Presenting PCB 111000: 56 Projects for the novice C-programmer

Features Atmega devices, programmer and mini-OS and Introduces the free WinAVR compiler

Target audience: Anyone wishing to try out WinAVR, who also wants a basic pcb to check that they have set up their WinAVR development environment correctly without having to worry about: hardware, programmers, configuration bytes, makefiles, projects, power supplies or any knowledge of electronics and

1. Who would like a brief introduction to writing C for the Atmega family of devices and using it to set up some of the on-chip peripherals.

2. Who would like sample projects, each of which comes with a commentary and provide an introduction to:

   a. Basic C features including: loops, strings, variables, types, logic, binary, hex.

   b. Basic hard ware features including: Timer/counters, the watchdog timer, the UART, interrupts, the EEPROM, Reset functions, displaying numbers, data input using the user switches, the I2C bus.

   c. The WinAVR maths library functions and some of the I/O subroutines.

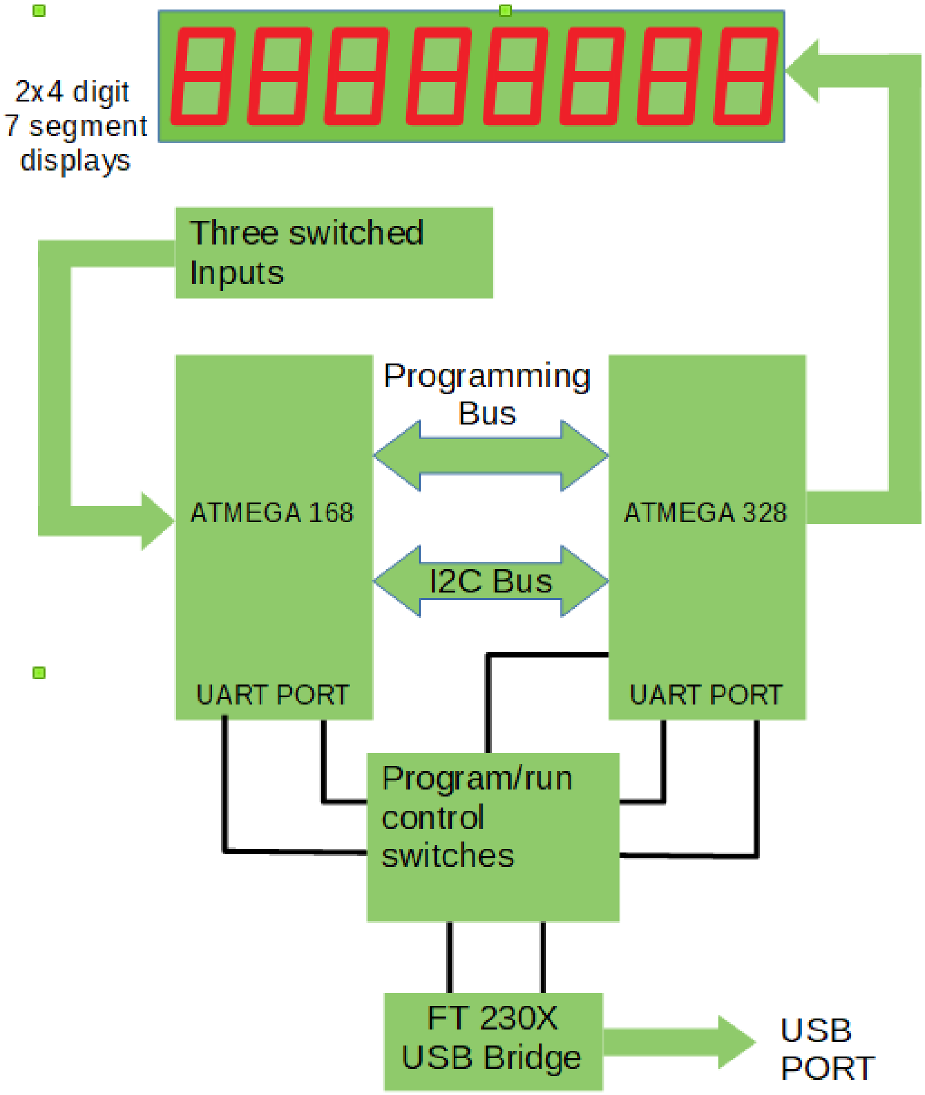
3. Who would like a platform on which to try out some on the numerous program segments available on internet forums.

System description

The pcb contains the following circuit blocks:

1. An Atmega 328 that drives an 8 digit 7 segment display and downloads user code to an Atmega 168. (See Figure overleaf.)
2. An ATMEGA 168 for which the user writes simple C-programs
3. Three switches that can be used to provide inputs for the ATMEGA 168
4. An FTDI 230X USB bridge that enables the module to be connected to a PC.
5. A programming interface: This is normally used to program the ATMEGA 168 but can also be used to update the ATMEGA 328 code.
6. An I2C interface via which user code controls the display.

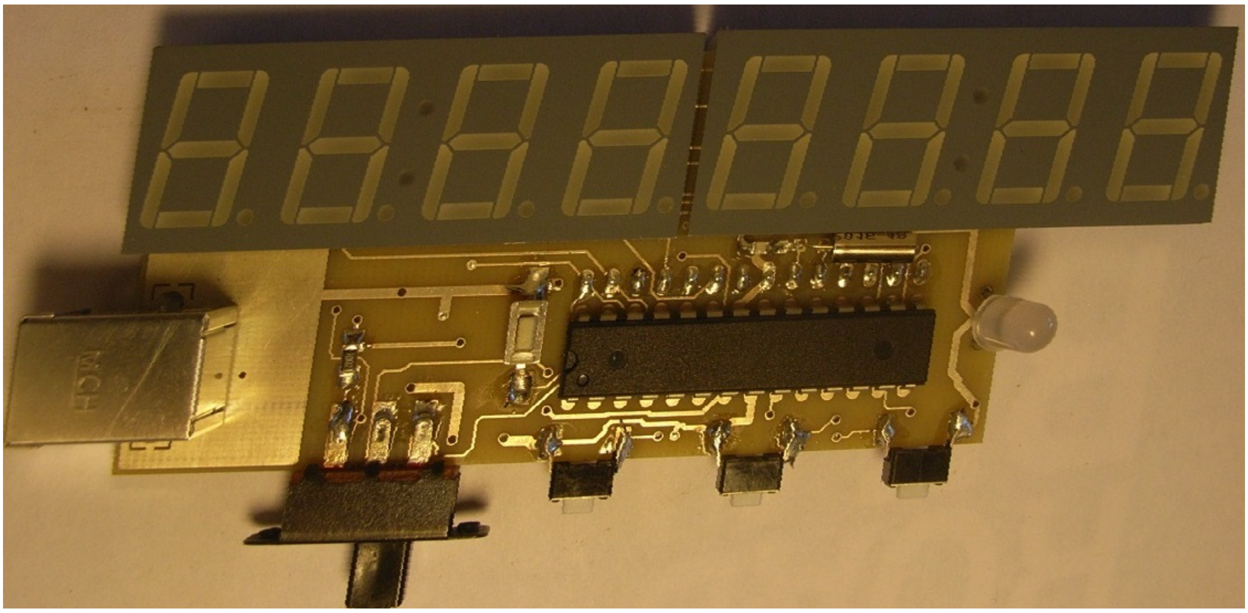
System Diagram

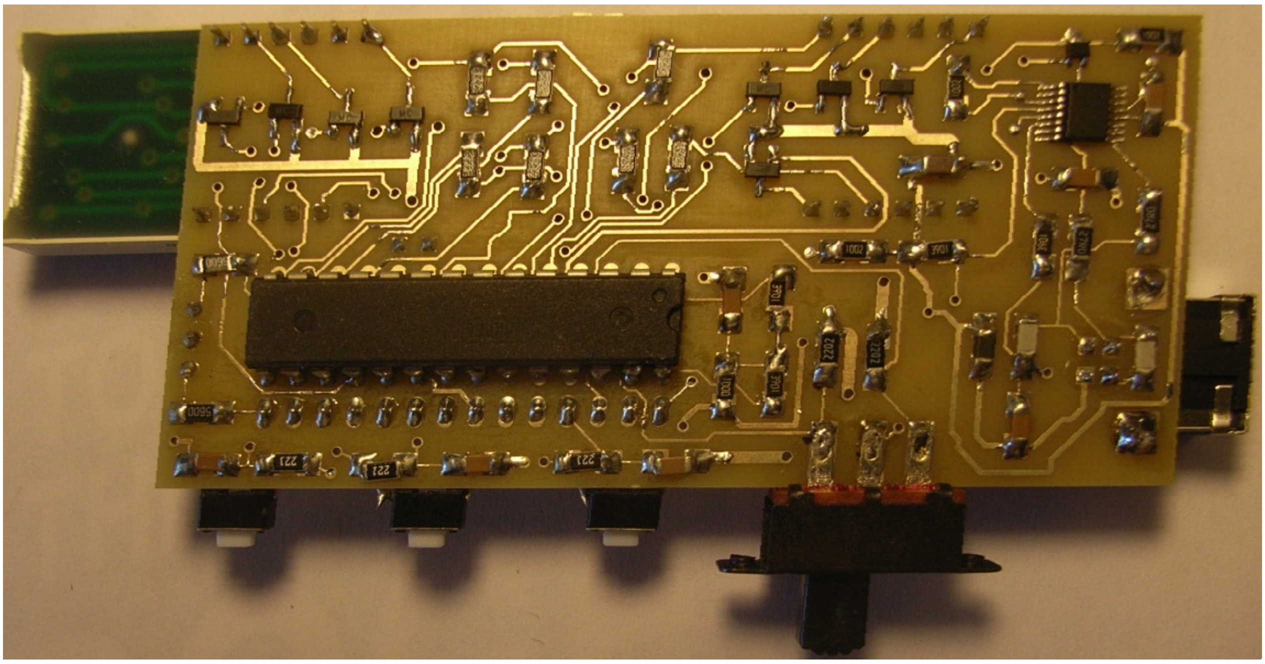


ATMEGA 168 hosts: User programs

ATMEGA 328 hosts: Programmer,
I2C master, display driver, clock, stopwatch
basic calculator……..

Photographs of a prototype version of the ATMEGA bootloader PCB





Other resources:
A terminal program such as Tera Term or Br@Y++.
A copy of Jo Pardue's book “C programming for Microcontrollers” which is available for free on the internet.
(Try “https://epdf.tips/c-programming-for-microcontrollers.html”.)