Design of Network Simulator supporting Bluetooth Protocol using System Verilog

Prof Morris Jones

Department of Electrical Engineering, San Jose State University, San Jose, California 95192

Introduction

Implementation of network simulator using System Verilog is a different approach to represent it in hardware model. This network simulator deals only with Bluetooth protocols.

Features of this network simulator:

•The uniqueness of this project is in terms of its use as a verification environment to test behavior of baseband controller IP.

• One of its kind because network simulator is never built as a hardware model.

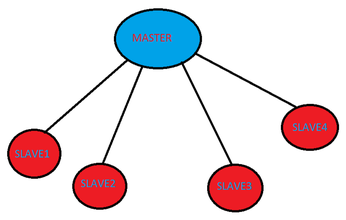
•It is supposed to be faster and efficient than other network simulators.

Modeling

Building blocks of the network simulator:

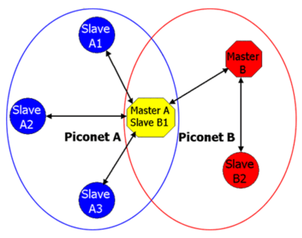
Piconet –

- This is called basic building block of a network simulator.
- A piconet is formed when at least two devices, such as a portable PC and a cellular phone, connect.
- A piconet can support up to eight devices.
- When a piconet is formed, one device acts as the master while the others act as slaves for the duration of the piconet connection.



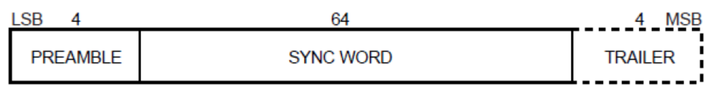
Scatternet –

- A **scatternet** is a type of ad hoc computer network consisting of two or more piconets (not considered in our implementation).



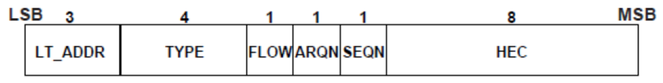
Bluetooth Packet:

1. Types of Access Code

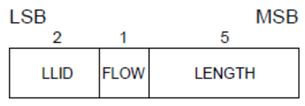


1. Device access code
2. Channel access code
3. Inquiry access code

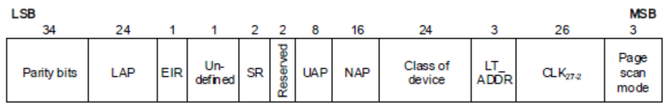
2. Header



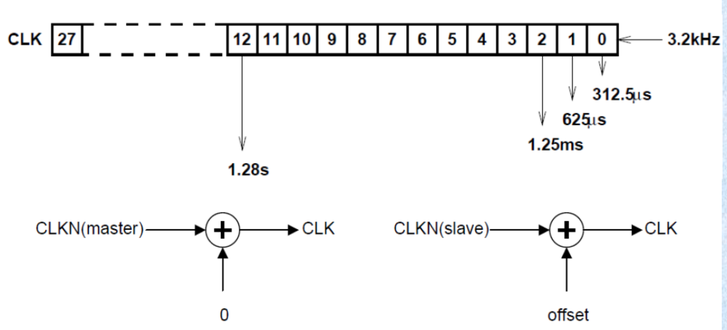
3. Payload



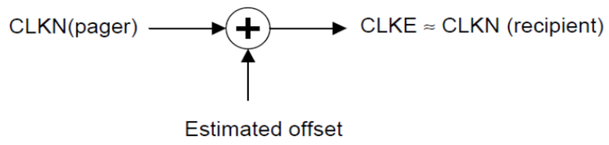
FHS Packet:



Bluetooth Clock:

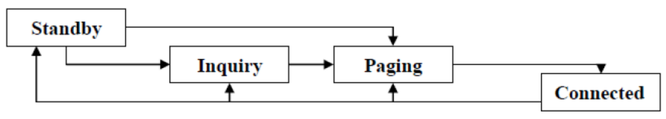


For successful establishment of piconet, master clock (CLK) is used. All activities regarding timings and scheduling in piconet is done with respect to this clock. All slaves use this master clock (CLK) to synchronize their clock.



During paging state, paging device or pager adds an estimated offset to make its CLKN equivalent to the native clock of the page scanning device (CLKE).

Bluetooth State Machine:



Results

Simulation Results:

There are total 6 devices in network. Since there is no Host here therefore test bench act as a host for all the devices. There are following flags through which these devices are controlled:

1. M<DEVICE>: If high then that device will try to become master
2. SI: If high then that device will initiate inquiry
3. IS: it notifies that device (slave) is scanning or listening
4. IR: it notifies that slave has received inquiry packets and sends FHS packet back on next slot
5. MTRANS: it notifies that master has started transmitting inquiry packets

The below table shows how all 6 devices change their state based on inputs given from test bench.

Inputs	Slot	Device 0	Device 1	Device 2
M0 = 0, M1 = 0 M2 = 0, M3 = 0 M4 = 0, M5 = 0		In standby state	In standby state	In standby state
M0 = 1 SI(dev0) = 1 Rest are same as above	Even	In Inquiry state	In inquiry scan state and listening at freq = 3	In inquiry scan state and listening at freq = 6
Value of FHS reached 3	Even	In Inquiry state	Received inquiry packet. IR (flag) = 1	In inquiry scan state and listening at freq = 6
	Odd	FHS packet received from dev 1	FHS packet transmitted to dev 0	In inquiry scan state and listening at freq = 6

Inputs	Slot	Device 3	Device 4	Device 5
M0 = 0, M1 = 0 M2 = 0, M3 = 0 M4 = 0, M5 = 0		In standby state	In standby state	In standby state
M0 = 1 SI(dev0) = 1 Rest are same as above	Even	In inquiry scan state and listening at freq = 5	In inquiry scan state and listening at freq = 4	In inquiry scan state and listening at freq = 9
Value of FHS reached 3	Even	In inquiry scan state and listening at freq = 5	In inquiry scan state and listening at freq = 4	In inquiry scan state and listening at freq = 9
	Odd	In inquiry scan state and listening at freq = 5	In inquiry scan state and listening at freq = 4	In inquiry scan state and listening at freq = 9

Conclusion

From the simulation results it can be seen that how master is initiating an inquiry and slave is responding back by sending FHS packet back to master. As this simulator is written in System Verilog therefore any baseband controller IP can be interfaced easily with this simulator. That IP can also be controlled through same test bench by adding additional signals related to that IP. Instead of choosing master from one of the six in built devices, one can make the above IP as a master. Then through simulation one can verify how that IP is interacting with other devices. This will give verification engineer an opportunity to verify their IP.

Key References

[1]. IEEE 802.15.1 (http://standards.ieee.org/getieee802/802.15.html)

[2]. http://www.isi.edu/division7/publication_files/advances_in_network

[3]. Bluetooth SIG (http://www.bluetooth.com/bluetooth/)

Acknowledgments

We want to thank Professor Morris Jones for providing invaluable suggestions, expertise and lab access.

For further information

Please contact *ankit.mehta@sjsu.edu & ujjwal.kant@sjsu.edu*. SV code, simulation files are available upon request.