4099MEE IAP2 planning report

Title: **Low-cost USB MIDI controller**

Date: 20. Feb. 2004

Version: 0.3

Name: Paul Harvey

GU ID: 1733426

Table of Contents

1. Introduction

Analogik (http://www.analogik.com) is a Brisbane based electronic music and multimedia group. They have recognized a market for, and require the design, of a low cost Musical Instrument Device Interface (MIDI) controller. Analogik hopes that at conclusion of this Industrial Affiliates Program 2 (IAP2) subject, a working prototype from which a production version can be derived will be created.

There are four main persons involved with this project. Paul Harvey (Project leader/developer), Dr. Yongsheng Gao (University supervisor), Dr. David Rowlands (Technical supervisor), Dejan Petrovic (Representative from Analogik, may perform final independent testing/verification).

This project has already had some work done prior to the beginning of Industrial Affiliates Program 2 (IAP2). This has consisted primarily of research into MIDI technology, suitability of various Micro Controller Units (MCUs), and Universal Serial Bus-First In First Out [buffers] (USB-FIFOs) Integrated Circuits (ICs). See Appendix A3 for more details.

A MIDI controller is used by electronic music enthusiasts and professionals to assist in the composition and editing stages. Generally, a typical unit will have a number of continuous controls such as knobs and sliders, a few octaves of keyboard keys, as well as digital "triggers" in the form of buttons. Some units have feedback in the form of motorized sliders, LEDs, and LCDs. Sophisticated models have specialized input devices like pitch wheels and joysticks for sound effect creation. See Figure 2 for an example.

2. Program Justification and Development

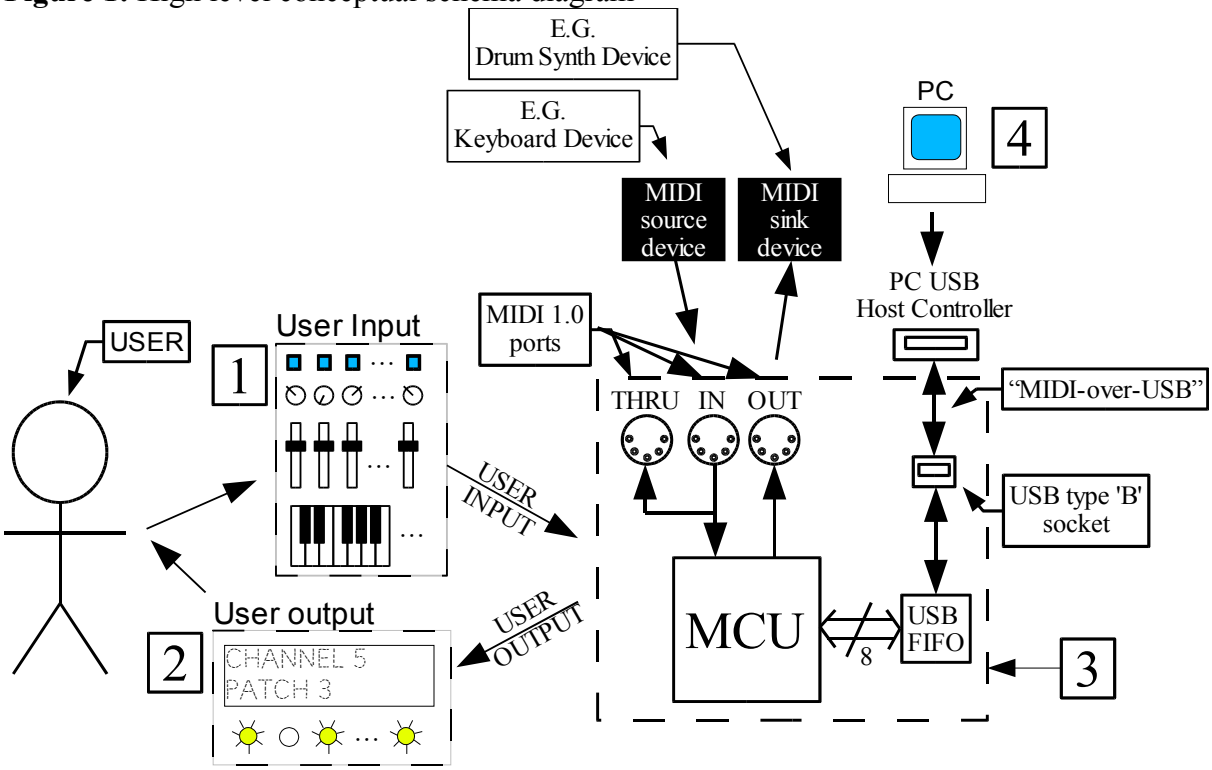
As technology becomes increasingly affordable, powerful and accessible, the public use it more and more for their creative works. As PC software has rapidly replaced all the physical MIDI hardware that traditionally used to do the final synthesis into audio, MIDI controllers are now an integral part of electronic music creation, live performances and studio work.

The cheapest existing product that is almost in direct competition with the Analogik MIDI controller concept is the Evolution UC33E, with 47 knobs/sliders and 14 assignable buttons. Priced at $649 AUD[i], shown in Appendix A2, Figure 4, along with more information.

Analogik hopes to initially sell 50 – 100 units of a device with similar specifications and that is no more expensive than the UC33E. If necessary, a production unit for Analogik may incorporate an extra feature such as 16 bit audio line in/out ports. However, this is beyond the scope of the IAP2 project.

The minimum deliverables are described in Section 8, Technical Specifications - "Minimal Outcomes". A working demonstration to Dejan Petrovic and/or Dr. David Rowlands would confirm the completion of the minimum deliverables.

Figure 1: High level conceptual schema diagram



NB: USB, MIDI 1.0 IN/OUT are all digital serial-bus type interfaces.

Figure 1 above shows a simplistic data flow diagram of a typical MIDI controller setup. The user interacts with the device using input controls such as knobs, sliders and buttons (block 1), and also monitors output in the form of an LCD and status LEDs (block 2). For actual input/output specifications, see Section 8.1. This I/O is relayed via the MIDI controller circuitry (block 3, where the entire scope for this project lies) to a PC running MIDI capable software (block 4) via USB, "MIDI-over-USB". The MIDI source/sink blocks demonstrate the ability for the MIDI bus to be expanded with more MIDI devices. The MIDI 1.0 bus traditionally connects via a legacy 15 pin d-sub gameport connector on a PC with an adapter, however, modern laptops and PCs often do not have this so USB is used instead.

Project goals

1. Create a prototype low-cost MIDI controller, where at the conclusion of this IAP2 project a functioning PCA to which the specified user I/O (See 8.1) will connect as test jigs
2. A MIDI IN port will be implemented to which a MIDI source can be attached (see Fig. 1)
3. A MIDI OUT port will be implemented to which a MIDI sink can be attached (see Fig. 1)
4. A USB port will be implemented to which a PC will be attached, communicating MIDI messages (See Fig. 1, block 3-4 connector)
5. Actual form factor of input panel controls is not within the scope of this project – ergonomics, economics, and manufacturing issues need to be addressed. Hence the usage of test jigs for user input/output – they are not to be considered part of the assessed project.

Currently, the HC9S12A64CFU MCU from Motorola has been chosen, as has the FTDI FT8U245AM USB FIFO/UART IC. See Appendix B: Current Design decisions for justification and driver software issues.

3. Program Schedule

Refer to Gantt chart in Appendix C. All work is to be carried out on a standard PC workstation under the GNU/Linux platform with Internet connection. Even if not listed, this resource is assumed to be used throughout the entire project.

Work breakdown – major milestones are marked in bold

Week Item

0. Research. Items:

- MIDI 1.0 specification must be acquired through library or purchased from the MIDI manufacturer's association
- Further investigation into USB technology and MIDI over USB
- Develop block diagram of all significant components in the MIDI controller

Resources required:

- *Complete MIDI 1.0 Detailed Specification v96.1 (second edition)* (2001); available from library OR the MIDI Manufacturer's Association at http://www.midi.org/about-midi/specinfo.shtml for US$19.00 per book

1-2. Items:

- Design and develop HC12 RTOS firmware using simulator
- Implement input reading and LCD routines using simulator
- Start the design of "full" (all I/O present) schematic, decide on which MCU pin will do what

Resources required:

- PCB CAD software; University's copy of Protel (NB: voids use of PCB design for commercial purposes)

3-4. Items:

Milestone: Implement firmware on a real HC9S12DP256 evaluation board

- Test code should show evidence of working input routines on debug port
- Add (breadboard) MIDI ports, LCD
- Set up a testing environment for MIDI.

Resources required:

- HC9S12 evaluation board (provided by Paul Harvey)
- Breadboard, MIDI DIN sockets, MIDI -> legacy PC game port adapter, cables, opto-coupler, misc. components (may be provided by Paul Harvey)
- Workbench space, CRO, lab PSU, multimeters (supplied by the University)

5-6. **Firmware milestone: LCD, MIDI 1.0 IN and OUT ports functional**

- Correct MIDI messages via legacy gameport generated in response to input
- LCD shows (hex) controller ID and value of last potentiometer change

Project Confirmation

7-9. Items:

- Add all input circuitry to breadboard circuit
- Create a FT8U245 PCB test jig (to adapt QFP IC for breadboard use)
- Develop & test input scanning firmware routines

Resources required:

- Test jigs for: 64 LEDs, 64 buttons, 16x2 LCD (provided by Paul Harvey)
- 2x FT8U245 USB FIFO ICs; supplied by Gigatechnologies, Gold Coast. See http://www.gigatechnology.com/chipprod.html. Cost: $10 AUD.
- Misc. components, incl. USB type 'B' socket
- University's PCB fabrication services for FT8U245 test jig
- University's copy of Protel PCB CAD software
- Workbench space; Software: VMWare Workstation 4 (provided by Paul Harvey) – USB driver development

10-11. Items:

- Finalize hardware/schematics – INCLUDING choosing one model of HC12
- Design PCB
- **Have PCB fabricated**

Resources required:

- University's PCB fabrication services (est. $70) OR
- BEC Manufacturing Brisbane (http://www.becman.com) offers new jobs, bare/double sided, 10”x16” panels for $202 AUD ex GST.
- University's copy of Protel PCB CAD software

12-13. **Firmware/driver milestone: MIDI over USB functional**

- Correct MIDI messages via USB in response to test input

Resources required:

- Workbench space

14-15. Items:

- Assemble, debug, test.
- Prepare prototype for verification
- **Demonstrate working; verification of project minimum deliverables**

Resources required:

- Workbench space incl. CRO
- Technical supervisor and Dejan Petrovic for demonstration purposes

4. Initial Cost Estimate

- **Components -** Although the MCU selected so far is priced at ~ $10 ex GST, Arrow electronics has a minimum order of $100 + shipping. An alternate supplier or a co-ordinated purchase with others at the University may be required.
- In total, judging from the PICmicro based original prototype, total component costs excluding MCU/USB FIFO may approach $100 (See Appendix D: Early PICmicro prototype component list). For a full test input panel to be created, the cost will be more.

Total: $200 approx.

- **PCB -** There are two SMT ICs to be loaded onto the final PCB. This may not possible with the University's PCB workshop. In this case, at least $224 should be allocated for manufacturing at BEC.
- An SMT soldering/rework station may be necessary to install the SMT ICs.

Total: $300 approx.

- **Labour (in-kind) -** Project manager (Paul Harvey) $15/hr x 50hrs/wk x 12 wks = $9000
- Technical Supervisor (Dr. David Rowlands) $120/hr x 0.5hrs/wk * 12wks = $720
- Academic Supervisor (Dr. Yongsheng Gao) $120/hr x 0.5hrs/wk * 12wks = $720

Total: $10440 approx.

Total costs of creating the prototype should be within the $900 offered by the University for this IAP2 incubator project. Any costs exceeding this amount will either result in re-negotiation of the project plan or be covered by the project manager (Paul Harvey).

5. Assessment of Potential Benefits

- Direct cost savings due to donation of labour (>$10,000)
- Cost savings via usage of University laboratories (free use of $1k's worth of test equip.)
- Prototype savings via University resources including academic licensed CAD software (Protel 2004 is >$10,000 RRP) and will provide solid starting point for production version

6. Risk Assessment and Ethical Considerations

Issues affecting project completion:

- **Drivers** - The 2.6 Kernel may have an unforeseen problem due to it being a newly released kernel; emulation of existing controllers may not be feasible, taking more work
- **USB -** The FT8U245 may have some limitation preventing it working MIDI over USB
- **PCB** – Delays in fabbing. the PCB may occur; it may be defective or designed with errors
- MIDI over USB implementation may have latency issues making MIDI OUT inadequate
- Ordering cheapest MCU that meets requirements exactly could take longer than expected
- The MCU ordered has some difference that causes it to behave differently with same code

Ethical risks – risks to society and/or the company

- This is a consumer device that is to be powered with an external 3rd party PSU; there is little chance this device could physically harm a human being
- Could cause expense to Analogik if a production had to be recalled due to a design flaw

Patent and Intellectual Property issues

- A device advertised to be USB/MIDI compatible must undergo USB consortium and MIDI Manufacturers Association registration, which must pass expensive "Logo" testing.
- It is believed that apart from USB and MIDI, the technologies used in the prototype are royalty-free. However without doing a patent search it is difficult to be 100% certain.

7. Control Procedures

Changes to the final deliverables and/or the approach to achieve them will be documented in this report. Whilst a version number and a date will identify which version of the planning report a certain copy may be, all changes will be recorded in "Appendix Z – Changelog". Each change must be approved by all parties mentioned in the introduction of this report. If a major schedule slippage has occurred, and the final delivery deadline will be missed, the action mentioned under "Failsafe Notice" in Section 8, Technical Specifications may be used.

8. Technical Specifications

Connections

1. Standard MIDI 1.0 IN/OUT/THRU ports

   - Mechanical: 5 pin DIN sockets

   - Electrical: 5mA opto-isolated current loop, 31250 bits per second as specified by MIDI 1.0. "Voltage" (non-cable) side of the opto-coupler should have test points on the PCB.

2. USB 1.1 port, as provided by FTDI 245 USB FIFO IC

   - Mechanical: USB type 'B' socket (peripheral device socket)

   - Electrical: NRZI, 3.3V differential 1.5Mbps as specified by USB 1.1 Low-Speed specification and supported by the FTDI FT8U245 USB-FIFO IC. Testpoints for D- and D+ should be provided on the PCB.

3. Power input 9V DC

4. TTL debug I/O port header (a TTL <-> RS232 adapter would be used to connect a PC)

User-input/output – See Appendix F2 for test jig setup details

1. 64 digital push-buttons (8x8 scanned matrix, debounce in firmware) and 64 LEDs

   Of those 64, four will be reserved for LCD menu navigation buttons: left, right, enter, esc. A further 16 will be reserved for patch changes (input association maps). The rest will be assignable MIDI triggers.

2. 48 potentiometers configured as voltage dividers (linear; can be either knobs or sliders. MIDI 1.0 allows for "coarse" controllers to have 127 levels, "fine" to have 16,384 levels)

3. **Table 2:** Input Assignments (over page)

Of the 48 knobs/sliders, the following standard controller types will appear:	Of the 64 trigger buttons, the following standard triggers will appear:
0 Bank Select (coarse) 1 Modulation Wheel (coarse) 2 Breath controller (coarse) 4 Foot Pedal (coarse) 5 Portamento Time (coarse) 6 Data Entry (coarse) 7 Volume (coarse) 8 Balance (coarse) 10 Pan position (coarse) 11 Expression (coarse) 12 Effect Control 1 (coarse) 13 Effect Control 2 (coarse) 16 General Purpose Slider 1 17 General Purpose Slider 2 18 General Purpose Slider 3 19 General Purpose Slider 4 32 Bank Select (fine) 33 Modulation Wheel (fine) 34 Breath controller (fine) 36 Foot Pedal (fine) 37 Portamento Time (fine) 38 Data Entry (fine) 39 Volume (fine) 40 Balance (fine) 42 Pan position (fine) 43 Expression (fine) 44 Effect Control 1 (fine) 45 Effect Control 2 (fine)	64 Hold Pedal (on/off) 65 Portamento (on/off) 66 Sustenuto Pedal (on/off) 67 Soft Pedal (on/off) 68 Legato Pedal (on/off) 69 Hold 2 Pedal (on/off) 80 General Purpose Button 1 (on/off) 81 General Purpose Button 2 (on/off) 82 General Purpose Button 3 (on/off) 83 General Purpose Button 4 (on/off) 120 All Sound Off 121 All Controllers Off 122 Local Keyboard (on/off) 123 All Notes Off 124 Omni Mode Off 125 Omni Mode On

4. 16x2 character alpha-numeric LCD (as provided by HD44780 compatible LCD), to display MIDI channel the controller is on at startup, and the ID + value of the last moved control.

Figure 2: A concept 3D drawing of a MIDI controller

Illustration 1 (C) 2002 Dejan Petrovic

Approx. dimensions of theoretical production unit: 400mm x 250mm

The final arrangement of these inputs is subject to change, since enclosure design (probably at least dust proof) and other production issues have not been considered. However, it is trivial to change input assignments; the controller ID is merely modified in the firmware code. Additionally, the MIDI software package can assign any arbitrary MIDI input to any arbitrary action in the program.

Firmware

1. Scan all controller input and generate correct MIDI messages on the MIDI OUT port and IN USB endpoint in response to changing input
2. Set the state of each LED as necessary either by responding to MIDI IN or OUT USB endpoint messages, or the toggle state of an associated digital push-button
3. Respond correctly to MIDI IN messages or OUT USB endpoint. That is, forward messages on to the MIDI OUT port if necessary, respond to MMC codes, reset requests, and so on.
4. Provide a facility utilizing the LCD and some of the digital push-buttons to implement a menu system to configure the unit's MIDI channel number, edit a controller mapping and loading controller mappings.
5. Control and maintain the USB FIFO IC
6. Control and maintain the LCD

Linux kernel driver

1. Allow a user-land application to receive MIDI messages given an appropriate /dev/midiXX node which would be registered with the driver
2. Allow the user-land application to send MIDI messages to the MIDI controller, such as program changes, sync and timing information

8.1 Minimum outcomes

1. Each potentiometer input reflects a full-scale linear response in the MIDI output it produces. Further, each potentiometer input produces output for a unique MIDI control.
2. Each digital push-button causes either a unique MIDI trigger or patch change response
3. Each LED either reflects the toggle-state of a MIDI trigger associated with a button or identifies which patch is selected
4. The LCD displays the current MIDI channel the unit is on during power-on, and the hex ID and value of the last moved controller input is shown, showing that the LCD is functional.
5. All of the above functions when a PC is connected via the legacy MIDI 1.0 ports, demonstrating MIDI firmware functionality

8.2 Additional outcomes

1. All of that described in 8.1 above functions when a PC is connected via USB, demonstrating USB circuitry and driver functionality.
2. **Failsafe notice:** If the project is two weeks or more behind at around week 8, USB firmware/driver development should stop and work should concentrate on getting the PCB design completed on schedule. Even then, there is possibility that an untested USB implementation may still work after the PCB has been fabricated and assembled.

9. Quality Schedule

This project will be considered successful for the purposes of the purposes of IAP2 if:

1. A working prototype is developed, and approved as per "8.1 Minimum outcomes".
2. The prototype is completed by the required date, and
3. The prototype is useful as a starting point for further commercial development.

Generic useful development practices will include logging changes made to each project file; storing each version as they are modified; and updating this project plan as new problems and circumstances arise. Specific best-practice engineering methods include, for C coding - running code past a code checker (such as lclint); for PCB design, setting each design rule possible to known fabrication parameters and have the CAD package check the design; for code performance, an ELF binary profiling tool has already been created ("bytewise.pl") that shows a function-by-function, file-by-file breakdown of code space usage in a C project.

Further with PCB design, general prototype design practices will be used to ensure maximum "correctability" of the first PCB. Jumper blocks or solder pads can be used to isolate certain circuits easily. Test points can be created on critical I/O lines and other signals such as the MCU CLK pulse. If at all possible, signal lines traveling exclusively in the top layer will avoid the underside of DIP ICs. Also, as a matter of component flexibility, component footprints that cater for more than one pin-out profile should be used (eg. capacitors, switches, sockets).

Verification and Validation Plan

All five milestones in the project plan can be independently verified. For a complete test plan, see Appendix F. An outline is listed below:

Wk Milestone

3. **Implement firmware on a real HC9S12DP256 evaluation board**
   - Show debug output in response to test input such as slider movement
4. **Firmware milestone: LCD, MIDI 1.0 IN and OUT ports functional**
   - Show MIDI software on PC-side:
     - Respond linearly with full range on each of the potentiometer inputs
     - Respond to digital triggers with each of the buttons
   - Show breadboard-prototype respond to MIDI events from PC-side software or patch changes/trigger button toggle states with LEDs
   - Show CRO waveform indicative of correct bit rate and 5mA current specification
   - Show LCD displays controller ID and value of last potentiometer to have moved
8. **Firmware/driver milestone: MIDI over USB functional**
   - Show the same points verified in wk. 4 using USB-over-MIDI ONLY
   - Show CRO waveform indicative of USB MIDI OUT messages present on

breadboard-prototype's legacy MIDI 1.0 port

10. **Have PCB fabricated**

12. **Demonstrate working; verification of project minimum deliverables**
    - Show the same points verified in Wk. 4 & Wk. 8 for final prototype PCA
    - Confirm all points outlined in Section 8.1 satisfied

At each validation point, the supervisor may operate the controls and/or observe the PC-side software, LCD, and LED activity for themselves. However, it is expected that the bulk, exhaustive testing of each input will be performed by the project manager/student. Each milestone is to be demonstrated to the technical supervisor, except for the week 12 (final) demonstration which may also be demonstrated to Dejan Petrovic.

10. Wider Engineering Issues

Marketing

Analogik believes that as part of the marketing push, the sourcecode to the firmware and drivers could be released to the public under the GNU GPL. Such licensing may provide free additional publicity, and also guaranteed opportunity for updates if Analogik drop support. Copyright will still always reside with the author (Paul Harvey) unless specifically transferred in writing. However, there is a risk that a competitor will pick up the firmware and use it in their own directly competing product (legally, of course – with restrictions). By the same token, any improvements made to the codebase must be made accessible and hence could be reclaimed back into the code distributed with the Analogik product.

Design for manufacturability

Unfortunately with a consumer device such as this project, a great deal of effort must go into sourcing of proper components (of suitable quality, cost and ease of assembly). This may affect the interfacing circuitry in the final electronic design and the enclosure setup. Also, enclosure design and custom plastics for the various controls may be necessary. Development and tooling for these activities is extremely costly, far more so than the electronic design aspects. In other words, PCB layout would probably be in the final stages of a production version's development cycle.

Product support

Although not directly an ethical issue, many consumers become emotional when they receive less than satisfactory support for their product after purchase. Warranty and general support is perhaps a large hidden cost in any device when the Recommended Retail Price (RRP) is being calculated.

11. Learning Contract - Skills expected to be gained during this project:

- USB internals, driver development, MIDI internals, PCB design with SMT components
- Advanced C coding with the HC12 MCU, Linux Kernel internals
- Ability to judge complexity and work involved in a given engineering task

Appendix A: MIDI technology summary

MIDI controllers often connect to a PC or other MIDI devices via MIDI 1.0 ports. Many MIDI controllers offer USB connectivity to a PC, since USB is cheaper than MIDI cables, plugs and adapters.

Inputs from the various controls on the device are sent to a computer software package running on a PC for interpretation using the MIDI protocol. Although there is a set of standard MIDI control/message types declared by the General MIDI specification that the device can assign to each input, most software allows arbitrary mapping of any MIDI input to any function the user desires. Such mappings may control stereo panning, volume, tempo, pitch, reverb, echo, or any other effect for any number of tracks running simultaneously. As for triggers and note keys, possible functions include: normal notes, MIDI Machine Control commands (such as stop, play, pause), program changes, and patch changes. A MIDI controller may even be used to control non-musical devices, such as stage lighting and smoke machines. For most users, controlling all these parameters entirely with a standard keyboard and mouse can become tedious, hence the MIDI controller.

MIDI is a real-time serial protocol used to communicate music in an instructional form, IE. parameters and attributes of notes, programs, tracks, and higher-level musical control. MIDI 1.0 does not carry any samples or actual raw sound for any of the notes; rather, MIDI can be thought of as electronic sheet music. Final rendering of a MIDI stream depends entirely on the MIDI interpreter and configuration of the wavetable synthesiser used to play it.

The MIDI 1.0 Specification, set by the MIDI Manufacturers Association (http://www.midi.org), is a complete standard that includes electrical, mechanical, cabling, timing, protocol, and implementation specifications. The PD-02 aims to provide compatibility with legacy MIDI 1.0 IN/OUT/THRU ports (5 pin DIN sockets, 5mA current loop) and also follow the MIDI-MA recommendations on MIDI-over-USB.

Appendix A2: Current MIDI controller market summary

Figure 4: A competitor's product similar to the Analogik MIDI controller concept

Source: Sound & Music, Melbourne; http://www.sound-music.com/images/evolution/evolution_uc33_1.gif

Most MIDI controllers are designed around a keyboard of two to seven octaves, with various knobs and sliders, pitch wheels and other input devices scattered around the outside. Often, musicians feel that they "don't have enough control". Keyboards with more than 16 knob/slider inputs are generally full sized and quite expensive. Analogik believes there is a market for a controller that is focused on just the input controls, leaving the consumer to stick with their existing, cheaper MIDI keyboard.

There are a number of existing products available, some coming from well established names such as *Korg*, *Roland*, and *Yamaha*. Of these, the product closest resembling the aim of this IAP2 project is the *Korg Microkontrol*, which has an RRP of 275 GBP (~$660 AUD)[ii]. It has 8 knobs, 8 sliders, 16 trigger pads, joystick and a 37 note mini keyboard. Also available is the US-428 from Tascam, with 2x 24bit audio in/out channels, no keyboard, 18 knobs/sliders, various buttons and a jog wheel - priced[iii] at 449 GBP RRP (~$1070 AUD)[iv]

Appendix A3: Reasons for dropping early PICmicro based prototype

The result of this research activity can be seen in an incomplete prototype Printed Circuit Assembly (PCA). Although electrically sound (the MCUs and USB-FIFO are functional and interfaced correctly), the MCU selected was not suitable for the application. The PICmicro 16F series MCUs selected were very tedious to program in their native RISC assembly language, and there was no cheaply available C compiler. Therefore, a completely new design with lower IC count and more developer friendly MCU supported by a freely available C compiler is required.

Appendix B: Current design decisions

Hardware - MCU: HC9S12A64CFU

- Subject to change (quantity of I/O lines, amount of EEPROM)
- 64KiB FLASH, 1KiB EEPROM, 4KiB SRAM
- 25MHz CLK max., 8x A/D channels, 80 pin QFP IC package
- Supports I2C, SPI, SCI serial interfaces; 59 total I/O pins
- $9.80 USD ($12.30 AUD) from Arrow electronics
- Already in possession of two HC9S12DP256 evaluation boards for development

Hardware - USB FIFO/UART IC: FTDI FT8U245AM

- 28 pin QFP IC package
- A Linux driver exists for FT8U232; rights to royalty-free windows driver with each IC
- $10 AUD from gigatechnology.com (located at Gold Coast)
- All USB enumeration, endpoint setup and other USB tasks are performed internally

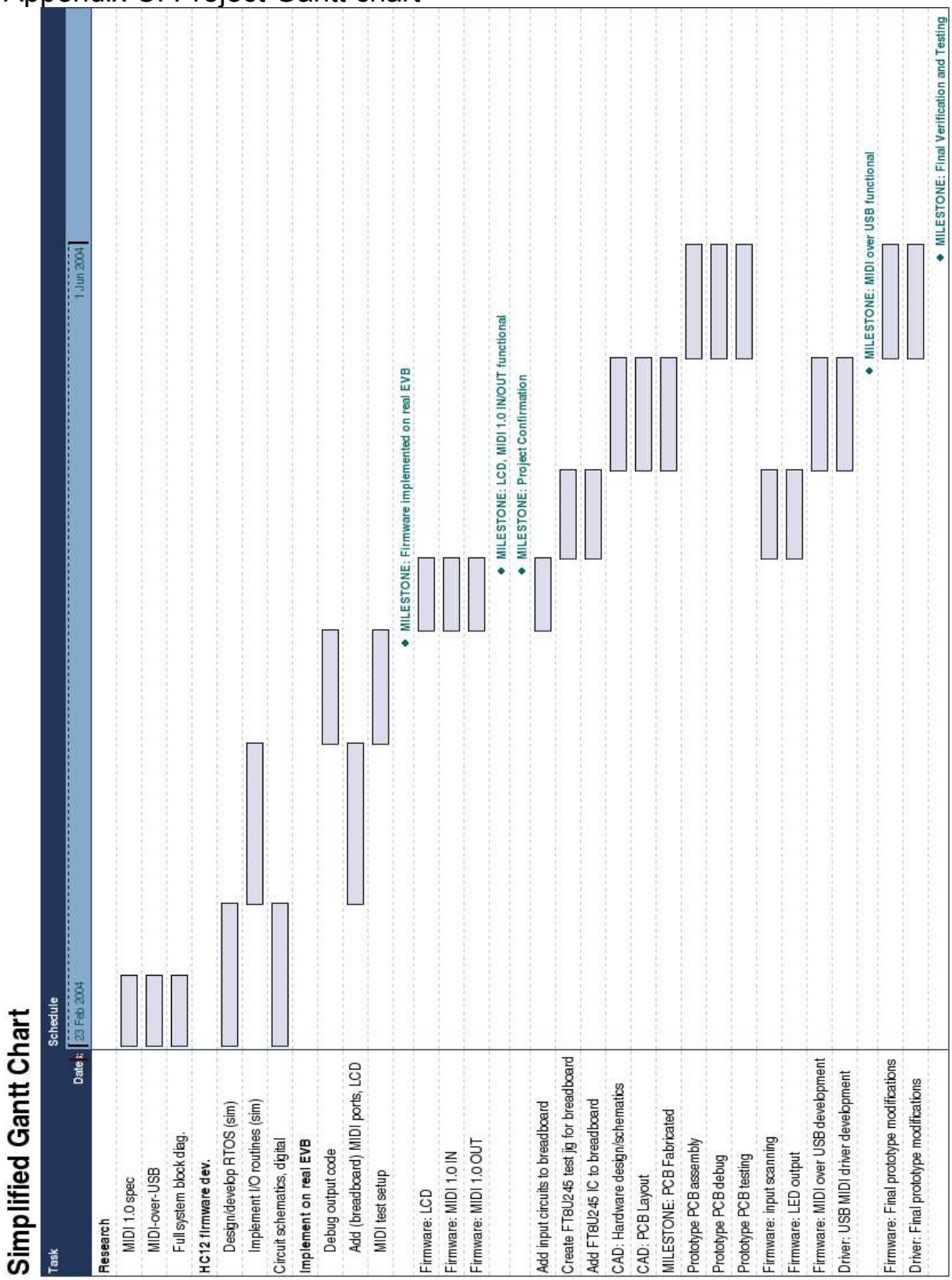
Although the target market will be a mostly Windows XP userbase with some MacOS X, commercial grade consumer driver software is beyond the scope of this project. The easiest and most accessible platform to develop with is Linux. It does not take very long for one to navigate the Linux kernel source and see that, at first glance, there appears to be just three source files associated with interfacing MIDI over USB (See Table 1).

By studying this part of the kernel source, it can be seen that device-specific information relating to existing products on the market are mostly in usb-midi.h, with the exception of a few products that have extra features and/or “quirks” that need work-arounds. It should be possible to build a USB MIDI device with minimal modification to these modules; in fact, by studying the source closely, it may be possible to emulate an existing device so that no changes are necessary.

Table 1: Linux Kernel 2.6 usb-midi driver related files

File	Comment
drivers/usb/class/usb-midi.c	This code actually “understands” MIDI; it appears to be the equivilent of a sound card's MPU401 emulation. That is, responsible for open/close/ioctl of the /dev/midiXX device nodes.
drivers/usb/class/usb-midi.h	Device-specific “magic” (like USB-IF Vendor/Product IDs) go here
sound/usb/usbmidi.c	Handles passing MIDI between ALSA/user-land & USB frames

Appendix C: Project Gantt chart



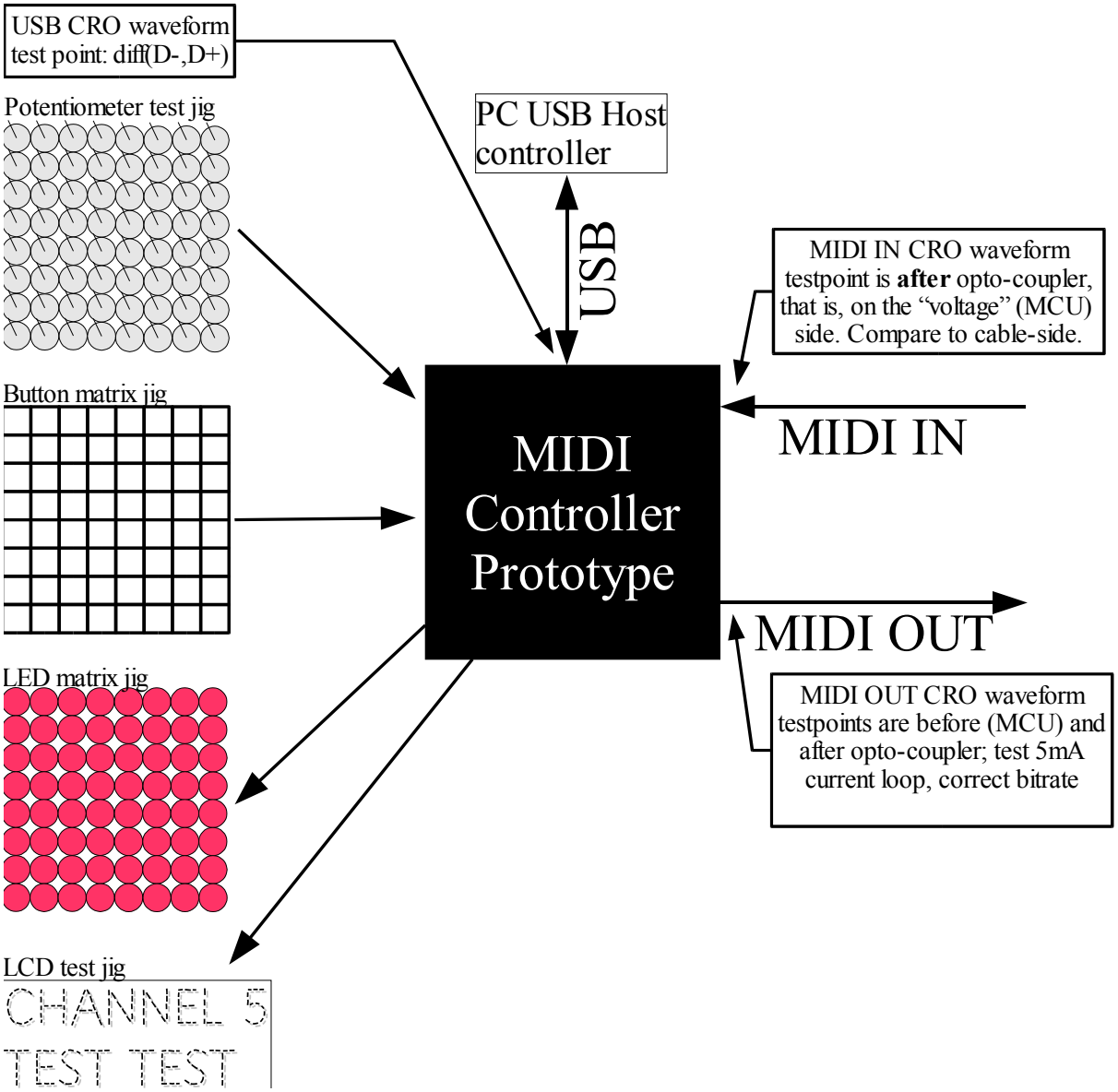
Appendix D: Early Original prototype partial Bill Of Materials

Item	Value	Notes	Package	Cat. No.	Supplier	Cost ea. < 10	Cost ea. < 25	Cost ea. > 25	Cost ea. > 100	Qty. per unit	Qty. per proto	Qty. per prod	Subtotal (proto)
Capacitor	150pF		Ceramic Disc	RC-5326	Jaycar	$0.28	$0.28	$0.19	$0.16	1	1		$0.28
Capacitor	15pF	XTAL	Ceramic Disc	RC-5314	Jaycar	$0.28	$0.28	$0.19	$0.16	2	2		$0.56
Capacitor	1nF	Multivibrator	Ceramic Disc	RC-5336	Jaycar	$0.28	$0.28	$0.19	$0.16	1	0		$0.00
Capacitor	0.1uF		Ceramic Disc	RC-5480	Jaycar	$0.20	$0.20	$0.14	$0.12	1	0		$0.00
Capacitor	1uF		Radial	RE-6032	Jaycar	$0.25	$0.25	$0.17	$0.15	5	3		$0.75
Capacitor	1uF	IC Decoupling	Radial	RC-5490	Jaycar	$0.20	$0.20	$0.14	$0.12	51	30		$6.00
Capacitor	10uF	VR2	Radial	RE-6066	Jaycar	$0.20	$0.20	$0.14	$0.12	1	1		$0.20
Capacitor	1000uF	Power Supply	Radial	RE-6220	Jaycar	$0.70	$0.70	$0.45	$0.40	3	3		$2.10
Diode	1N4004		Axial	ZR-1004	Jaycar	$0.36	$0.30	$0.25	$0.25	1	1		$0.36
Diode	1N4148		Axial	ZR-1100	Jaycar	$0.32	$0.24	$0.17	$0.17	12	1		$0.32
Diode	1N5404		Axial (Large)	ZR-1014	Jaycar	$0.25	$0.18	$0.13	$0.13	1	1		$0.25
Diode	P6KE16CA	(1N5352 - ITVS)	Axial (Large)	ZR-1450	Jaycar	$1.60	$1.40	$1.40	$1.10	1	1		$1.60
Diode	BZX85C6V2	(1N4735)	Axial	ZR-1405	Jaycar	$0.40	$0.35	$0.25	$0.25	2	2		$0.80
Fuse	2A		5 x 20mm	SF-2160	Jaycar	$0.35	$0.30	$0.25	$0.25	4	1		$0.35
Optocoupler	6N138		DIL8	ZD-1940	Jaycar	$4.25	$3.75	$2.90	$2.90	1	1		$4.25
LED	3mm			ZD-1740	Jaycar	$0.25	$0.22	$0.17	$0.17	3	3		$0.75
Resistor	180R	1 - Pack of 8		RR-0654	Jaycar	$0.38	$0.25	$0.22	$0.22	2	2		$0.76
Resistor	220R			RR-0656	Jaycar	$0.38	$0.25	$0.22	$0.22	5	5		$1.90
Resistor	270R	Panel LEDs		RR-0658	Jaycar	$0.38	$0.25	$0.22	$0.22	63	0		$0.00
Resistor	4.7k			RR-0588	Jaycar	$0.38	$0.25	$0.22	$0.22	2	2		$0.76
Resistor	10k			RR-0696	Jaycar	$0.38	$0.25	$0.22	$0.22	6	4		$1.52
Resistor	100k			RR-0620	Jaycar	$0.38	$0.25	$0.22	$0.22	10	0		$0.00
Variable resistor	100R	(500R)	Horizontal	RT-4304	Jaycar	$0.32	$0.32	$0.22	$0.19	1	1		$0.32
Variable resistor	5k		Horizontal	RT-4310	Jaycar	$0.32	$0.32	$0.22	$0.19	1	1		$0.32
Variable resistor	10k		Horizontal	RT-4312	Jaycar	$0.32	$0.32	$0.22	$0.19	1	1		$0.32
Variable resistor	100k		Horizontal	RT-4318	Jaycar	$0.32	$0.32	$0.22	$0.19	1	1		$0.32
													$24.79
IC: 4x2 In. NAND	74LS00N		DIL14	ZS-5000	Jaycar	$0.60	$0.50	$0.35	$0.35	1	1		$0.60
IC: Hex Inverter	74HCT04N		DIL14		Jaycar								$0.00
IC: Hex Inverter	74LS04N		DIL14	ZS-5004	Jaycar	$0.60	$0.50	$0.35	$0.35	2	2		$1.20
IC: 4x2 In. AND	74LS08N		DIL14	ZS-5008	Jaycar	$0.60	$0.50	$0.35	$0.35	1	1		$0.60
IC: 2x4 In. NAND	74LS20N		DIL14	ZS-5020	Jaycar	$0.70	$0.60	$0.40	$0.40	3	1		$0.70
IC: 4x2 In. OR	74LS32N		DIL14	ZS-5032	Jaycar	$0.70	$0.60	$0.40	$0.40	1	0		$0.00
4 bit D-Latch	74LS74N		DIL14	ZS-5074	Jaycar	$0.90	$0.75	$0.50	$0.50	1	0		$0.00
IC: 2x one-shot multivibrato	74LS123		DIL16	ZS-5123	Jaycar	$1.50	$1.30	$0.90	$0.90	1	0		$0.00
IC: 2x 1-4 DEMUX	74LS139N		DIL16	ZS-5139	Jaycar	$0.95	$0.80	$0.55	$0.55	1	1		$0.95
IC: 8x Bus driver	74LS245N		DIL20	ZS-5245	Jaycar	$0.85	$0.70	$0.50	$0.50	7	7		$5.95
IC: 8 bit addressable latch	74LS259N		DIL16		Jaycar								$0.00
IC: 8x D-Latch	74LS373N		DIL20	ZS-5273	Jaycar	$0.90	$0.75	$0.50	$0.50	1	1		$0.90
IC: 20 key encoder	74C923N		DIL20		Jaycar								$0.00
IC: 8x driver	ULN2803A	LED Driver	DIL18	ZK-3855	Jaycar	$3.50	$3.15	$2.45	$2.45	8	0		$0.00
IC: Analog MUX	74HC4051N		DIL16	ZC-4051	Jaycar	$1.30	$1.10	$0.75	$0.75	7	7		$9.10
IC: A-D Converter	ADC0801		DIL20	ZK-3862	Jaycar	$9.95	$8.50	$6.50	$6.50	1	1		$9.95
IC: +5v Voltage Regulator	LM7805CT	L78S05CV (2A)	TO-220	ZV-1505	Jaycar	$1.20	$1.05	$0.80	$0.80	2	2		$2.40
													$32.35
MCU: PIC16F874/P	20Mhz		DIL40		Jaycar								$0.00
Parallel resonance XTAL	20Mhz	15pF - 30pF CL	HC49U		Jaycar								$0.00
DC Power jack	2.5mm		PCB Mount	PS-0520	Jaycar	$1.00	$0.90	$0.90	$0.80	1	1		$1.00
DIN Socket	5 pin MIDI		PCB Mount	PS-0350	Jaycar	$1.70	$1.50	$1.50	$1.35	1	1		$1.70
DIN Plug	5 pin MIDI		Inline	PP-0304	Jaycar	$0.78	$0.70	$0.70	$0.60	1	1		$0.78
1x PCB Header pins	50 pins	(25 pins Rqrd.)		HM-3250	Jaycar	$1.80	$1.60	$1.60	$1.60	1	1	1	$1.80
2x PCB Header pins	2x 50 pins	(2x 82 pins Rqrd.)		HM-3250	Jaycar	$1.80	$1.60	$1.60	$1.60	2	2	1	$3.60
1x PCB Pin Header Socket	32 holes	For LCD (14 Reqrd.)		PI-6470	Jaycar	$2.80	$2.45	$2.15	$2.15	1	1	1	$2.80
IC Socket	DIL8	For Optocoupler		PI-6800	Jaycar	$0.25	$0.22	$0.20	$0.20	1	1	1	$0.25
IC Socket	DIL14			PI-6501	Jaycar	$0.28	$0.25	$0.22	$0.22	10	6	1	$1.68
IC Socket	DIL16			PI-6502	Jaycar	$0.34	$0.30	$0.25	$0.25	17	9	0	$3.06
IC Socket	DIL18			PI-6503	Jaycar	$0.35	$0.30	$0.25	$0.25	8	0	0	$0.00
IC Socket	DIL20			PI-6504	Jaycar	$0.45	$0.40	$0.35	$0.35	1	9	2	$4.05
IC Socket	DIL40	For MCU		PI-6508	Jaycar	$0.70	$0.60	$0.55	$0.55	1	1	1	$0.70
Heatsink	TO-220	For +5v reg.	25.4mm grid	HH-8520	Jaycar	$1.50	$1.95	$1.95	$1.95	1	1	1	$1.50
					Jaycar								$22.92
					Jaycar							Proto:	$80.06
					Jaycar								
					Jaycar								
OTHER STUFF													
IDC plug, 50 way	2x25	For Analogue sample	Inline cable	PS-0990	Jaycar	$2.55	$2.15	$2.15	$1.95	2	1	2	$2.55
IDC socket, 50 way, PCB	2x25	For Analogue sample	Vertical PCB	PP-1116	Jaycar	$2.20	$1.95	$1.95	$1.75	2	1	2	$2.20
IDC cable, 50 way	1m	For Analogue sample		WM-4508	Jaycar	$5.60	$5.60	$5.60	$5.60	1	1	1	$5.60
Locking header plug, 8 way		For ICSP		HM-3408	Jaycar	$0.95	$0.85	$0.85	$0.75	1	1	1	$0.95
IDC cable, 16 way	2m	For ICSP		WM-4602	Jaycar	$2.00	$2.00	$2.00	$2.00	1	1	1	$2.00
Slider pot	Linear 10K	For Analogue sample		RP-3920	Jaycar	$2.35	$1.60	$1.60	$1.40	48	4	48	$9.40
					Jaycar								$22.70

Appendix F: USB MIDI Controller Test Plan

USB MIDI controller Test Plan		
Modified: 20/02/2004	**Version: 0.1**	
Week – milestone	**Test item**	**Pass?**
3 – EVB Firmware implementation	Debug output response to:	
	Potentiometers	
	Digital triggers	
4 – LCD	Contrast OK	
	INIT OK	
	Backlight OK	
	Display pot. input ID+val. on last moved	
4 – MIDI 1.0 OUT port CRO waveform	5mA current loop present on cable	
	Bit rate @ 31250 bps	
4 – MIDI 1.0 IN port CRO waveform	5mA current loop in cable vs MCU input (“voltage”-side) of opto-coupler are correct	
4 – Input reading firmware/circuitry	All pots. create appropriate response on PC-side software	
	All buttons create appropriate response on PC-side software or controller LEDs	
4 – LEDs firmware/circuitry	All LEDs respond correctly to PC-side software MIDI messages and/or controller buttons	
4 – MIDI 1.0 IN port firmware MIDI protocol functionality	PC-side MIDI test software responds correctly to all Pots. and buttons.	
4 – MIDI 1.0 OUT port firmware MIDI protocol functionality	Controller responds correctly to MIDI messages generated by PC-side test software	
8 – USB port CRO waveform	3.3V differential NRZI encoding	
	Bit rate @ 1.5Mbps	
8 – USB-MIDI firmware MIDI protocol functionality	PC-side MIDI test software responds correctly to all Pots. and buttons.	
8 – USB-MIDI firmware MIDI protocol functionality	Controller responds correctly to MIDI messages generated by PC-side test software	
8 – USB-MIDI to Legacy MIDI 1.0 port functionality	Appropriate waveform present on MIDI 1.0 OUT port when receiving messages over USB. May use loop test using legacy gameport on PC	
10 – PCB fabricated	Visual inspection	
12 – Final Prototype test	Smoke test	
12 – LCD	Contrast OK	
	INIT OK	
	Backlight OK	
	Display pot. input ID+val. on last moved	
12 – MIDI 1.0 OUT port CRO waveform	5mA current loop present on cable	
	Bit rate @ 31250 bps	
12 – MIDI 1.0 IN port CRO waveform	5mA current loop in cable vs MCU input (“voltage”-side) of opto-coupler are correct	
12 – Input reading firmware/circuitry	All pots. create appropriate response on PC-side software	
	All buttons create appropriate response on PC-side software or controller LEDs	
12 – LEDs firmware/circuitry	All LEDs respond correctly to PC-side software MIDI messages and/or controller buttons	
12 – MIDI 1.0 IN port firmware MIDI protocol functionality	PC-side MIDI test software responds correctly to all Pots. and buttons.	
12 – MIDI 1.0 OUT port firmware MIDI protocol functionality	Controller responds correctly to MIDI messages generated by PC-side test software	
12 – USB port CRO waveform	3.3V differential NRZI encoding	
	Bit rate @ 1.5Mbps	
12 – USB-MIDI firmware MIDI protocol functionality	PC-side MIDI test software responds correctly to all Pots. and buttons.	
12 – USB-MIDI firmware MIDI protocol functionality	Controller responds correctly to MIDI messages generated by PC-side test software	
12 – USB-MIDI to Legacy MIDI 1.0 port functionality	Appropriate waveform present on MIDI 1.0 OUT port when receiving messages over USB. May use loop test using legacy gameport on PC	

Appendix F2: MIDI Controller Test Jig setup



Appendix Z: Changelog

0.1 – 0.2: (first signing) 20/02/2004

- Improved clarity of conceptual schema diagram
- Removed block diagram of system showing PC software layers (some of it merged with Fig.1)
- Adjusted work schedule (had only allocated 12 weeks, now 14 are used)
- Update Gantt chart
- Fixed small typographical errors
- Update specifications to include function of most input controllers
- Changed USB functionality from “Minimum Outcome” to “Additional Outcome”

02 – 0.3 (submit to Dr. Yongsheng Gao) 20/02/2004

- Insert gantt chart into document as one page in Appendix
- Fixed ordering of project schedule to match Gantt chart

i Sound & Music (Australia) *Product Information for the UC33E*, http://www.sound-music.com/product_info.php?PID=158 [accessed 19/02/2004]

ii Sounds Live (Newcastle) Ltd. (UK) *Korg Microkontrol*, http://www.soundslive.co.uk/moreinfo.asp?ID=2672 [accessed 19/02/2004]

iii Promenade Music (UK) *Tascam US-428,* http://www.promenademusic.co.uk/catalogue/details.asp?pid=941 [accessed 19/02/2004]

iv Sounds Live (Newcastle) Ltd. (UK) *Tascam US-428*, http://www.soundslive.co.uk/moreinfo.asp?ID=1050 [accessed 19/02/2004]