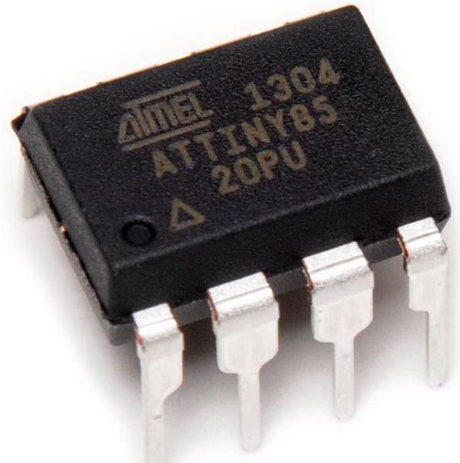
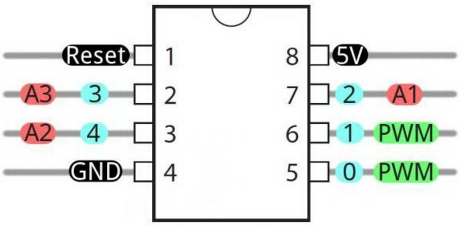
Attiny85 – Little Brother of Arduino

By Kevin Mc Donald ZS6KMD

Welcome to this month's informative article on Arduino, and let me start by introducing you to our new friend the Attiny85. Little brother to the Atmega 328 used in the Arduino. The Attiny is a low-power Atmel 8-bit AVR RISC-based microcontroller combining 8KB ISP flash memory, 512B EEPROM, 512-Byte SRAM, 6 general purpose I/O lines, 32 general purpose working registers, one 8-bit timer/counter with compare modes, one 8-bit high speed timer/counter, USI, internal and external Interrupts, 4-channel 10-bit A/D converter, programmable watchdog timer with internal oscillator, three software selectable power saving modes, and debugWIRE for on-chip debugging. The device achieves a throughput of 20 MIPS at 20 MHz and operates between 2.7-5.5 volts. All of that crammed into a small 8 pin device.

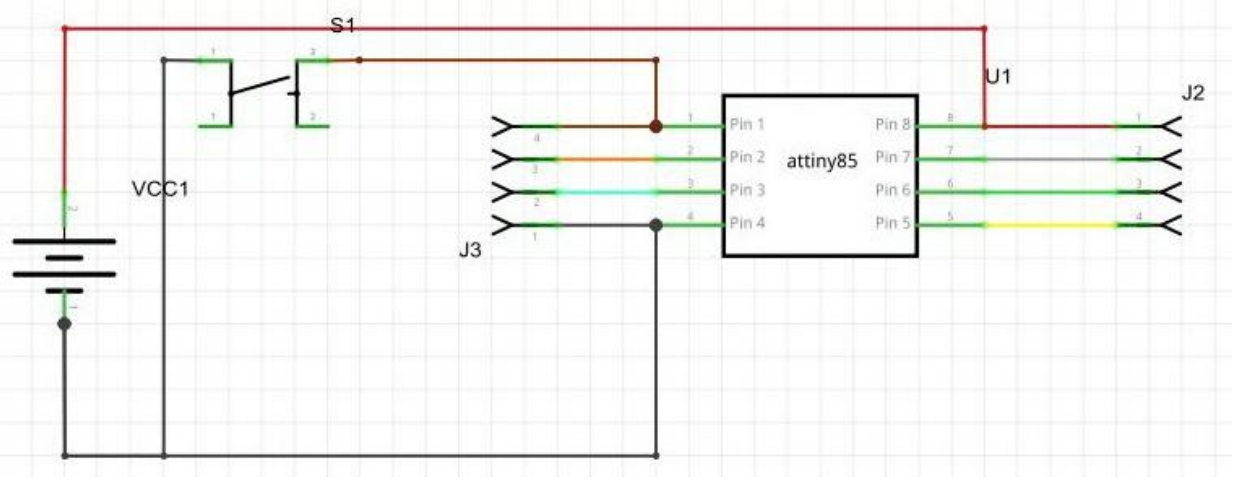




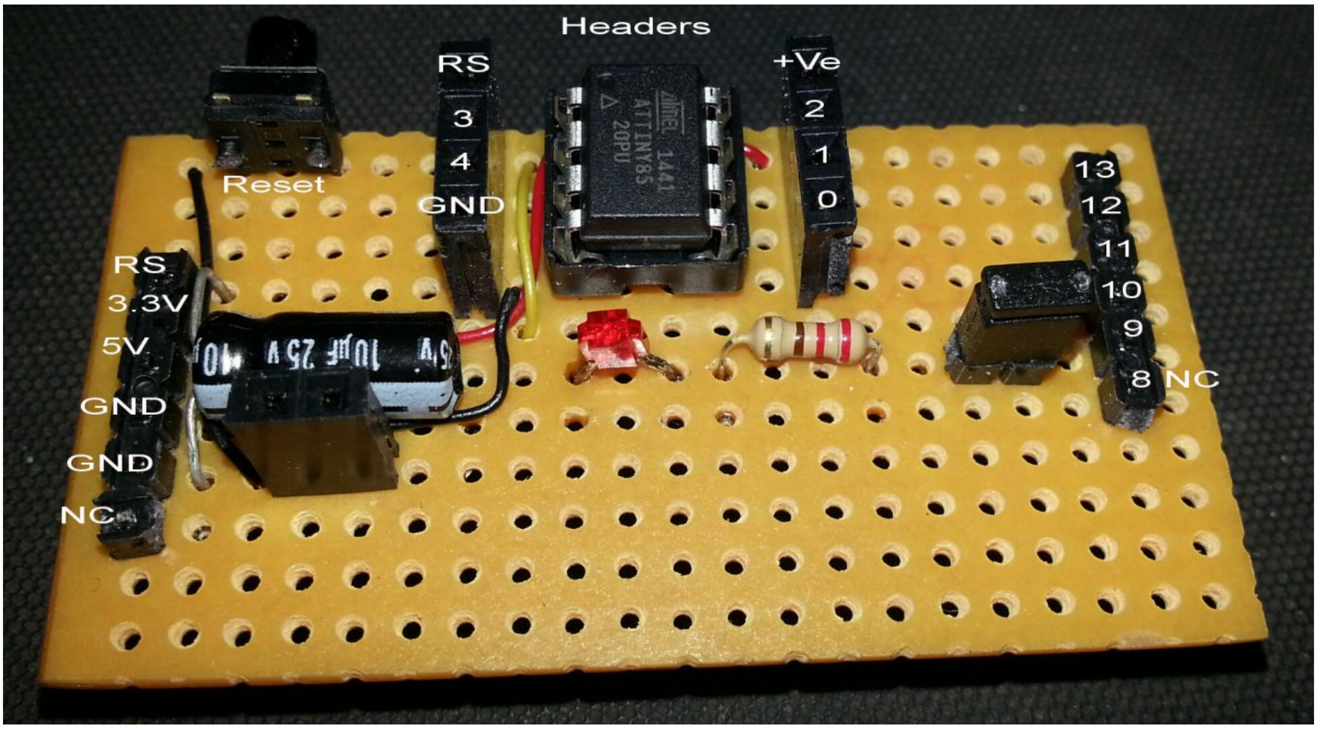
So what can you do with the Attiny you may ask… For starters, you can miniaturise many of your Arduino projects. The code used by both is identical, the only real limitation is the size of the memory and the fact that the Attiny does not have the normal RX/TX serial pins and needs to be programmed using an AVR ICSP programmer or another Arduino in ICSP mode. Additionally, you could build devices from games to robotics and even the odd Amateur Radio projects such as Iambic morse keyers, intelligent battery chargers and so on.

Now that I have introduced you to the Attiny85, I will show you how to build an interface to program it using a standard Arduino Uno as well as one using the ISP interface from an AVR programmer.

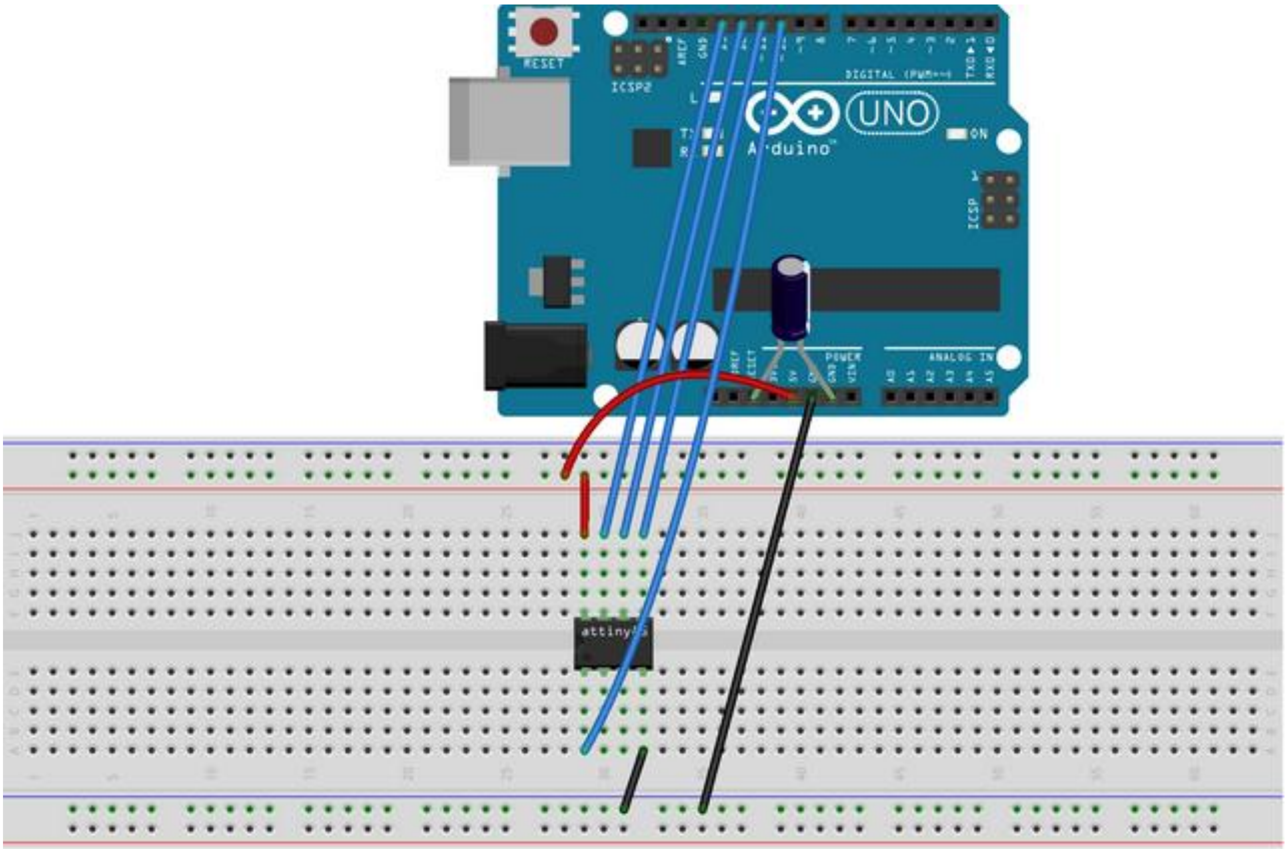
The images below show my prototype shield developed to fit on top of the Arduino Uno or KMDuino ONE. The basic circuit for the Attiny and the additional requirements to allow it to be programmed.



On my prototype I used a piece of Vero Board 11 rows by 21 holes, some male header pins, some female header sockets (so we can use the Attiny once programmed), a reset switch, a 10uF capacitor(required by the Uno for reset) an LED and pull up resistor and a socket for the Attiny so we can remove it once programmed or program many Attiny's.

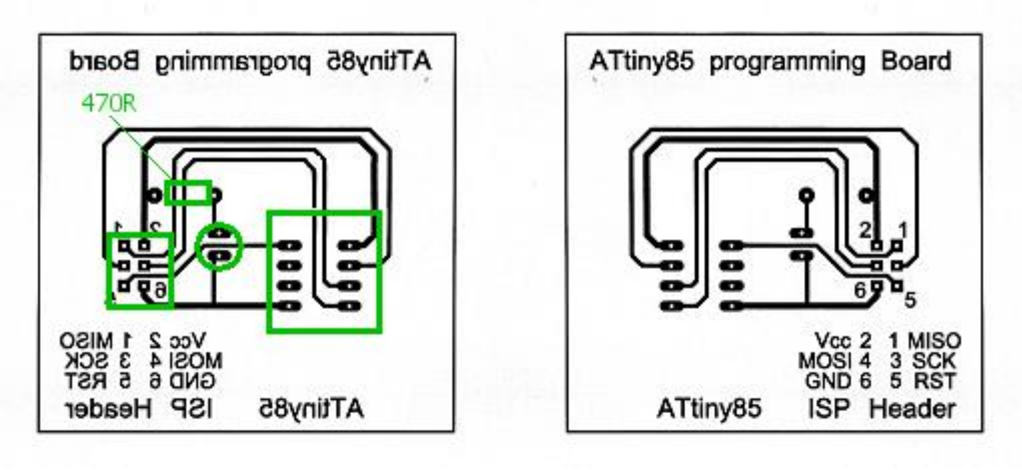


The connections to the above circuit that are required to make it possible to program the Attiny are shown in the Arduino image below.



Pins are connected as follows, Gnd to pin 4, +5V to pin 8, Pin13 on Uno to pin 7, Pin12 on Uno to pin 6, Pin11 on Uno to pin 5, Pin10 on Uno to pin 1(Reset). Pin 9 goes through a 220R resistor and LED to ground as a heartbeat to show that it is being programmed.

As I said before I would also show the design for an AVR ISP, this can be built on a small piece of PC board using the following layout.



All that is required is a 470R resistor, an LED, an IC socket and a 6way ISP header. This programming board can be used with a USBASP programmer or the Arduino ISP pins.

Next month we will look at programming the Attiny with some basic instructions like blinking and fading an LED and then building a capacitive touch Iambic morse keyer or some other useful items for the shack. Please send your pictures of your boards and projects to projects@zs6kmd.za.net and we will include them in future articles

Happy building

73