The Arduino Graphics Interface

By Ed Andrews WA9UQN

Turning an Arduino Due into a Vector Graphics Display

Part 1: Design Concepts and Hardware Fabrication

The early days of electronic computers used teletype machines and line printers, and later, alphanumeric screens and keyboards as their primary input-output devices. As computer usage grew beyond census and accounting applications, the need for improved graphical displays and output devices became apparent.

Graphical display terminals using large screen cathode ray tubes (CRT) soon evolved. These displays were essentially "up-sized oscilloscopes" whose X-Y beam deflection was driven by a computer.

Figure 1 shows a specialized XY CRT terminal that was part of SAGE (Semi-Automatic Ground Environment): an early air defense system deployed in the US. By the late '70s, Tektronix, Hewlett-Packard, and others sold general-purpose XY graphics terminals able to show graphs, charts, and line drawings of all types.

While a far cry from what we take for granted today, these were important stepping-stones in the evolution of computer graphics and display technology.

FIGURE 1. SAGE air defense system used an XY interactive graphics terminal.

Arduino Graphics Interface

As an undergrad electrical engineering student some eons ago, I worked on a MODCOMP (Modular Computer Systems, Inc.) 16-bit minicomputer that featured a large 19" Hewlett-Packard XY graphics display screen. A few months ago, I began to wonder: *Would it be possible to replace those half dozen racks of circa 1975 minicomputer hardware with one of today's $50 credit-card sized computers?*

The answer proved to be YES! What resulted, I call the **A**rduino **G**raphics **I**nterface, or ***AGI*** for short. With AGI, you can transform a leftover analog oscilloscope into a high resolution computer graphics display and gain valuable insights into computer graphics, digital-to-analog conversion (ADC), and advanced Direct Memory Access (DMA) hardware and software techniques. It's easy to add a real time clock (RTC) module and interactive controls to turn the AGI platform into a "CRT CLOCK" (**Figure 2**) or other high resolution computer graphics display device.

In this article, we'll focus on the overall concepts, circuit design, and hardware fabrication. In Part 2, we'll integrate the hardware to ***XYscope***: the software control and plotting library that enables an Arduino Due CPU + analog oscilloscope to become a high resolution XY graphics display. *Let's get started!*

AGI Block Diagram & Operational Overview

The block diagram of the AGI is shown in **Figure 3**.

You can see that a pair of digital-to-analog converters (DACs) inside of the ATMEL SAM 32-bit CPU (the heart of an Arduino Due) is used to drive the X and Y axes of an analog oscilloscope. Rather than use the normal oscilloscope Amplitude vs. Time display mode, we run the scope in X-Y mode. That is, the internal oscilloscope time base that usually drives the X axis is *not used*, but rather all XY information for deflecting the CRT beam comes directly from the Arduino through the AGI circuits.

To create graphical images with AGI, the programmer need only build a list of 12-bit X-Y integers that make up the individual points we want to show. One after the other, the DACs of the Due receive and convert each XY integer coordinate pair into small X-Y voltages that drive the CRT beam about the oscilloscope screen. The block diagram shows that we also provide a blanking pulse (a.k.a., Z axis blanking) as a third signal to the scope.

As with all CRT type displays, one must repeatedly send the whole display list to the screen to keep the points visible and refreshed. Even though the points are actually flashing at a very high rate, human Persistence Of Vision (POV) causes the displayed image to appear stable and flicker-free.

FIGURE 2. AGI can become a CRT CLOCK!

Key AGI Features

High Resolution — By using 12-bit DACs, we have a display resolution of 4096 x 4096. This *exceeds* even HD TVs and beautifully renders smoth step-free lines, arcs, and characters.

Easy to Use — To display graphical images, the AGI programmer simply creates a list of XY integers to define the individual points to be plotted. A small background process automatically sends the point-list to the CRT.

Ready to Use — A graphics and control library of simple subroutine calls makes it easy to create and load graphic and alphanumeric content into the XY point-buffer. The AGI Library includes:

- Plot POINT, LINE, RECTANGLE, CIRCLE, ELLIPSE, CIRCLE-ARC-SEGMENT, ELLIPSE-ARC-SEGMENT
- Plot CHARACTER, TEXT_STRINGS, INTEGER NUMBERS, FLOATING POINT NUMBERS

Complex Graphics Possible — The display list can contain up to 15,000 X-Y points so that complex objects can be rendered.

Fast — The Due is a very fast processor that can very quickly calculate and store points into the XY display list array. There's plenty of processor time available to make changes in the list, making animated graphics easy.

Educational — The AGI is a great introduction to graphical coordinate systems, X-Y point/vector style computer graphics, and DACs.

AGI is also a solid demonstration platform to learn

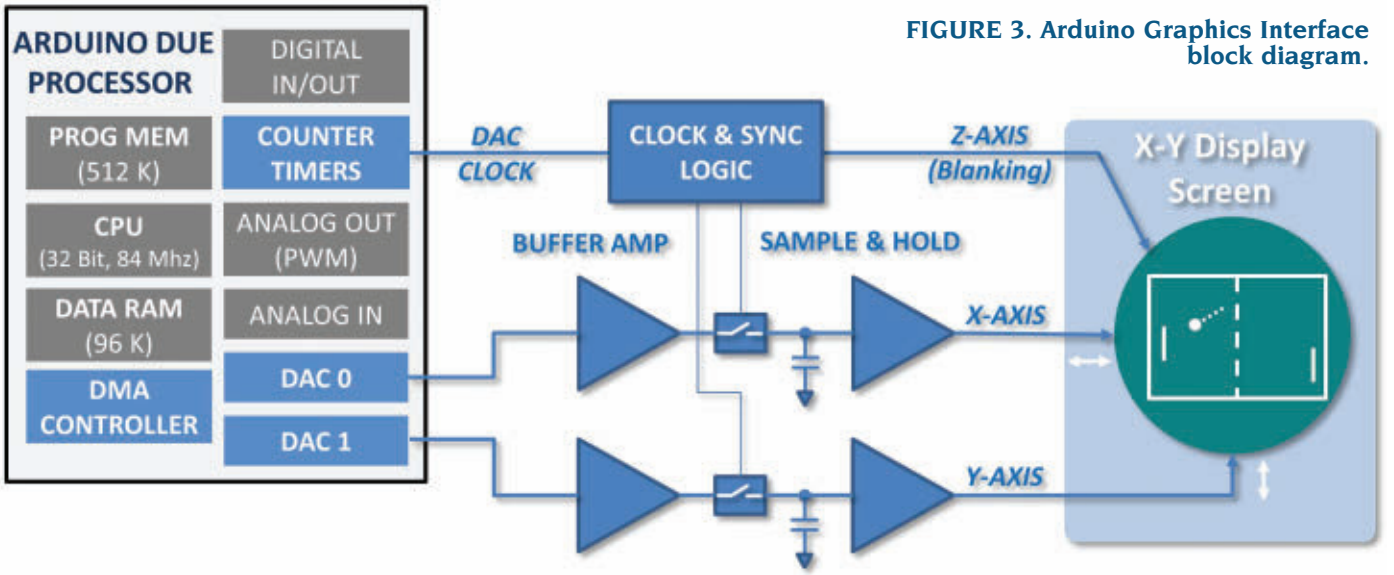

FIGURE 3. Arduino Graphics Interface block diagram.