Arduino Display for miniVNA

Karl Berger, W4KRL

AMRAD - Vienna Winterfest

February 24, 2013

AMRAD Single Board Computer Projects

AMRAD is promoting the development of amateur radio applications for single board computers such as the Arduino, Raspberry Pi, PICAXE and BASIC Stamp.

Modern single board controllers are becoming more powerful even as they become cheaper. They are ideally suited for station control, test instruments, and software defined radio.

About AMRAD

- **The purpose of AMRAD is to develop skills and knowledge in radio and electronic technology, advocate design of experimental equipment and techniques, promote basic and applied research, organize technical forums and symposiums, collect and disseminate technical information and provide repeaters.**
- AMRAD began as an informal group in 1972 and incorporated in the Commonwealth of Virginia in 1975 as a nonprofit organization, Corporate ID 0159906-7.
- The U.S. Internal Revenue Service recognizes AMRAD as a 501(c)(3) tax-exempt scientific and educational organization, EIN 51-0176767.
- AMRAD is affiliated with the National Association for Amateur Radio, ARRL.
- AMRAD is a member of The Mid-Atlantic Repeater Council (T-MARC).

AMRAD Near Term Project Goals

•Combine the miniVNA with an Arduino controller and LCD touch screen shield to make a completely self-contained “stand alone” test instrument.

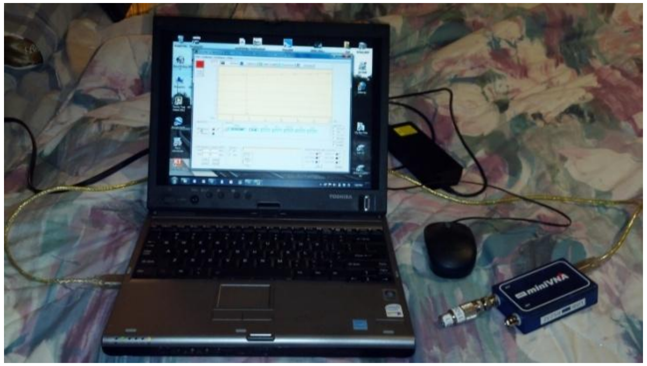
•Interface between the Arduino and miniVNA is via a 3-wire RS-232 connection. No level shifting is necessary.

•The touch screen shield comes with a micro SD card holder. It should be possible to store test results on the SD card and read them on a standard PC.

•Follow the progress at www.w4krl.com

- Develop low cost color touch screen display for "Stand Alone" measurement of SWR with a miniVNA.
  - Learn how to program the Arduino.
  - Learn how to interface the Arduino and miniVNA.
  - Design battery supply for both units.
  - Store measurements on a micro SD card.
- Share design and experience with AMRAD community. www.amrad.org

miniVNA
Vector Network Analyzer
Mini Radio Solutions

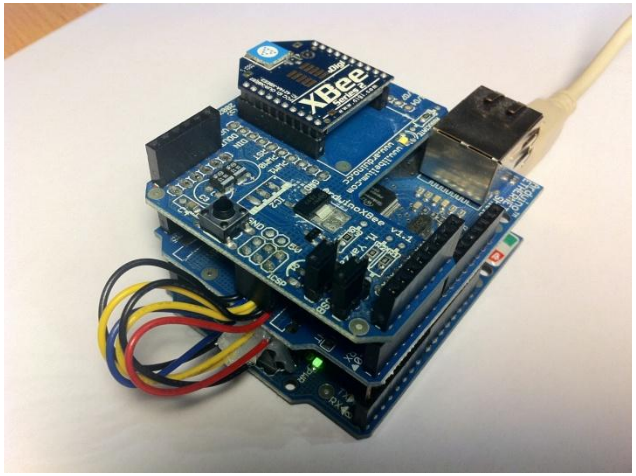


- The miniVNA is a versatile test instrument that measures antenna SWR, cable parameters, filter performance and many RF network values.
- It is small and portable.

BUT

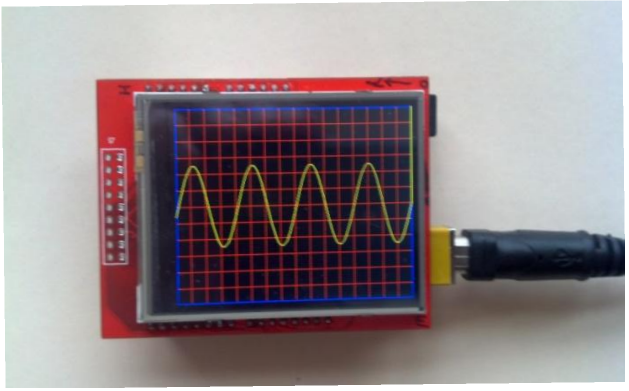
- It must be connected to a PC for power, control and output display.

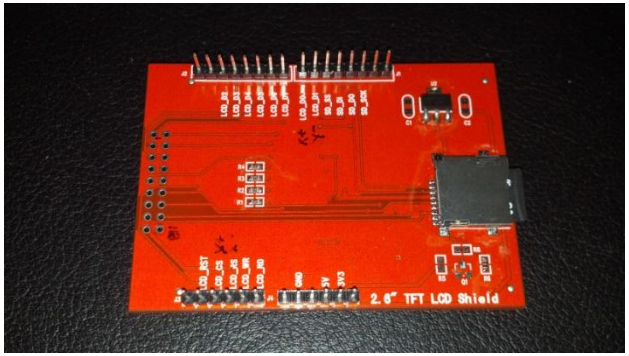
Arduino UNO rev 3



- The Arduino was developed in Ivrea, Italy, the home of King Arduin 955 - 1015.
- Complete single-board microcontroller:
  - ATmega328 controller chip
  - 14 digital I/O pins (6 PWM outputs)
  - 6 analog input pins (10-bit ADC)
  - 32kB memory / 16 MHz clock
  - Hardware & Software UARTs for serial communication
  - Easily expandable with plug-in "shields".
  - Mutiple sources for Arduino andg shields.
- Programmed in C++. Free development tools. Excellent tutorials.
- Vast, helpful user community.
- Cost less than $20.

Touch Screen Display





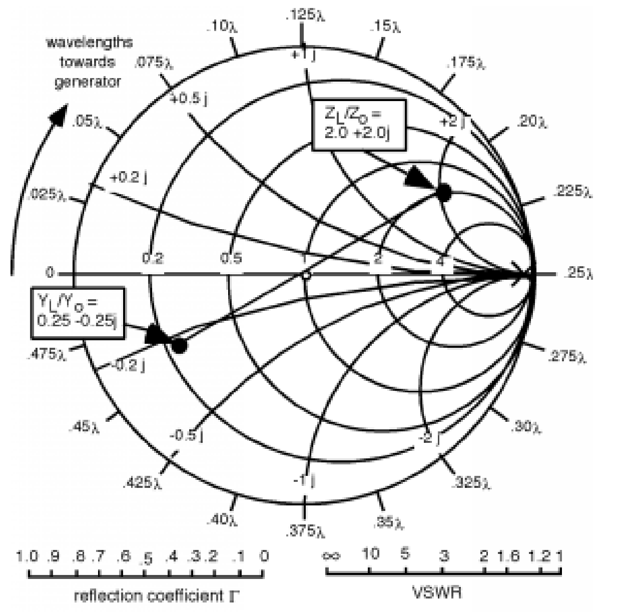
- 2.6-inch diagonal TFT LCD Display
- 16-bit (65K) colors
- Integrated Resistive Touch Screen
- Micro SD card slot
- Arduino shield format
- Cost less than $20 on eBay

Project Status

```
void autoScale(float uMin, float uMax) {
  // calculate divisions between asMin and asMax
  // and set them to a 1, 2, 5 range
  // asMin, asMax, and asDiv are global variables
  float dm;   // decimal multiplier
  int divisor, numDivisions;
  int divisors[] = {1, 2, 5};     //graph scale intervals
    for (dm = 0.001; dm < 1000000 && (numDivisions < 6 || numDivisions > 12); dm *= 10) {
    for (int j = 0; j < 3 && (numDivisions < 6 || numDivisions > 12); j++) {
        divisor = divisors[j];
        asDiv = divisor * dm;
        asMin = int(uMin / asDiv) * asDiv;
        if (uMin < 0) asMin -= asDiv;     //drops asMin to next lower division
        asMax = int((uMax / asDiv) + 0.999) * asDiv;
        numDivisions = (asMax - asMin) / asDiv;
      Serial.print(dm); // results are wrong unless this line is included - IDE bug?
    }
  }
} // autoScale
```

- Auto-scaling chart - working
- Touch screen response – working (needs development)
  - Will use object oriented programming to ease layout of control buttons
- Control of miniVNA – under development – it is a simple 3-wire RS-232.
- Battery power supply – not started – will use Lithium Ion battery with step up switch mode converter.

Long Term Goals



- Add measurements of R, X, L, C.
- Add Smith Chart display.
- Distant goal of replacing the miniVNA with an Arduino-based Vector Network Analyzer.
- Follow progress at www.W4KRL.com

- Become an AMRAD member and join in the technical advancement of amateur radio.
- www.amrad.org

C++ Button class
(a work in progress)

C++ (pronounced "see plus plus") is a statically typed, free-form, multi-paradigm, compiled, general-purpose programming language. It is regarded as an intermediate-level language, as it comprises a combination of both high-level and low-level language features. Developed by Bjarne Stroustrup starting in 1979 at Bell Labs, it adds object oriented features, such as classes, and other enhancements to the C programming language. Originally named C with Classes, the language was renamed C++ in 1983, as a pun involving the increment operator.

C++ is one of the most popular programming languages and is implemented on a wide variety of hardware and operating system platforms. As an efficient compiler to native code, its application domains include systems software, application software, device drivers, embedded software, high-performance server and client applications, and entertainment software such as video games.

From Wikipedia, the free encyclopedia

```
class Button {
 public:
  Button();
  // contructor - TBD
  void setButton(int x, int y, int width, int height, unsigned int color);
  // place a button on the screen with upper left corner at x, y with
     width w, height h, and color
  void updateColor(unsigned int color);
bool isPressed();
  // true if button is pressed

 private:
  Point p;                              // place where user is pressing
  int _x, _y, _w, _h;                   // button coords & dimensions
  unsigned int _c;                      // button background color
  int _tXmin, _tXmax, _tYmin, _tYmax, _tZmin, _tZmax;  // touch plate
     ADC readings
  int xo, yo;
};

void Button::setButton (int x, int y, int width, int height, unsigned int
     color) {
 _x = x;
 _y = y;
 _w = width;
 _h = height;
 _c = color;
 lcd.fillRoundRect(_x, _y, _w, _h, 5, _c);          // radius 5
}
```

Useful Resources

Join the local AMRAD members every Saturday 12:30pm at Tippy's Taco House in Merrifield for wide ranging discussions of all Amateur Radio topics.

Technical meetings every first Thursday 7:30pm at the Dolley Madison Library in McLean.

- www.arduino.cc – the official site. Helpful community of users.
- www.adafruit.com – Limor "Ladyada" Fried's site with tutorials and products. See also –
- www.ladyada.net/learn/arduino - clear tutorials for the hardware and software.
- www.cplusplus.com – tutorials and example code.
- eBay.com – search for "Arduino shield" for may low cost products.
- Micro Center – Fairfax – wide selection of Arduino boards, shields, books and many useful parts.
- Radio Shack – Most stores carry Arduino starter kits and a selection of shields.