

QUADD (Quite Useful Arduino Debugging Device) initial ideas

The following is a (wish)list of features I think would be useful for a quite useful Arduino debugging device.

Note that this is a fairly ambitious collection of features and I'm not 100% sure I can pull it all off but it should be doable.

If you are interested in this either as a co-developer or potential user please get in touch, rob@robgray.com

I2C

- Non-intrusive monitoring of a connection at 100kHz, 400kHz or 1MHz
- Storage of data for protocol analyser
- Up to 64k buffer depth
- External connection from headers

SPI

- Non-intrusive monitoring of a connection at rates up to 10MHz
- Monitor MISO and MOSI, gated by SS or not. If multiple slaves connect the user inputs to the SS signals then post processing will display in a more readable form
- Up to 64k buffer depth
- Storage of data for protocol analyser, split according to the active SS
- External connection from headers

UART

- Non-intrusive monitoring of connections at rates up to ~1Mbps
- 4 x serial inputs (2 full-duplex serial links, 4 independent signals, or other combinations)
- Up to 64k buffer depth
- Storage of data for protocol analyser
- External connection from headers

Digital capture channels (logic analyser)

- 27 bits – 20 standard Arduino IO pins plus 7 “user” inputs
- Up to 64k buffer depth
- 20MHz sample rate (possibly faster)
- The 20 Arduino inputs are hard-wired connections to Arduino backplane, 7 user inputs from headers

Oscilloscope

- 6-channels connected to Arduino analogue pins
- 12-bit sampling up to 400kHz (1ch) less for multiple channels
- Use with digital channels for mixed-signal recording
- Up to 64k buffer depth
- Hard-wired connections to Arduino backplane

All the above can be sampled concurrently (probably not at full speed, TBD) and the data interleaved so you can see the relationships between for example SPI and an analogue signal. The 64k buffer is shared and used for data from all sources.

Buffer could be expanded to 512k but this would cost more, maybe an add-on board?

Trigger

- Analogue value on oscilloscope (edge sensitive ?)
- Any combination of bits/edges on 27 logic analyser channels
- Data pattern on any serial port
- Any combination of the above
- Adjustable number of pre/post trigger samples
- Qualified by time trigger is valid
- Trigger output for other test equipment
- Delayed trigger
- Possible sequencing

Frequency counter

- Count, frequency, period, pulse width, duty cycle, servo position
- Min, max, average of all above
- Simple spectrum analysis
- Digital only

Analogue waveform generator (AWG)

- 10 bit resolution
- 1MHz bit rate
- Useful sine wave freq about 40kHz
- No filtering so stepped waveforms
- Any waveform length up to 64k
- Trigger output at start of waveform
- Replay sampled waveform
- Use as variable DC output

Digital pulse generator (DPG)

- 1 bit
- Driven by 32-bit PWM for high speed repetitive signal
- Any duty cycle and speed up to ~20MHz
- Use as an arbitrary single-bit waveform generator

NOTE: With regard to all the above capabilities, there may be limits on what can be done at the same time, TBD.

Logic probe

- Visual and audible
- Pulse catching
- Test for signal driven high, driven low, pulled high or floating

The logic probe and AWG share hardware so only one can be used at a time.

Software debugging

- Built-in AVRmon support for various Arduino processors allows viewing and modifying memory and IO plus other control functions and low-intrusive viewing of some variables
- High-speed SPI interface for the target system's program, use SPI or shiftOut on the target Arduino to print data to screen
- Serial debug UART, any value received printed to screen. Use (New)Softserial or small debug serial function

Display control software 1

Simple dumb terminal interface using any standard terminal program.

Display control software 2

PC-based app

Data displayed in all expected formats

Written in VB.NET? This will be a very large project in its own right, hence the above dumb terminal version to get things started

IO voltages

All inputs 5v tolerant

All outputs 3v3 or 5v selectable

Block diagram

