



Developing Wireless High-Definition Video Modems for Consumer Electronics Devices

By Guy Dorman, AMIMON

Consumer electronics devices capable of delivering high-definition (HD) video signals are growing rapidly in number and variety. DVD players and cable or satellite set-top boxes have been joined by Blu-Ray players and video game consoles, with PCs, video cameras, and even mobile phones close behind. While each wave of innovation has improved video quality, it has also added to the tangle of cables connected to high-definition televisions (HDTVs) and created a need for more video inputs.

AMIMON has developed Wireless Home Digital Interface (WHDI™) technology that enables HDTVs to interface wirelessly to all HD video sources while providing the same quality as a wired connection. Developed using MATLAB®, WHDI technology can deliver 3 Gbps, enabling 1080p frames to be sent 60 times per second through walls to a device 100 feet away (Figure 1). Because the frames are uncompressed, latency is as low as 1msec, making WHDI ideal for game consoles that require extremely brief response times. WHDI devices not only eliminate the need for video cables; they can also act as a wireless switch, connecting multiple video sources to multiple HDTVs and displays.

We relied on MATLAB to verify our designs early in development by simulating the WHDI algorithms, and later by comparing the results produced by hardware implementations with MATLAB simulation results.

Challenges of Delivering Wireless, Uncompressed, High-Definition Video

From an engineering perspective, transmission chains for video are more demanding than transmission chains for data packets. Data packets can be retransmitted without affecting the user's

Products Used

- MATLAB®
- Optimization Toolbox™
- Signal Processing Toolbox™

experience. With video, there is no retransmission; each frame must be passed across the channel when the source delivers it. Because the viewer will notice any errors in the video, worst-case performance, not average performance, must be the gauge of quality.

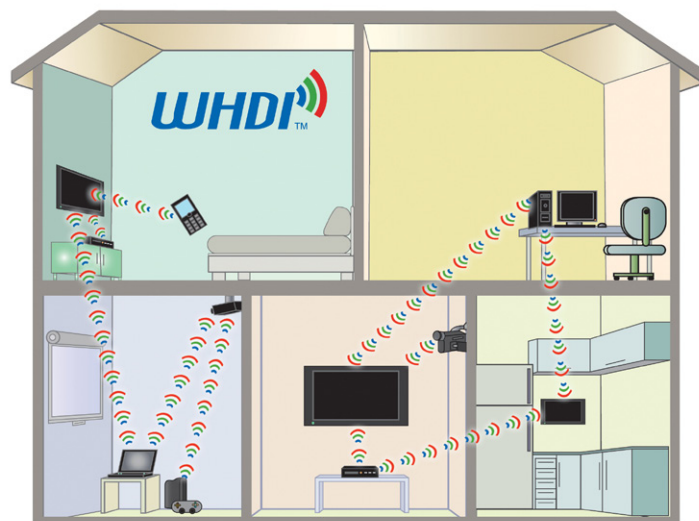


Figure 1. Schematic illustrating WHDI interface and communication capabilities.

Wireless video products that use alternative technologies have significant drawbacks. Some use compression, which not only adds a significant amount of latency but also requires a more powerful and costly processor. Others transmit in the very high frequency 60 GHz band, which requires direct line-of-sight between the transmitter and the receiver, and thus cannot be used to transmit from room to room. In addition, the 60 GHz technology does not support multicast transmission schemes in which one source is transmitted to multiple receivers.

For AMIMON engineers, the challenge was to develop video processing and modulation algorithms that would enable us to transmit uncompressed 1080p frames at 60 frames per second using the 5 GHz unlicensed band and just 40 Mhz of bandwidth.

Developing the Algorithms

Because the technology we were developing was so new, we needed to try innovative techniques. The MATLAB environment is ideal for this kind of development because we can very quickly try out new ideas and run simulations to evaluate them. As a result, we were able to try several schemes that would otherwise have been too time-consuming to evaluate. For example, we needed to design a motion classifier, which detects whether an 8x8 pixel block is static or dynamic. This straightforward implementation required the last frame to be kept in memory for comparison against the new frame. This frame buffer contains a large number of pixels. Using such a large memory buffer would increase the product cost significantly. By using MATLAB, we tried several schemes that store a signature

of each 8x8 pixel block. The signature from the last frame was used to compare against the signature from the current frame to determine whether motion was present. We tried several signatures. For each signature under test we ran simulations on real movies to quantify the false alarm rate and the miss detection rate.

The transmit and receive components of the system are partitioned into video processing and modulator modules (Figures 2a and 2b). The modulator must squeeze a great deal of data into a relatively narrow band of the spectrum. (The 40 Mhz bandwidth requirement is driven by FCC regulations and by our design goals of using a commonly available RFIC that operates at 5 Ghz with 40 MHz of bandwidth and allowing simultaneous transmission of many WHDI systems as well as WiFi systems in the same room.)

In MATLAB, we developed algorithms for the modulator that use multi-input, multi-output (MIMO) technology. With this design, we transmit four streams simultaneously using four separate antennas. On the receiving side we have five antennas, with the extra antenna used to compensate for fade-out. We tailored the video processing component to work with this special modulator.

The video processor and modulator are further partitioned into approximately 25 functional blocks, each modeled and simulated separately in MATLAB. For example, the video processing module includes discrete cosine transform (DCT) algorithms that process frames in 8-by-8 pixel blocks, while the modulator includes inverse fast Fourier transform (IFFT) blocks to implement the orthogonal frequency division multiplex (OFDM) modulation.

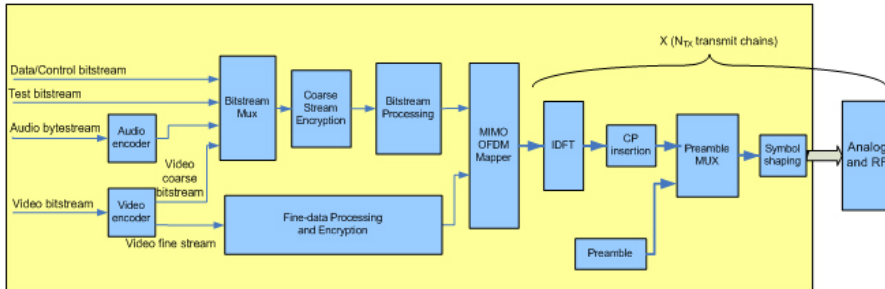


Figure 2a. DLPHY transmitter block diagram for 20 MHz mode and 40 MHz mode.

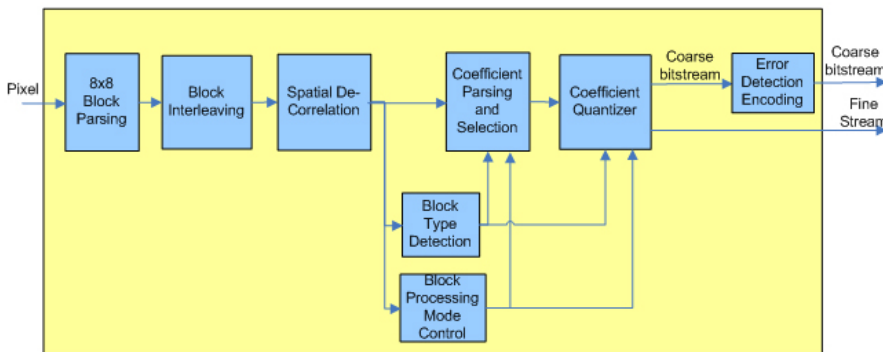


Figure 2b. Video encoder block diagram.

For another block we developed finite impulse response (FIR) filters using Signal Processing Toolbox™ and then used Optimization Toolbox™ to identify an optimal set of filter parameters that would bring the design into compliance with FCC regulations.

Simulation and Verification

Verification of our design started early. After developing the algorithms in floating point to maximize development speed, we converted them to fixed point for maximum simulation fidelity. Once a functional block was modeled with MATLAB, the engineer simulated the block by providing test stimuli inputs, capturing the output, and comparing the output with expected results.

A team of six engineers developed individual blocks in parallel until the entire system could be integrated and simulated together. The input to this full system simulation in MATLAB is a single-frame image stored as a bitmap file. The simulation performs all the video processing and modulation algorithms on the transmit side and then performs the reverse algorithms on the receiver side to reconstruct the image. At the end of the simulation we verify that the resultant image matches the input image with high fidelity. During simulation we capture the input and output of individual subsystems and reuse them to verify the RTL implementation.

Once a block has been tested via simulation in MATLAB, VLSI engineers use the MATLAB algorithm as a reference to develop the RTL code. Using a test-bench automation technology, the engineer then verifies the implementation by comparing the results produced by the RTL implementation with the test vectors generated

by MATLAB. The RTL block is stimulated with the same test inputs as the MATLAB code, and the results are analyzed to see if they are bit-exact matched.

In the next verification step we download the RTL onto an FPGA board and run the algorithms in real time on real channels using actual RF devices and antennas. On the receiver side we have a large memory bank in which we capture the input signals from the receive antennas. When we identify anomalies in the received signal, we analyze the data captured by injecting the received signal into the receiver MATLAB simulation and determining the source of the problem. This step enables us to improve our design to compensate for the kind of channel fades and noises that the device will encounter in real-life settings. Finally, we tape out the verify RTL implementation and send it for fabrication as an ASIC chip.

Second- and Third- Generation WHDI Technology

The second generation of WHDI devices is now entering production, and several leading consumer electronics manufacturers are incorporating the technology into their products. For third-generation devices, we plan to support 2k-by-4k video resolution and 3D technologies in video while reducing the device's cost and extending its range. We continue to use MATLAB to optimize the video processing and modulation algorithms and verify our ASIC implementation. We are also exploring opportunities to generate the RTL directly from MATLAB algorithms. ■

Resources

VISIT

www.mathworks.com

TECHNICAL SUPPORT

www.mathworks.com/support

ONLINE USER COMMUNITY

www.mathworks.com/matlabcentral

DEMOS

www.mathworks.com/demos

TRAINING SERVICES

www.mathworks.com/training

THIRD-PARTY PRODUCTS AND SERVICES

www.mathworks.com/connections

Worldwide CONTACTS

www.mathworks.com/contact

E-MAIL

info@mathworks.com

© 2010 The MathWorks, Inc. MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See www.mathworks.com/trademarks for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.

9187900 08/10

Learn More Online

- Automatic Hardware Implementation of Digital Filters for an Audio Codec
mathworks.com/audio-codec
- Philips Consumer Lifestyle Develops One-Piece Surround Sound System
mathworks.com/philips