How to succeed in tablet PC design

By Tim Saxe
Chief Technology Officer
QuickLogic Corp.

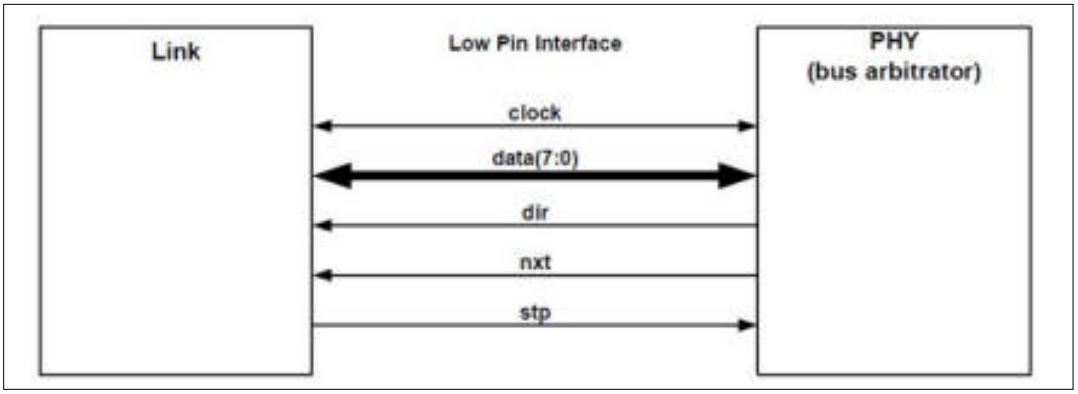

Figure: The ULPI interface developed for USB allows a small parallel bus connection between devices that behaves as a USB connection at the software link layer.

The growing interest in consumer and user-generated mobile video, along with increasing demand for higher resolution, is creating a challenge for tablet design. Devices will need to handle large amounts of data quickly in order to satisfy consumer expectations. Two keys to meeting this need are enhanced Universal Serial Bus (USB) connectivity and off-loading data tasks from the CPU.

Video is clearly emerging as a critical application for mobile devices such as tablet PCs. According to networking giant Cisco Systems, 40% of the mobile data bandwidth consumed in 2010 carried video. Further, the company expects that to grow to more than two-thirds of all mobile traffic by 2014.

Much of this traffic is user-generated. YouTube alone is responsible for nearly one-fifth of mobile data traffic. Streaming of consumer or copyrighted content such as mobile television and movies are another source. In addition, users are increasingly downloading full movies for off-line playback onto their tablet PCs.

Increases in video resolution are compounding this growing demand for video data bandwidth. Traditional VGA (640x480) resolution is already giving way to 720p for mobile devices. Tablets, with their larger display panels, will soon need to deliver full 1080p high definition video.

A second critical need for tablet devices is enhanced connectivity. Externally, consumers want to be able to link with home PCs for media downloading and file synchronization. They also want the option of adding keyboards, pointing devices, flash drives, memory cards, and data modems externally. Internally, developers need to provide the applications processor with connections to data ports, multiple radio modems for compatibility with various cellular and data networks, internal flash memory mass storage, and potentially many other peripheral devices.

The importance of video in mobile devices requires that this connectivity have as high a performance as possible. A movie download from a home PC or a WiFi network to a tablet, for instance, requires the transfer of 2 GB for standard definition and as much as 4 GB for a high-definition movie. Consumers have limited patience, so the faster a tablet can complete this transfer the user satisfaction. Users would also prefer to be able to use the tablet for other purposes during the transfer.

USB to the rescue

One key to addressing the video and connectivity needs of tablet PCs is effective leveraging of the USB. For the external connection, a dual-mode controller would allow the tablet to connect to peripheral devices as a host or to a PC as a device. Internally, a USB hub can serve many of the applications processor's connectivity needs. USB-based controller IP is widely available for flash disks, data cards, and radio modems, for instance, eliminating the need for the processor to provide separate connections or handle flash and modem interface protocols in software.

USB is particularly convenient as an interface to storage or peripherals handling video information. One of the approved data classes for USB transactions is the Media Transfer Protocol (MTP), which allows the USB link to support data transfers on a file basis. The traditional USB mass storage device class (MSC) works with fixed blocks, which may not be a match to the logical file size. Further, MSC provides access to storage on an undifferentiated bulk basis. In practice this means that the target device is under the host processor's absolute control for as long as the device is mounted and the device cannot modify any of its contents without risking data corruption.

The MTP, on the other hand, works on a logical file basis. The USB host shares access to the target device's file system, allowing the target to continue working with the files the host is not using. This allows a storage device to be involved in multiple operations, such as giving the user access to stored documents while the system is downloading a video file to memory. The MTP can also be run over TCP/IP and Bluetooth, allowing such file-oriented transfers to occur over wireless connections, as well.

USB without the PHY

One drawback of using conventional USB for is that the link is serial, which adds considerable latency in data transfers during command handshaking and as information moves between parallel and serial formats. A solution to his problem is the use of the UTMI+ Low Pin Interface (ULPI) (**figure**) without a physical layer (PHY) component. The PHY-less ULPI permits two devices to connect through an 8- or 12-line parallel bus while still appearing as the USB at the software link layer.

Use of the parallel bus to connect devices eliminates the overhead of serialization and greatly speeds handshaking between devices. Further, because the signals use today's low-voltage logic levels, ULPI eliminates the need for charge pumps or voltage converters to generate serial USB signals, reducing cost, parts count, and battery drain. These benefits of high performance, modest signal count, and simple connection to a wide range of peripheral device controllers are so compelling that ULPI is even beginning to appear on many embedded processors as an alternative to traditional wide local bus structures for peripheral connections.

While the USB in all its forms helps solve connectivity challenges in tablet design, though, the resulting data flow presents a significant performance burden. High speed USB allows transfers with a theoretical limit of 60 Mbytes/second but many PC host processors have only about half that bandwidth in and out of system memory. Further, the flash

memory technology used in mobile devices typically only achieves system memory bandwidths on the order of 8-10 Mbytes/second. Moreover, CPUs in mobile devices can introduce latency in transferring data between USB and mobile flash memory if they cannot provide a direct data path between the two.

The key to opening this bottleneck is to offload data-intensive tasks from the tablet processor. Providing a direct path from the USB controller to the data card controller and using MTP, for instance, would allow the downloading of a video file from the home PC to mass storage without host CPU intervention - a process known as side loading. This sideloading eliminates the need to first write to and then read from system memory during the file transfer. Sideloading thus both speeds the file transfer and frees the system processor to perform other tasks.

Offload DRM from CPU

Another task to offload from the applications processor is the encryption involved in secure data transactions such as digital rights management (DRM) for media file playback. Virtually all commercial digital media use some form of user-keyed encryption to prevent unauthorized file copying from producing playable files. Mobile devices must be able to decrypt such files on the fly during playback and an external security engine could offload that task from the tablet processor.

In addition to freeing the tablet processor, the use of an external security engine provides an extra measure of protection for digital rights management. Unless the CPU has a protected operating mode, handling the decryption entirely in software leaves open an opportunity for content pirates to reprogram the system CPU to decrypt and then copy the media file rather than pass it along to the playback software. An external hardware security engine would prevent such "hacking." The use of an external engine can also speed the certification process required by several DRM schemes. Rather than proving to the certification agents that their mobile design is acceptable, developers can simply insert a pre-certified building block.

Not all of that security engine should be in hardware, however. The decryption block itself can be readily implemented in hardware as it implements the well established advanced encryption standard (AES). But DRM involves more than just decryption. There are authentication protocols and key transfer processes involved, as well, and there is no unifying standard for this aspect of DRM. In Japan, for instance, the content protection for recorded media (CPRM) scheme is in widespread use. Windows Media Player and Apple's iPod use other schemes. Implementing this part of the DRM in firmware would allow developers to adopt one hardware design that could support different market needs with only a firmware change.

Implementing DRM, sideloading for data transfers, and otherwise offloading the system CPU is most effective if the various hardware elements are all integrated into a single device. With the USB and memory controllers on a single device, for instance, there are opportunities to optimize the sideloading path's performance by ensuring the controllers work together efficiently. Similarly, integrating the DRM processing with the sideloading hardware can streamline the entire media playback pathway so that the system processor has minimal involvement and is free to handle other tasks. The integrated DRM can even allow the playback of one movie while downloading another one through sharing of the encryption block.

The effective utilization of USB for both internal and external connectivity and the offloading of the system processor are thus two keys to success in tablet PC design. Together they will help ensure that the design fully meets user expectations for speedy media file transfers while supporting other user operations. They also help simplify and speed the development process while resulting in a flexible design that is readily adapted to localized markets and extended to incorporate new peripheral functions as market requirements continue their never-ending evolution.