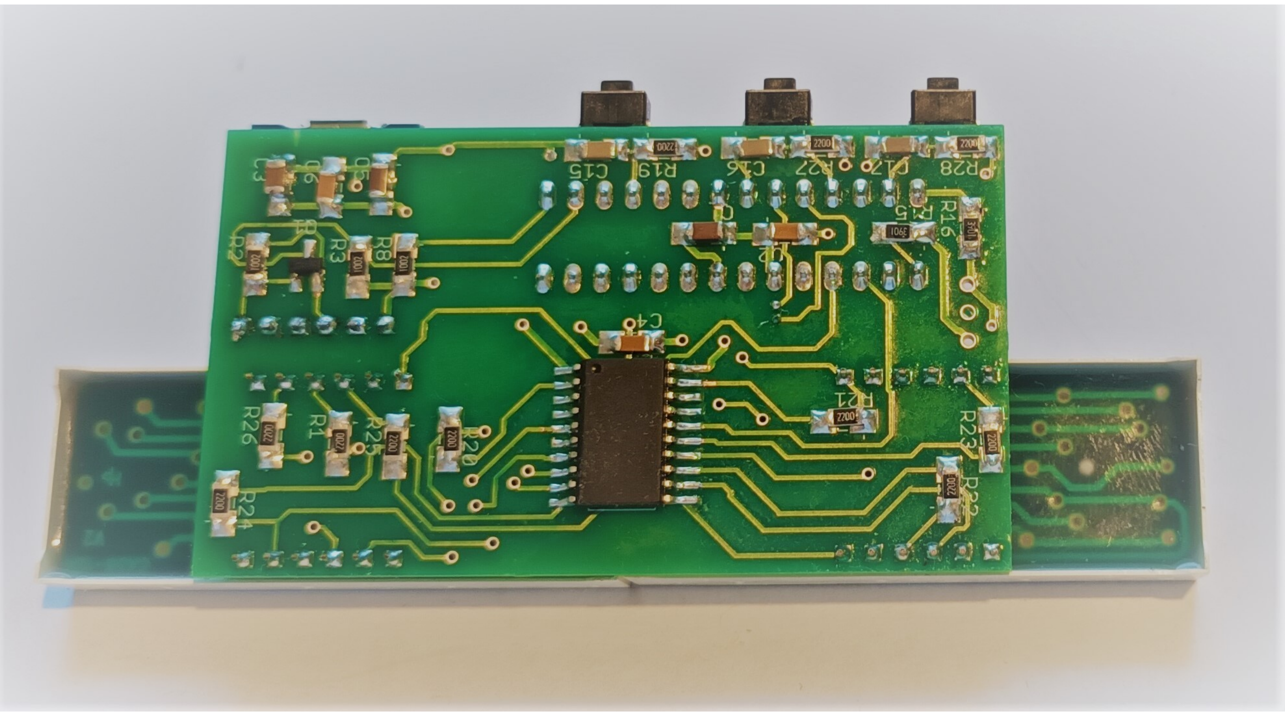

Photo of PCB111000_CP2102 viewed from below