This document outlines some details to help you better understand the features of the new IRLP Embedded machines. It also contains a couple of "gotchas" and FAQ type information pieces. As more questions are asked, more information will be added to this document.

An embedded node is an IRLP node. Most of the information you will require is available from the following website:

http://www.irlp.net/new-install/

This document will live online as a PDF file. The version of the PDF will be updated as more things are added to the document. Please refer to the ChangeLog on the last page of this document for details on the updates.

David Cameron – VE7LTD
IRLP System Designer

Table of Contents

Technical Details

Each unit is developed using the same series of motherboard, the VIA M/ML series of mini-itx motherboards. This simplifies the installation, configuration, and support of these systems. The individual case used in your system may differ, but the core system is the same.

The specifications of the system are as follows:

667 MHz clock speed (VIA EDEN ESP fanless processor)
256 Meg DDR233 RAM
128 Meg IDE Flash Drive
Standard version 3.0 IRLP board

More specifics of the motherboards can be found on the VIA site:
http://www.via.com.tw/en/products/mainboards/motherboards.jsp?motherboard_id=301
(or for TV OUT version)
http://www.via.com.tw/en/products/mainboards/motherboards.jsp?motherboard_id=81

Wiring

The IRLP-to-parallel cable is directly wired to the solder side of the motherboard. In some 19" rack cases, the cable is wired to the component side (due to low clearance between the bottom of the board and case).

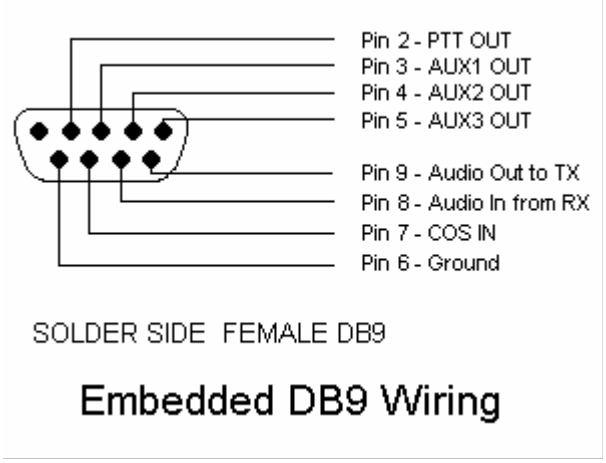
The DB-9 interface between the radio and the IRLP board is directly wired to the audio input and output on the motherboard. On the "fanless" cases, the audio wires from the motherboard are soldered to the IRLP board for simplicity.

The wiring of the embedded node DB-9 connector is identical to the Version 3.0 wiring, except that audio output is available on pin 9 of the DB-9 connector. This means that audio input (and the input to the DTMF decoder) is on pin 8 and the audio output is on pin 9. There is no need to use 1/8" stereo connectors to connect to the audio jacks on the motherboard. The audio fed to the IRLP Embedded node on Pin 8 must be squelched and de-emphasized audio (speaker audio).

IMPORTANT – There are TWO male DB-9 connectors on the back of the embedded node. On ALL cases, the IRLP DB-9 is the one located closest to the power connector. The other is an RS-232 serial port. Plugging the IRLP wiring into the wrong port will damage the radio or serial port.

IMPORTANT – Many scripts (including the morse ID) require that the AUX1 and PTT outputs be bridged. This means that you should solder a jumper between pin 2 and pin 3 on the DB-9 cable you build. This is REQUIRED for all controller installations and for many scripts developed for the Embedded IRLP nodes.

The wiring is as follows:



Embedded DB9 Wiring

Pin 2 – PTT Output
This pin should be connected to the radio PTT. This lead is active LOW only (goes to ground for radio to transmit).

Pin 3 – AUX1 Output
This pin should be connected to any AUX source. For repeater operation and to use some scripts, this must be tied to PTT (pin 2).

Pin 4 – AUX2 Output
This pin can be connected to any AUX source. This lead is active LOW.

Pin 5 – AUX3 Output
This pin can be connected to any AUX source. This lead is active LOW.

Pin 6 – Ground (common to radio and power source)
The ground should be common to the chassis of the radio, audio lines, and common to the power source ground.

Pin 7 – COS (Carrier Operated Squelch) Input from radio
COS is a digital ON/OFF signal from the receiver that changes with the presence of a received signal. The COS can be active high or active low. The COS jumper on the board switches between active HIGH and active LOW. All embedded nodes are shipped with the jumper in the COS active LOW position.

Pin 8 – Audio In (from receiver)
This pin should have the squelched audio from the receiver. This is the audio input to both the node and the DTMF decoder on the IRLP board.

Pin 9 – Audio Out (to transmitter)
This is the audio source for the transmitter. It should be tied to the microphone input of the transmitter. Some microphones have a biased voltage applied to them to power the microphone element. **THIS OUTPUT IS NOT DC PROTECTED**, and a series 4.7 or 10uF capacitor can be added to remove any DC portion on the line. Failure to do so has damaged the motherboard.

The Operating System

The operating system is based on Slackware Linux 10.2. The images are built from a live operating system, operating under VMWARE on my development computer. As the flash based system matures, flash updates will be made available to the owners of the Embedded machines.

Flash Disk File System

The flash disk is a 128 megabyte solid state disk, which operates as a standard IDE device. It is partitioned in two partitions: a 2 megabyte FAT12 partition (used for booting using syslinux) and a 126 megabyte ext2 partition (used for the Linux and IRLP filesystem). Neither filesystem is compressed, as 128 megabytes is plenty of space.

There is approximately 35 megabytes used by default for the Linux filesystem and the IRLP files. This leaves about 70 usable megabytes free for log files and customization. This also leaves 100 megabytes of RAM free for use by Linux. 16 megabytes of the RAM is used by the VGA display controller.

The entire system loads from the ext2 partition into RAM on boot, and then continues to run from RAM, so that there can be no physical damage to the flash device from repeated write/re-write cycles. The flash device is designed for several million read and write cycles, but in the design of the embedded system, the flash device should outlive the life of the computer. This system should also never require a reboot.

Updating the Operating System Components

There is currently no planned way to update the filesystem operating components. As the project advances, you will be able to download the latest flash images, and update your system with the latest flash images. A backup script similar to the one on other IRLP nodes will be designed to backup your current system before a re-flashing. The re-flash will be able to be performed on another computer running Linux (and maybe even Windows).

ROOT Password

The default root password on every shipped embedded IRLP machine is

```
rootirlp
```

The system will force you to change this on first login. DO NOT forget your password, as we can not recover it for you. Your whole system will have to be re-flashed and reconfigured from scratch. This is because it is very difficult to modify files (including the password file) on the flash drive, as it is only used read-only. Once the initial passwords are set, the changes are automatically synced to the flash device for the next boot.

The repeater user password and IRLP Web Admin user (username=remote) passwords are not set initially, but are set on the first boot.

Configuring the Network Adapter

The network adapter driver is already preloaded into the system. The network adapter is initially set for DHCP assigned addresses. If you are operating your node behind a router, you will want to set your IP address statically. The script

```
/root/set-static-ip
```

is used to configure the help you configure a static IP address. ONLY if that script does not work, you should try the program:

```
netconfig
```

to set up your network card. Please use the following entries in the netconfig program:

Hostname – your node number (stn9999)
Domain – ip.irlp.net

The static IP address should be determined from the IP addresses your router hands out. There are several ways to assign one, and if you need help determining what your static IP should be, please contact the support group.

Forwarding IRLP Ports Through a Router

As with any other IRLP node, if your Embedded node is behind a router, you have to forward some TCP and UDP ports to the node, or put the node into the DMZ of the router.

The IP address of your node should have automatically been set static by the scripts in the section "Configuring the Network Adapter". You will need to forward the following ports through your router to the IRLP machine:

```
15425               TCP
2074 to 2093        UDP
```

If you want to be able to access the remote web administrator from the outside, you will also need to forward:

```
80                  TCP
```

If you are using Echolink on the Embedded node, you will also need to forward:

```
5198 and 5199       UDP
```

If you want to access your console remotely (through SSH), or would like help from the support crew remotely, you will have to forward:

```
15426               TCP
```

Alternately, if your router has a "DMZ" feature, you can put the IP address of the node into the DMZ, and all ports will be forwarded to the IRLP node.

Timezone Configuration and Time Setting

The timezone is configured using the program:

```
timeconfig
```

***NOTE**: You should choose NO at the first prompt (hw clock is set to local time).*

The current time is set using the internet activated program:

```
/usr/sbin/set-time
```

This program uses rdate, which requires that the internet connection is active to set the time. The set-time script is also run daily from the /etc/cron.daily directory.

Synchronization of the Flash Drive

The script:

```
flash_sync
```

synchronizes the current file system operating in RAM with the flash drive. This is done using the same sync software that IRLP uses (rsync) to keep its filesystems up to date. This currently backs up all log files, configurations, settings etc. Since your filesystem is volatile, you must remember to run the flash_sync script after any major changes to make sure the flash drive is current before a reboot.

Be careful NOT to write large files to the flash device. If the flash device is full, the filesystem may not load on boot, and your embedded node will have to be re-flashed from scratch, losing its entire configuration.

You can control which files are NOT updated by adding the files to the following file:

```
/etc/flash_sync_exclude
```

DO NOT remove any of the entries that are currently pre-configured in the file. If you do, you run the risk of crashing the flash drive, requiring a full re-flash.

The flash_sync script is run daily from the following script:

```
/etc/cron.daily/flash_sync_quiet
```

The script must be run as root, so if you plan to keep your flash drive more up to date than once per day, you should set the quiet script to run more often.

SSH

The SSH port defaults to 15426. This is made to prevent standard hacking scripts that hunt SSH servers have a harder time finding your machine. This means you have to program port 15426 into any SSH or SCP program (such as PuTTY or WinSCP) you may use. Also, if you wish to have remote access to your IRLP node, you will need to forward port 15426/TCP as well.

In winSCP and puTTY you can set the port to 15426 where you set the IP.

In linux, you can use the format `ssh -p15426 IP_address`.

If you wish to change the SSH port to another port, you can edit the Port option in the `/etc/ssh/sshd_config` file.

NOTE: On the embedded nodes, I am experimenting with installing a "public key" on your node which will allow me to login to the nodes with root access without a password. You can remove this feature by removing the file:

`/root/.ssh/authorized_keys2`

Installed Features

The Embedded machines are pre-loaded with a series of commonly used add-on features.

1) REMOTE WEB ADMIN

The IRLP Remote Web Admin is installed by default on all embedded IRLP computers. This allows you to perform some simple control features from any web enabled computer. You can send DTMF to the decode script, enable/disable the node, drop the current connection, edit several files, view the messages file, and see the status of your node.

You access your remote admin system by entering the URL

```
http://stnxxxx.ip.irlp.net/control/
```

NOTE: You must forward port 80/TCP through your router for this to work from the outside. Some ISPs will block incoming data on port 80.

The default username and password are: remote/remote. These will be changed on first login with the root password.

This feature can be disabled (the web server is actually turned off) by adjusting the USE_WEBADMIN flag in the `/home/irlp/features/features.env` file.

If you lose the remote user password for the remote web administrator, it can be reset by entering the following command as the root user:

```
htpasswd /etc/htpasswd remote
```

2) MORSE_ID

The morse code generator cwpcm is installed by default on all embedded machines. There are a series of scripts that can be run using this program to generate morse code identifiers.

The format for using the program is

```
echo $CALLSIGN | cwpcm -sx -l 10 -p 1000 > /dev/dsp
(plays "callsign" at extra fast rate, 10% volume, 1000 Hz)
```

The sample script which keys, plays the morse ID, and unkeys is at:

```
/home/irlp/features/morese_id/id_now
```

cwpcm arguments:
-ss *for slow code rate (USA Technician?)*
-sm *for medium code rate*
-sf *for fast code rate*
-sx *for extra fast code rate (USA Extra?)*
-l <0..100> *set the output level*
-p <0..3000> *set the pitch*

3) CONTROLLER

The IRLP controller program is a simple controller that provides hangtime, DTMF muting, activity controller ID, and controllable courtesy tones.

The IRLP controller requires a pair of courtesy tone files to be setup. The controller requires files to be made that are RAW (no header), 8 bit, with a sample rate of 8000hz. The names are `ct1.raw` and `ct2.raw`. `ct1.raw` is played for local traffic, and `ct2.raw` is played for remote (IRLP and controller generated traffic). These can be made to whatever you like, using a program like CoolEdit. You can then use a program like WinSCP to transfer the files from a windows machine onto the embedded node. After the files have been sent, the `flash_sync` script must be run to save the changes.

The controller has an environment file at
`/home/irlp/features/controller/controller.env`
which can adjust many of the features of the controller. You can adjust:

- The hangtime in milliseconds.
- The shortkey timer in milliseconds (COS time required for hangtime activation).
- The alligator timer in seconds (COS timeout).
- Toggle DTMF mute on/off.
- The DTMF mute duration in milliseconds.
- The ID interval in seconds (time between IDs).
- The courtesy tone frequency if ct1/2.raw files are not available (plays through through PC speaker).

After making any changes to the environment file, the controller must be restarted by running:

```
/home/irlp/custom/rc.irlp
```

The DTMF mute is accomplished by setting the LINE volume slider to zero. This is done in software. This only affects the audio being passed from the Line INPUT to the Line OUTPUT, and not the volume of your IRLP output audio.

4) SPEAKTIME

The speaktime script is already pre-installed on the system. The DTMF code “C” is used to invoke the speaktime script.

5) STAR69

The star69 script is already pre-installed on the system. The DTMF code “*69” is used to invoke the star69 script.

6) EchoIRLP

The EchoIRLP system is installed on each embedded system. You must configure your own EchoIRLP system, by editing a series of configuration files. Some embedded nodes will be preconfigured with your EchoIRLP settings.

The configuration files for EchoIRLP are located in

```
/home/irlp/features/EchoIRLP/custom/
```

The files you will need to customize are:

```
userdata.txt
tbd.conf
echo_environment
```

7) IRLP_VPN

The IRLP_VPN is a system which will give you a routable IP address, even when you are located behind a router (or multiple routers) which you do not have access to. The IRLP_VPN service is offered (currently free of charge) to IRLP users who request an account. Your Embedded machine is preconfigured with a test account, so you can see the benefits of using the VPN. The test account may be terminated at any time. If you wish to continue using the IRLP_VPN system please contact dcameron@irlp.net to be set up with a permanent account. Each IP address costs money, so heavy users may be asked to support the system.

To activate the IRLP_VPN, you use the script:

```
start_tunnel
```

To close the connection, you use the script:

```
stop_tunnel
```

The tunnel sets up routes to/from your machine and the server, and will disconnect any active connections you are currently running. It also takes about a minute for your new IP address to synchronize across the IRLP system, so do not expect to receive any incoming calls right after you start the tunnel.

There is also a flag to start the tunnel when the machine is booted in the `/home/irlp/features/features.env` file.

If you plan to use the tunnel full-time, you will require some scripting to ensure that the tunnel is re-established after a loss in connectivity.

Adjusting Audio Levels

The LINE should be set as the only recording device. This is the default behaviour for all embedded nodes.

The level of audio passed from INPUT to OUTPUT (repeated audio) is adjusted with the LINE slider. If you are not using the IRLP controller, you should set this to zero.

The level of audio passed from INPUT to the IRLP encoder (IRLP input level) is adjusted with the IGAIN slider.

The level of audio passed from the IRLP decoder to the OUTPUT (IRLP output level) is adjusted with the PCM slider.

The overall OUTPUT gain of the system is adjusted with the VOL slider.

Be sure to run :

```
aumix -S (that is a CAPITAL S)
```

as root after adjusting any audio levels.

Mixing Audio Channels

One of the best and most useful features of the driver/chipset combination of the ALSA/VIA audio chipset is the ability to easily mix audio signals into the output device. Up to six simultaneous signals can be mixed, just by playing them using their normal means. This is not true of all other IRLP systems.

For example, you can play IRLP audio, an MP3, a courtesy tone, a morse ID, and a voice ID… All at once! All of these use the same DSP device at the same time. It may also be possible to pull multiple inputs at the same time, although this has not yet been attempted.

Custom IRLP Audio Files

The Embedded machines currently are set not to download customized audio files. This was done to save space. In the near future, IRLP will start "streaming" the audio files when the connection is established, which will allow the Embedded machines to retain the same functionality as other nodes. When this feature is available, your node will automatically download the required updates from the IRLP server.

The custom wave files that other nodes play when your node connects can be created and sent to the IRLP servers for re-distribution, just as they are for other IRLP nodes. Details on how to create those files is available at:

http://www.irlp.net/new-install/afterinstallv2.pdf

The document outlines using a floppy drive to place the files onto your system. Since embedded IRLP nodes obviously do not have floppy drives, you need to transfer the files to the node, and place them in the /tmp directory. This is performed using the WinSCP program. Details on setting up the WinSCP program can be found online. IRLP node owner Gary McDuffie has created a small "howto" on using WinSCP and PuTTY.

http://garymcduffie.com/irlp/winscpputty.html

Just a note you must be aware, both SSH and SCP listen on **port 15426** on your Embedded IRLP node (the default is port 22).

The /home/irlp/custom/update-file-list File

The file:

```
/home/irlp/custom/update-file-list
```

contains some important entries to prevent certain files special to your node from being overwritten during IRLP updates. In order to prevent duplication, the wget, rsync, and nc (netcat) files are not pulled from the IRLP server. Instead, the locally installed versions (which are newer) are retained and linked to the irlp file tree. If you wish to edit any of the files or scripts under the /home/irlp/ directory, and do not want them to be overwritten, just add entries into this file. This is a feature common to all IRLP nodes.

Updating your Node's Status Page Information

The status page information is updated the same way that a normal IRLP node is updated. The program lynx is used by issuing the following command:

```
lynx http://status.irlp.net:15427/updatenode
```

You can also use any other browser that is browsing from the same public IP address (behind the same router) to update the information.

ChangeLog

Date	By	Rev	Change
Aug 16, 06	VE7LTD	1a	Initial release
Sep 28, 06	VE7LTD	1b	Changed some errors, added info on DB9 location
Jun 20, 07	VE7LTD	1c	Updated some info, added wiring diagram, added WinSCP info