Use of Cycle.exe Performance Modeling Tool (For CM-1 only)

Steve Gray, Cirrus Logic, Inc.

Cycle.exe is a program used to model processor and network bandwidth requirements for different CobraNet^{TM} configurations. The modeling is done for a single CobraNet^{TM} interface and is limited, at the time of this writing, to modeling the CM-1 module only. *Cycle .exe* processes a file named *cycle.ini* as its input. *Cycle.ini* is a text file which is edited in order to define the configuration and environment of a CobraNet^{TM} interface. Processor and network bandwidth necessary to support the defined configuration are then determined.

Cycle.ini Sections

Effective use of this tool requires an understanding of the *Cycle.ini* file. *Cycle.ini* is divided into a number of sections with user definable variables in each section.
The sections are:

[notes]

This section is not processed by cycle.exe nor should it be edited by the user. It is included to show the valid values permissible for each definable variable.

[system]

The system section is used to define variables which affect performance globally within the CobraNet^{TM} interface.

[ip packets]

This section is used to define the IP packets that will be processed by the interface, including packet bridge packets and packets addressed to the interface that are not needed or processed and are dropped.

[receiver*n*]

There can be multiple [receiver] sections. The value of *n* can be 1 through the maximum number of receivers supported by the interface. This section(s) defines the configuration of the bundles processed by each receiver.

[transmitter*n*]

There can be multiple [transmitter] sections. The value of *n* can be 1 through the maximum number of transmitters supported by the interface. This section(s) defines the configuration of the bundles processed by each transmitter.

[system] variables

dsp clock : The clock rate of the processor. This should NOT be changed.

sample rate : Self explanatory; 48000 or 96000 audio sample rate

ssi mux : 2, 4 or 8 depending on whether the CM-1 is configured for 16, 32 or 64 channels. This is the number of channels multiplexed per serial interface.

frame samples:

The number of samples per audio frame. Although not obvious, this value is used to define latency.

64 samples in one frame per isochronous cycle	= 5 1/3 mS latency
32 samples in two frames per cycle	= 2 2/3 mS latency
16 samples in four frames per cycle	= 1 1/3 mS latency

network bundles :

The number of bundles on the overall network. This affects the size of the beat packet and, thus, the CPU bandwidth necessary to process it. As the number of bundles increases, the size of the permissions field in the beat packet increases.

network transmitters:

The number of CobraNet^{TM} devices transmitting audio data on the overall network. This affects the size the MAC list in the reservation packet and, thus, the CPU bandwidth necessary to process it.

meters:

Should be set to value from 0 to 32. The number of active meters. Audio meters can be assigned to each input and output channel. This affects CPU bandwidth.

loops:

Should be set to value from 0 to 8. The number of audio loops. Audio can be looped-back from one channel to another internally. This affects CPU bandwidth.

transmitters:

A typical configuration will have four or eight transmitters. Should be set to a value from 0 up to the number of transmitters implemented in the interface.

receivers :

A typical configuration will have four or eight receivers. Should be set to a value from 0 up to the number of receivers implemented in the interface.

wait states:

Should be set to 1 or 2. All CM-1 modules prior to Rev. F require this value to equal 2. Set this value to 1 for Rev. F modules or later to correctly model the faster SRAM used in these revisions.

default txsubformat:

The value of subformat to be used in the [transmitter*n*] section in the absence of an explicit declaration. See *note

default rxsubformat:

The value of subformat to be used in the [receiver*n*] section in the absence of an explicit declaration. See *note

[transmitter*n*] variables

bundle:

Should be set to 0 or 1 to disable or enable the transmitter respectively.

destinations:

The number of copies of this bundle transmitted. i.e. multi-unicast count.

subcount:

Should be set to a value from 0 to 8. The number of channels in a bundle assigned to this transmitter.

subformat:

Should be set to 16. 20 or 24. The sample size of each channel within the bundle assigned to the transmitter or receiver. See *Note.

[receiver*n*] variables

subcount:

Should be set to a value from 0 to 8. The number of channels in a bundle assigned to this transmitter or receiver.

subformat:

Should be set to 16. 20 or 24. The sample size of each channel within the bundle assigned to the transmitter or receiver. See *Note.

***Note**: When *default txSubFormat* or *default rxsubformat* are assigned in the [system] section, the value of subcount in the [transmitter] or [receiver] sections is overridden to equal 8 and any undefined subformat values are set to the value of *default txSubFormat* or *default rxSubFormat* as applicable.

Usage

In order to determine if a particular configuration will or will not work, edit the *Cycle.ini* file as needed to reflect the configuration under consideration and run 'cycle' under a DOS window.

Cycle will then either display an error message (in the case where Ethernet bandwidth is exceeded) or will display a table of processor cycle usage. Negative numbers in the processor cycle data indicate that the processor does not have enough cycles to support the entered configuration. A negative number is an error.

There is a fair amount of margin in this model. If it indicates that there are enough resources to support the configuration, then this will be true even if the node is a conductor or one of the bridging functions is active.