Syndeo Module Design Specification

Proprietary and Confidential

Version 1.3
11/06/00

Executive Summary

You have a PDA, MP3 player or a digital camera and you want to transfer files to/from your PC. What would you have to do? You first have to drag out the synchronization cable, and then fire up the proper transfer software. This is fine when you have only one or two devices, but what happens when you have more? What if you don't want the hassle of tangled cables?

Portable devices such as Personal Digital Assistants (PDAs) and MP3 players have been gaining momentum in the last few years. As the price of PDAs decrease, many people wish to keep connected even when they are away from their computers. Although computer notebooks are cheaper than ever, they are very cumbersome to carry around. Portable MP3 player sales are also booming because of their light weight and small size in addition to their anti-skipping capability in comparison to CD players. Because of this, PDAs and portable MP3 players have been gaining popularity. However, the portability of these devices is sometimes very inconvenient, as they all require special software, cradles or cables to update or synchronize the data.

Adjungo Wireless is in the process of developing a next generation product that will allow communication between a PC and a remote, portable device, wirelessly. We hope to use the new wireless standard, Bluetooth™, as the primary means of wireless communication. This product will eliminate the need for cables or synchronization cradles that are currently used with the most portable devices. As an example of this wireless flash writer, our current project is to build the above mentioned wireless file transfer device and integrate it with a portable MP3 player called the iDAC.

Adjungo Wireless, through the design of this product, is establishing new grounds for mobile devices. The days of untangling messy cables and installing synchronization software are over with our product, the Syndeo Module. This next generation wireless transfer device aims to replace the needs for cradles altogether. By choosing to use the Syndeo Module in their designs, companies will have a clean, tested Bluetooth interface with decreased time-to-market.

This following document provides detailed design specification outlines the different components of the Syndeo module. The components of the system include the Host Side Device, and the Portable Side Device. There are also various subcomponents within both the HSD and the PSD. On the HSD, subcomponents include a wireless transceiver, user interface and PC application level software. On the PSD, subcomponents include the PSD driver software, a system controller, LCD, Bluetooth Module, and a user interface.

Table of Contents

List of Tables

List of Figures

Glossary

Table 1: List of acronyms

HSD	Host Side Device (the PC-based Bluetooth device typically responsible for providing files)
iDAC	Internet Digital Audio Cassette (the MP3 player developed by the DAE group during ENSC 340 1999-3)
Kbps	Kilo Bits Per Second
MB	Mega Bytes (~10^6 bytes)
LCD	Liquid Crystal Display
MP3	A compressed music format
PDA	Personal Digital Assistant
PSD	Portable Side Device (the module that uses Bluetooth to support writing to the mobile device's flash)
USB	Universal Serial Bus (a new standard in serial transfer of data, and can support speeds of upto 12Mbps. Version 2.0 is due out in 2001 and can support speeds upto 480Mbps)
UART	Universal Asynchronous Receiver / Transmitter (A long time standard in serial transfer. but have relatively low speed compared to USB – current maximum is around 1Mbps)
Flash Memory	Solid state memory that can retain information even when no power is supplied. They can also be rewritten over many times. These memory are commonly find in digital cameras, MP3 players and other portable devices.

1 Introduction

The Syndeo Module is a wireless flash writing add-on utility that operates using Bluetooth wireless technology. The Syndeo module consists of a microcontroller, LCD, input switches and a Bluetooth communications module, as well as the associated software. For debugging purposes, a UART port as well as an interface for in-system programming is also available. Companies choosing to use the Syndeo Module with their wireless devices can integrate this, or any subset of this hardware for use with their existing flash-based system to make them Bluetooth enabled. As our first example application of the Syndeo Module, Adjungo Wireless plans to develop a prototype device integrated with the iDAC MP3 player. Miniaturized, cost effective versions of the module will follow in the near future.

1.1 Scope

This document describes design specifications to be using in the creation of the Syndeo Wireless Flash Writer Module.

1.2 Intended Audience

The intended audience for this paper is our executive committee (Dr. Andrew Rawicz, Mr. Steve Whitmore, Mr. James Balfour and Mr. Jason Rothe), as well as Adjungo Wireless' team of design engineers.

1.3 Bluetooth Background

Bluetooth is a standard and specification for small-form factor, low-cost, short-range radio links between portable devices. A Bluetooth wireless flash writer is more than just a cord replacement. Some advantages a Bluetooth wireless flash writer can provides over a cable interface include:

- **Built in connection:** users need not worry about forgetting cables or having the hassle of carrying them. Computer clutter is also removed.
- **Universal interface:** any Bluetooth enabled device can change its software to communicate with any other device. Problems with incompatible protocols or connectors are eliminated.
- **Multiple connections:** a single server computer could provide transfers to multiple devices at once, each on a different frequency. This is seldom possible with corded devices where each needs its own socket.
- **Line of Sight:** Portable devices in different rooms could transfer data between themselves without having to run long cables through the walls.

2 System Overview

The block diagram, below, describes the relationships between the PSD and the HSD. The HSD consists mainly of software components and the Sigma Bluetooth Module. On the PSD, control is centered around the Atmel AT89 series microcontroller. The Atmel facilitate communication between the Bluetooth module, user control (LCD and Control Buttons) and the Flash memory.

Host Side Device

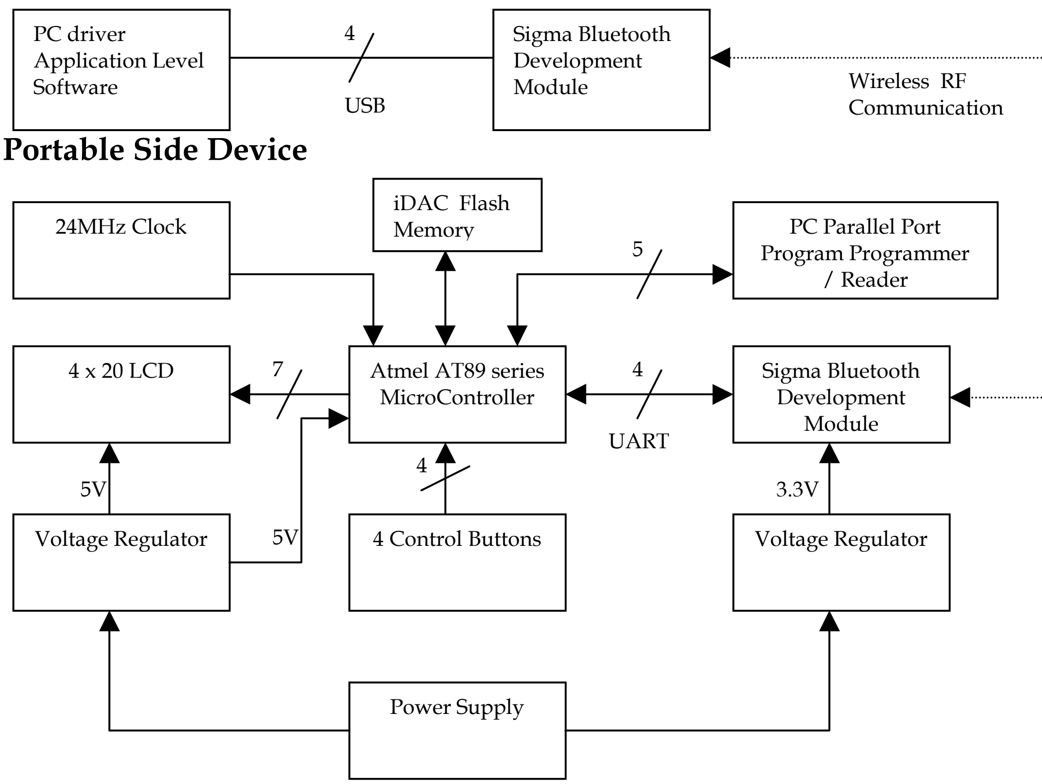

Figure 1: Syndeo System Block Diagram

3 Host Side Device Implementation

3.1 Wireless Transceiver Selection

The wireless transceiver is the module that will allow the portable device to communicate with the host side device. There are many factors that we considered when choosing this device. Some of these include:

- **Transfer Speed** – The unit will have to be able to transmit a large amount of data to the portable device in a small amount of time. This will reduce the amount of waiting time a person will have to endure in order to move the desired data from the server to their portable device. Because this is a cable replacement product, the transfer speed should be close to the speed at which a wired connection could obtain.

- **Range** – A longer range will allow users to obtain data even though they are far away from the server. If the devices are able to communicate without a line of sight limitation the user can download data without knowledge about the physical location of the server.

- **Maturity of Technology / Industry Support** – Many wireless transfer mechanisms are relatively new and thus have little support. Additionally, if a larger base of large companies supports the technology, the technology is more likely to be around in the future.

- **Interface Pins** – A lesser amount of pins is desirable because it reduces the complexity and development time of the system.

- **Package Type** – This defines how the receiver/transmitter are packaged. I^C specifies integrated circuit and PCI cards fit into the computer. An I^C is desirable because it is less costly and is significantly less complex than a PCI card.

- **Price / Part –** The prices of the parts have to be relatively low in order to gain adoptance from the consumer. A lower price is desirable.

The following implementations have been considered.

1. Bluetooth
2. Wireless Ethernet 802.11
3. Infra-red

Table 2 lists the design candidates along with information pertaining to them.

Table 2: Wireless Transceiver Characteristics

Characteristic	Bluetooth	Wireless Ethernet 802.11	Infra-Red
Transfer Speed	1Mbps	10Mbps	4Mbps
Range	30 ft.	100 ft.	10 ft. line of sight
Maturity of technology	Immature	Very Immature	Mature
Interface pins to host side device	4	PCI card (32 bit)	4
Interface pins to portable device	5	Varies (32 bit)	5
Package type	I^2 C	PCI card	I^2 C
Intended Use	Portable Devices	Mobile Computers	Various
Price/part (in mass production)	<$10	>$10	Low

After much consideration, Adjungo Wireless has decided Bluetooth™ technology fits our needs best. This specification has been adopted by many large companies and is supposed to be the great cable replacement technology.

We feel that the added range and transfer speed of 802.11 is offset by the increased development complexity. Additionally, the line of sight limitation offered by infrared, limits the market share of the product.

Currently, the prices for Bluetooth™ enabled devices are very high. But, as with most technology, the prices should drop as production increases.

3.2 User Interface Design Issues

The user interface that will reside on the host side device will have to allow the user to perform the functions outlined in section R8 of the Syndeo Module Functional Specification. The following sample screenshot illustrates the interface for the interface.

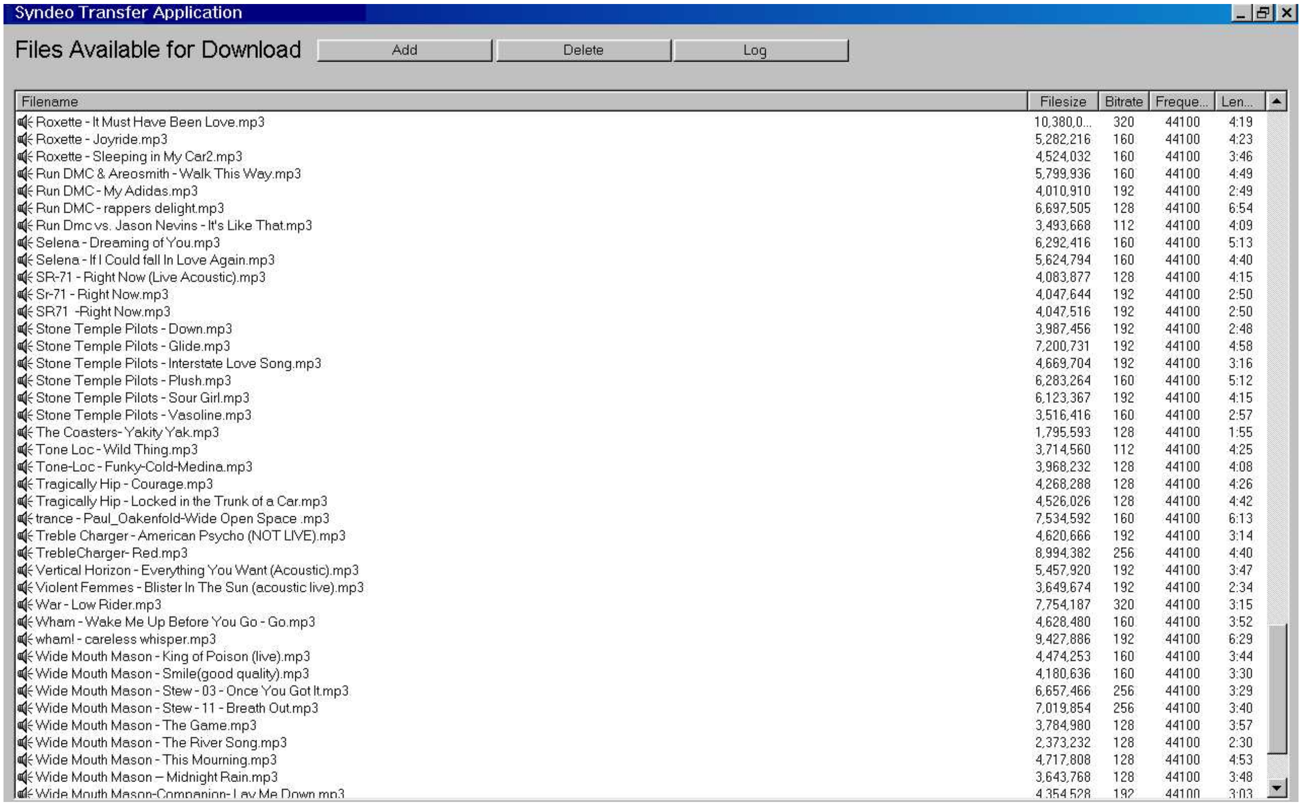

Figure 2: Syndeo File Transfer Application Screen Shot

The application consists of three screens. The first screen shows the files that are available to download. The second screen shows the files that have been taken from the server. Finally, the third screen shows all Bluetooth™ messages received by the system.

3.3 PC – application level software and drivers

The host server software must run on a Bluetooth enabled Windows based PC. In order to keep meet this requirement we felt that the best programming environment is Microsoft Visual C++ version 6.0 (MSVC). This development environment allows for quick and easy prototyping. Additionally, MSVC has been tested on all of the above platforms and is deemed to run stable.

At the application level there exists a set of functions used to access lower level devices. These functions allow the application to access the Bluetooth™ hardware with the actual method being transparent to the application. These drivers are to be written in MSVC to ensure stability while running under the above operating systems.

4 Portable Side Device Component Selection

The following sections describe the criteria used in selecting the major components of the PSD.

4.1 Mobile Device Selection

Mobile device selection is limited to the choices made for the host side device. To ensure compatibility, the portable side device should be running the same transfer protocol as the host side device. For the host side device, the Bluetooth™ transfer protocol has been selected. Thus the same method will be used for the portable side device.

4.2 LCD Selection

The LCD is one of the key elements in the user interface design for the PSD component of the Syndeo module. There are a number of factors to be considered in selecting an appropriate LCD, including:

- **Display Size:** As one of the larger components in the Syndeo module the LCD can be the limiting factor for the size of the module, making it desirable to have a small LCD. However, the LCD must not be too small since it displays key information to the user regarding the operation of the module, files available for transfer and other general messages.
- **Special features:** Features such as a backlight can be very useful, especially when using the device in a badly lit area or at night.
- **Number of I/O pins:** The number of I/O pins needed to control the LCD may have a significant impact on the total pin count of the module.
- **Price:** There is a wide variety of LCDs available in the market and since the price of an LCD could make up a significant part of the product price and it is highly desirable to minimize the cost of this part.
- **Availability:** It is very important that the LCD chosen is readily available in the market once production of the module begins.
- **Compatibility:** In many customer applications we will be making use of existing LCDs, already in the hardware and thus it is important to choose a widely used LCD controller for software reusability.

After considerable investigation we have found that LCDs with generic HITACHI drivers were the most accessible and are widely used in industry.

Upon consulting Mr. Jason Rothe, he informed us that he had an LCD available that fit our general specifications and offered it for our use. We reviewed the LCD specs and decided that the BT 42008 4 line by 20 character display would be of great use especially in the prototyping stages of the Syndeo module. It has a generic Hitachi controller (HD 44780) and driver (HD 44100).

4.3 Microcontroller Selection

Overseeing the entire Syndeo module's operation relies largely on the system controller. It coordinates the serial interface with the Bluetooth module and the user interface, including both the LCD and push buttons. A number of factors were considered in selection of the microcontroller, including:

- **Serial Communications:** Most Bluetooth link controllers on the market today typically offer both USB and UART communications so it was essential that the microcontroller supported one of these protocols. For more information on the communication, see 4.5, Blueooth Module / Microcontroller Communication Selection.
- **Program Memory - Type:** During product development it is critical that the chip's internal program memory be quick and easy to modify. UV erasable EPROM does not meet these criteria as it is too slow to modify. Flash memory and EEPROM, however, would be sufficient. In addition, it is desirable to have the program memory in-system programmable so that development does not require a separate memory programmer device and to decrease the likelihood or damaging parts with constant insertion and removal.
- **Program Memory - Size:** Since we will be implementing a Bluetooth protocol stack, as well as a rather complex user interface, it is important that the microcontroller have a non-trivial amount of program memory. Very optimized Bluetooth stacks are available commercially with code sizes as small as about 8kbytes[1]. By only implementing the features of a Bluetooth stack that are necessary for the transfer of files and short messages between only 2 devices at a time, we hope to keep our Bluetooth stack to 6kbytes. We also require a minimum of 4kbytes more for management of the user interface and serial port.

[1] http://www.adamya.com/sysdiv/bt/bt.htm

- **Data Memory:** In order to buffer data while copying from the serial port (from the Bluetooth module) to the external flash memory (for storage of MP3s in the iDAC) we desire a minimum of 128 bytes. In addition, system variables and string data for displaying to the LCD require a minimum of an additional 64 bytes of data, for a total of 192 bytes.
- **Speed:** In order to write to the external flash memory (for storage of MP3s in the iDAC) a minimum speed of 12MHz for the internal system clock is recommended[2]. In order to also handle data coming in through the serial port we will insist upon a minimum of 20 MHz.
- **Number of I/O Pins:** There is considerable demand for input and output pins including:
    - LCD Interface: Output (minimum 7, maximum 11)
    - Push Buttons: Input (minimum 4)
    - In-System Programming: Input (3)
    - Serial Communication: Input/Output (minimum 2)
    - Flash Memory: Input/Output (15)

  This gives us a total of 31 I/O pins as the minimum requirement, however, if we multiplex the 3 In-System programming pins with another set of pins (since no other pins will be required while the chip is being programmed) we can reduce the minimum requirement to 28.
- **Price:** Since these microcontrollers may be used in customers' products which may already be being sold at low prices, the cost of the microcontroller could make up a significant part of the product price and thus is highly desirable to minimize.
- **Compatibility:** Although our software is being developed for use with an example product (the iDAC) it is also hoped that this software could run on customers' existing microcontrollers in their portable devices. Thus, it is important to have a good chance of the code developed being compatible with commonly used microcontrollers.
- **Ease of Development:** Due to our limited time, budget and manpower, it is important to have a reliable, flexible development environment available free or at a low price.

After much investigation the following two microcontrollers, described in Table 3, were decided to be most appropriate.

[2] http://www.sandisk.com

Table 3: Microcontroller Comparison

	Atmel (AT)89S53 (8051 Style)	Microchip (PIC) 16F877
Serial Communication	UART	USART
Program Memory Type	Flash – In System Programmable	Flash– In System Programmable
Program Memory Size	12 kbytes	8k x 14bits
Data Memory	256 bytes	368 bytes RAM + 256 bytes EEPROM
Speed	24 MHz	20 MHz
I/O Pins	32	33
Price	$5.60US (Arrow – PDIP)	$6.41US (Arrow - PDIP)

As this comparison shows, the two microcontrollers are very similar in the features offered by the hardware. With respect to development support, the PIC part is preferable because of the integrated development environment provided free from Microchip. Although Atmel does not offer any development software for its 8051 themselves, there are several third party development tools (compilers, linkers, simulators) that should work well (although not flawlessly) with this Atmel microcontroller.

Adjungo Wireless was also concerned with the compatibility of the code developed for our device with the microcontrollers used in customers existing products. Microchip uses their own proprietary format that would not easily transfer to other microcontrollers, whereas the Atmel chip uses standard 8051 format which has a greater chance of being compatible. Thus, despite the increased difficulty anticipated in developing code for the Atmel processor, it was chosen because of its slightly lower price and greater likelihood of compatibility with customers' products.

4.4 Bluetooth Module Selection

The number of choices for a Bluetooth module is still fairly limited because there are not too many companies with an affordable functional product. Major companies involved with the Bluetooth technology include: Ericsson, Philips-Semiconductors, Toshiba, Comtec Sigma, Temic (Atmel) and Telelogic.

On investigating we found that the Bluetooth components may be acquired in several different formats, including: as a separate Bluetooth radio and a Bluetooth baseband processor, as a Bluetooth module (including the radio and baseband processor), and finally as part of a development board. After contacting several of the Bluetooth component manufacturers we found that individual components were either only available to major OEM vendors or with considerable back order time (12 weeks+). Other manufacturers (Temic) also had their product still limited by export regulations from the US. Considering that our project is in its inception stages and that we are under tight time constraints we chose to try and acquire a Bluetooth development kit.

We made contact with several companies that produced Bluetooth development kits and found that the most mature Bluetooth products are available from Ericsson. However, Ericsson's prices were quite high (on the order of several thousand USD) and they did not offer any special rates for student projects. As a result we decided to go with Comtec Sigma's Application and Training Tool kit. This kit is specially prices for university students interested in learning about Bluetooth and using it in a project. The kit uses a Bluetooth module from Ericsson, and provides a USB/UART interface to the module. The price of the Comtec Sigma's kit is $500USD/module (two are required to make a connection). While still quite pricey this price is an order of magnitude lower than the one for the equivalent kit provided by Ericsson

4.5 Blueooth Module / Microcontroller Communication Selection

The Sigma Bluetooth module has two interfaces available, a Universal Asynchronous Receiver / Transmitter (UART) and a Universal Serial Bus (USB). UART is an old standard from the 80s that is mainly used for asynchronous communication in electronic devices and computers. USB, on the other hand, is a new standard that is aimed at providing a much higher speed of serial transmission.

During the Microcontroller selection phase, we found that many of the microcontrollers that have USB interfaces do not meet our other specifications. The Atmel Microcontroller is able to provide a high UART transfer rate of 921,400bps (theoretical), higher than the 100kbps requirement for the prototype as specified in the Syndeo Module Functional Specification [Requirement R3.1]. As Bluetooth is currently limited at 1Mbps transfer rate, the UART interface between the microcontroller and the Bluetooth module will be close to the maximum data rate (but not optimal). In addition, as UART has been an industry standard for a long time, it is comparatively easier and cheaper to implement an UART interface than a USB interface. As the cost of USB decreases and as the speed of Bluetooth increases, we can scale the system to use the USB interface at a later time.

5 Portable Side Device Implementation

5.1 Portable Side Device Software Implementation

5.1.1 Modes of Operation

The execution of the Portable Side Device will be broken down into two primary activities, “browsing mode” and “transfer mode”, as shown in Figure 3.

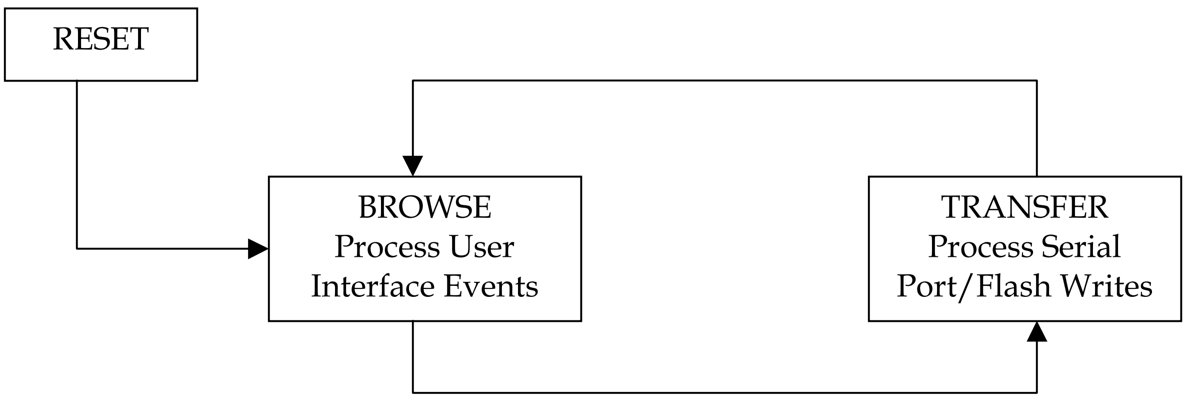

Figure 3: PSD State Diagram

5.1.1.1 Browse Mode

This mode is used for the user to browse through menus, and file lists (for both the PSD and those available on the HSD). It would also provide pop-up messages to be sent to the PSD from the HSD. In this mode interrupts could be enabled so that whenever a button is pressed, or data arrives from the UART, we jump to an ISR to record that the event occurred in a memory location and quickly return to executing code. In this mode we will allocate 128 bytes of RAM to the LCD related activities (enough to hold four 32 character song names), 64 bytes of RAM to the UART for buffering incoming /outgoing data and leave the remaining 64 bytes for miscellaneous data and stack space.

5.1.1.2 Transfer Mode

This mode is entered once a user has selected a file or list of files to transfer to or from the PSD. During execution in this mode, interrupts from the keypad are disabled and a message on the LCD is displayed, indicating that a transfer is in progress. By eliminating the distractions of the LCD and keypad, we can maximize the transfer rate over the UART and on to the flash without having to worry as much about losing data. Since we don't need to modify the LCD we can also use additional memory resources that would normally be taken up by strings of data being sent to the LCD. In this mode we will not allocate any dedicated memory to the LCD, but will set out 192 bytes of RAM to the UART for buffering incoming /outgoing data on its way to the Flash memory. This leaves us the remaining 64 bytes for miscellaneous data and stack space.

5.1.2 Software Hierarchy

The following diagram describes the organization of the software of the portable side device. Dashed lines indicate links that will be used only in the debugging mode when the UART is connected to a PC for debugging (rather than to the Bluetooth module).

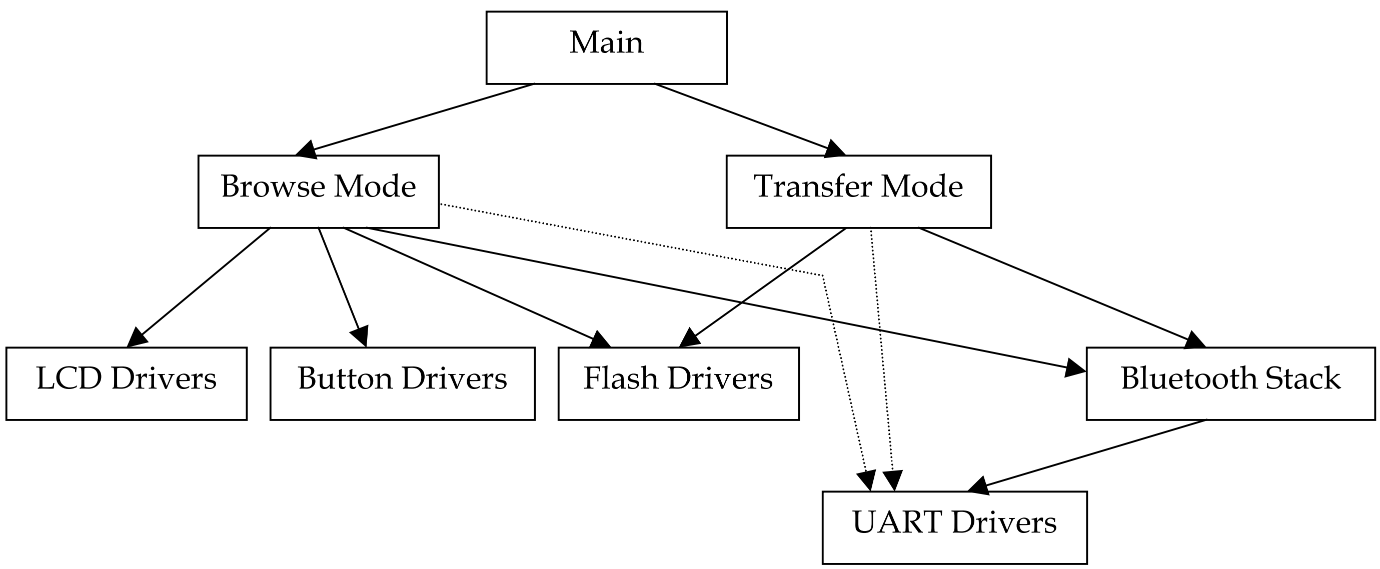

Figure 4: Software Organization for the PSD

5.2 User Interface Implementation

5.2.1 User Interface States

With the use of a state machine, the user interface can be easily implemented in modules and also allows for greater flexibility in terms of expansion. The diagram below describes in detail the states that are available in the user interface.

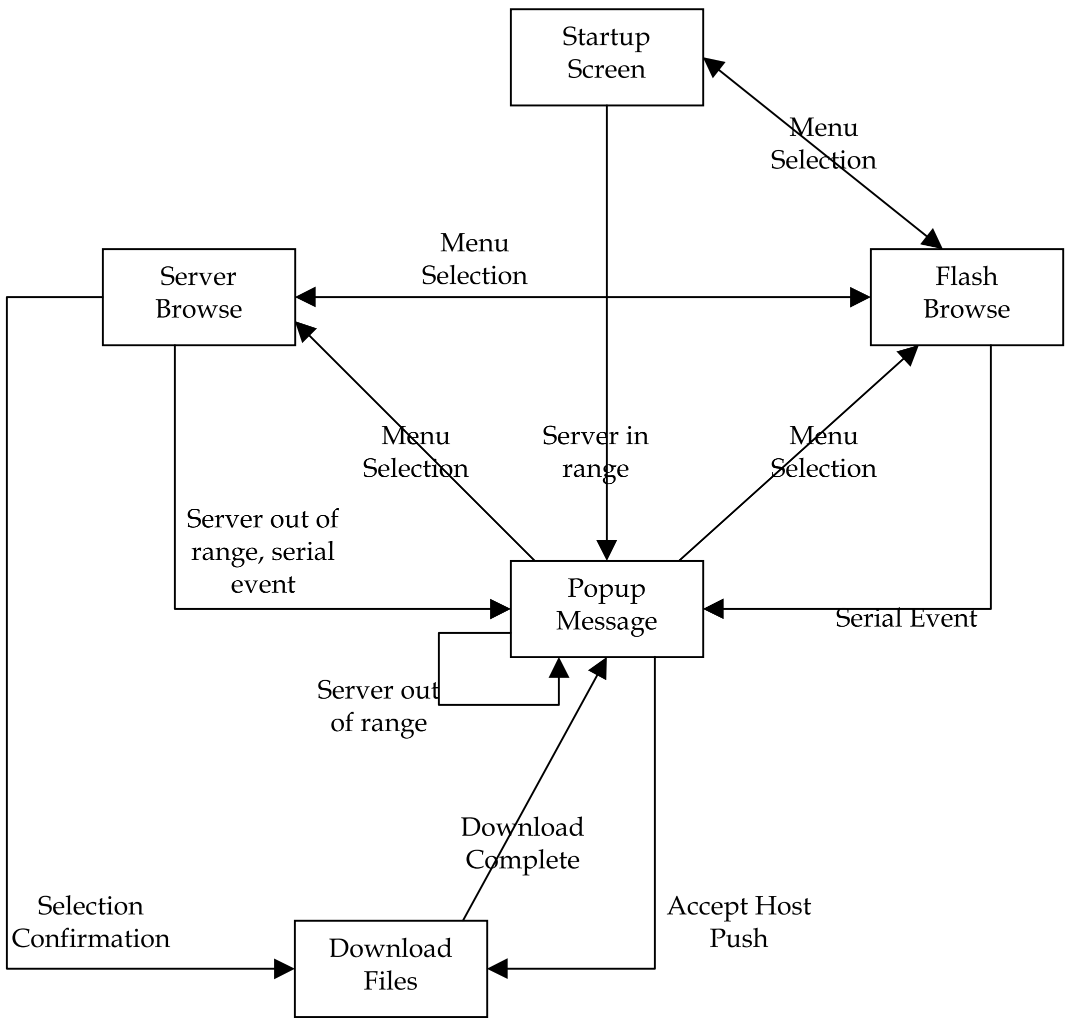

Figure 5: User Interface States

*Note: Options to go to "Server Browse" would only be available if server was in range.

5.2.2 Sample Screen Shots

This is the user interface that is envisioned at the current time. This current design is optimized for the 4x20 display and the use of four buttons. If the specification of the LCD or the number of button changes, modifications will be made to the user interface to optimize for clarity and user-friendliness but the overall design would remain the same. The 4x20 grid represents the 4x20 LCD and the ovals represent the 4 control buttons.

The sample, in Figure 6, below, shows the Flash / Server browsing interface

►	S	O	N	G	1														
	S	O	N	G	2														
►	S	O	N	G	3														
	U	P			D	O	W	N		M	E	N	U			S	E	L	

Figure 6: File Browsing Interface

Figure 7 shows the menu selection interface.

►	B	R	O	W	S	E		S	E	R	V	E	R						
	B	R	O	W	S	E		F	L	A	S	H							
	D	O	W	N	L	O	A	D		S	O	N	G	S					
	U	P			D	O	W	N		M	E	N	U			S	E	L	

Figure 7: Menu Selection Interface

6 Testing

In order to ensure that all the requirements for a functioning prototype are met our team is planning to perform rigorous testing. Each independent component will be tested before integration with the prototype and, as each new component is added to the system, extra tests will be performed to confirm proper interfacing between the system and the newly added parts. The main requirements to be tested in the prototype include:

6.1 Dual way communications

Dual way communications will be tested using the following procedure:

1. Inspect the contents of the PSD from the HSD, and then send several files from the HSD to the PSD.
2. Retrieve one of the files sent to the PSD from the HSD side, and send the remainder to the HSD from the PSD side.
3. Inspect the contents of the HSD from the PSD, and download several files.

This will ensure that file transfer is fully operational in both directions.

6.2 Data transfer rate

The data transfer rate will be tested as follows:

1. Start a transfer from the HSD to the PSD of a large file (several MB).
2. Measure the time it takes for the file transfer to complete with a stopwatch. This will be used to determine the HSD to PSD transfer rate.
3. Start a transfer from the PSD to the HSD of a large file (several MB).
4. Measure the time it takes for the file transfer to complete with a stopwatch. This will be used to determine the PSD to HSD transfer rate.

This test procedure will be executed as many times as possible and in several different environments in order to get a valid statistical average data transfer rate.

6.3 Connection stability and reliability

We hope to demonstrate the stability and reliability of the HSD to PSD through long high volume data transfers in several different environments.

6.4 Detection of devices within range and broken connections

The effective range of operation will be confirmed in the following way:

1. Power on both the HSD and the PSD.
2. Move the PSD farther from the HSD until the connection is broken.
3. Move the PSD back towards the HSD until a connection is re-established.
4. Power down the PSD and check that the HSD detects the lost connection.
5. Power up the PSD and power down the HSD and verify that the PSD detects the lost connection.

6.5 Graceful recovery

Graceful recovery from unsuccessful data transfers will be demonstrated as follows:

1. During a data transfer the PSD will be moved out of the range of communication.
2. During a data transfer the PSD will be powered down to simulate a power failure.
3. During a data transfer the HSD will be powered down to simulate a power failure.

6.6 Power consumption

Power consumption will be monitored often by measuring the voltage and current used by the Syndeo module while in operation. It is crucial to ensure parts are not damaged during development and that it will be possible to power the portable device for a reasonable amount of time before the battery would need recharging/replacement.

6.7 Functionality of LCD and buttons

The functionality of this part of the PSD will be demonstrated by controlling uploads/downloads between the PSD and the HSD from the PSD side.

6.8 User friendliness

In order to ensure ease of use several test runs are planned with non-engineering students and people with a non-technical background.

7 Conclusion

This document has detailed the technical design of an operational prototype for a wireless flash writer. The Syndeo wireless flash writer will be constructed based on the design specifications described in this document.

This document covers the requirements labeled by a (1) in the Syndeo Module Functional Specification which are to be implemented as part of the prototype device which is to be completed by December of 2000.

8 Sources and References

1) Sigma Comtec, http://www.comtec.sigma.se/

2) Bluetooth Solutions from Ericsson, http://bluetooth.ericsson.se/

3) Ericsson, http://www.ericsson.se/

4) Engineering Science Communications Handbook: Module 3 7th Edition, S. Stevenson, S. Whitmore, 1999.

5) *Syndeo Module Project Proposal*, A. Morning-Smith, R. Lee, S. Lau, V. Jovanovic, September 25th, 2000.

6) *Syndeo Module Functional Specification,* A. Morning-Smith, R. Lee, S. Lau, V. Jovanovic, October 16th, 2000.

7) Specification of the Bluetooth System, Volume 1 – Core, v1.0 B, December 1st 1999

8) Specification of the Bluetooth System, Volume 2 – Profile, v1.0 B, December 1st 1999

9) A commercial Bluetooth stack for microcontrollers, http://www.adamya.com/sysdiv/bt/bt.htm .

10) Flash Memory Specifications, http://www.sandisk.com

11) LCD Specifications, http://www.batron.com/displaytechnik/lcd-alpha/bt_42008.htm

12) Bluetooth specification, www.bluetooth.com

13) Online ordering (for pricing estimates), www.arrow.com