Kondalarao.punati, Naresh kumar reddy.Beechu, G.Subramanya Sarma / International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com
Vol. 2, Issue 2,Mar-Apr 2012, pp.1436-1439

An efficient aggregation of multiple Ethernet frames for higher data rate

Kondalarao.punati*, Naresh kumar reddy.Beechu, G.Subramanya Sarma*****

*(Student of M.Tech, Department of ECM, K.LUniversity, India,)
** (Student of M.Tech, Department of ECM, K.L.University, India,)
*** (Assistant Professor, Department of ECM, K.L.University, India,)

ABSTRACT

Now a days universal serial bus has supported a wide variety of devices from keyboard, mouse, flash memory device. In addition, user applications demand a high performance network connection between the PC and network device . In this paper mainly focusing on high speed network device connects to host for exchanging Ethernet frames to achieve higher data rate through universal serial bus. Network Control Model Subclass is a protocol by which hosts and devices can exchange Ethernet frames efficiently. This is useful for high speed network attachments. This specification builds for the Ethernet control model devices, with improvements to support much higher data rates are Multiple Ethernet frames can be aggregated into single USB transfers and minimize overhead when processing the Ethernet frames within the USB device. Although there are configurations in which a high speed Linux device may need little more than Ethernet host side drivers. In this paper we are proposing new aggregation technique for higher data rate. We use the ARM-8 cortex processor in device side and also RS-232 to usb converter cable used for checking the data transfer details of the device in the host.

Keywords – **Embedded device, Linux pc, ARM 8-Cortex processor, USB 2.0 , Rs232 to usb converter cable**

I. INTRODUCTION

The impact of wireless communication and networking is obviously significant and growing. The design objective is to build the connection and data transfer between Host and Embedded Device . In this paper both USB[1] interface and RS 232 cable interface available, now a days these two cables are using world wide in various fields.USB cable is used to connect the Host to device for data transfer between host and device and vice versa. where as RS232 to USB cable is used to check the data transfer details of Device in the Host. Over the last decade, the applications of wireless and cellular devices have expanded rapidly. To allow the appropriate class driver to manage that data, the host is presented with one or more interfaces. In Ethernet control model[2]specification, either host or device send single packet transfer through the universal serial bus. By transferring single packet each time, the time taken by the processor increases, throughput decreases and over head on USB increases. Network control model specification will be the better choice than the Ethernet Control Model. In this paper mainly focusing on how the network devices connected to host for higher data rates .Here we will discuss how the data transmitted through USB and how the Ethernet frames aggregation at sender and receiver and vice versa. Network control model Subclass is a protocol by which USB hosts and devices can exchange Ethernet frames eficiently.NCM is intended to be used with high speed network attachments. This specification builds with improvements to support much higher data rates are Multiple Ethernet frames can be encapsulated into single USB transfer. As the network operators prepare to roll out LTE and WIMAX communications, handset design are turning to network communications models to outperform yesterdays modem based data communications. Although there are configurations in which a high speed Linux device may need little more than Ethernet host side drivers. By using Linux operating system to write the code for host and device and connect each other using universal serial bus. In below sections will explain the block diagram and aggregation mechanism. In aggregation mechanism two formats of transfer is possible, those are 16 bit network control model transfer block and 32 bit network control model transfer block.

II.BLOCK DIAGRAM

The connectivity between host and device shown in figure1. In this section mainly explains personal computer act as Host. Before applications can communicate with a device, the host needs to learn about the device and assign a driver. Enumeration is the exchange of information that accomplishes these tasks. The process includes assigning an address to the device, reading descriptors from the device, assigning and loading a driver, and selecting a configuration that specifies the device's power requirements and interfaces. The device is then ready to transfer data.

The process by which a USB Host examines a USB Device before enabling it for use. The host reads various descriptors[3] from the device and uses information from them to match the device with possible USB Drivers. Once a driver has accepted the device, the host performs a set configuration (enables the device), and the device becomes configured and ready for use.

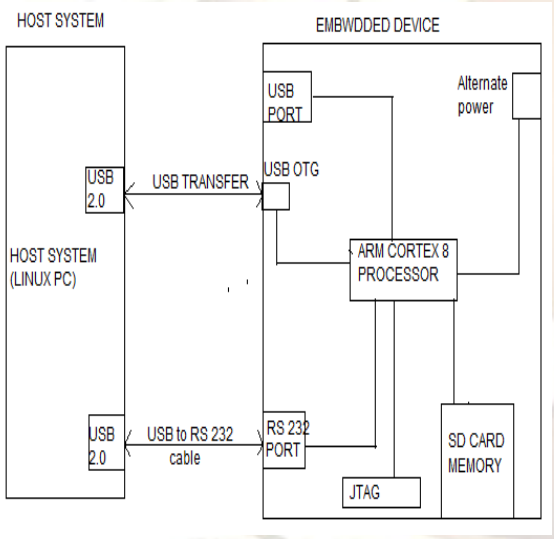

Fig .1.Block diagram

1.Embedded device :

Device is a logical or physical entity that performs a function. The actual entity described depends on the context of the reference. At the lowest level, device may refer to a single hardware component. At a higher level, it may refer to a collection of hardware components that perform a particular function, such as a USB interface device. At an even higher level, device may refer to the function performed by an entity attached to the USB. In figure 2 shows that the Device structure. The Device also works like computer. The Device implementation in this paper is composed of POP CPU/memory chip. Processor TI DM3730 Processor-1 GHz ARM Cortex-A8 processor[4],Imagination technologies power VR SGX 2D/3D graphics processor supporting dual independent displays,512 MB LPDDR RAM, 4 GB micro SD card, Peripheral connections are DVI-D (HDMI connector chosen for size - maximum resolution is 1280×1024),S-Video, USB OTG (mini AB), 4 USB ports, Ethernet port, RS-232 port, JTAG connector, Power socket (5 V barrel connector type),Camera port and Expansion port.

Figure 2. Embedded device diagram

2. NCM Function

An NCM function is collection of one or more interfaces in a usb device. NCM Communications Interface and NCM Data Interface are two types .The NCM Communications Interface is used for configuring and managing the networking function. The NCM Data Interface is used for transporting data, using the endpoints defined by that interface. Generally, the NCM Communications Interface and the NCM Data Interface are managed by a single driver on the USB host. The logical connections between host driver and NCM function are shown in figure 3.

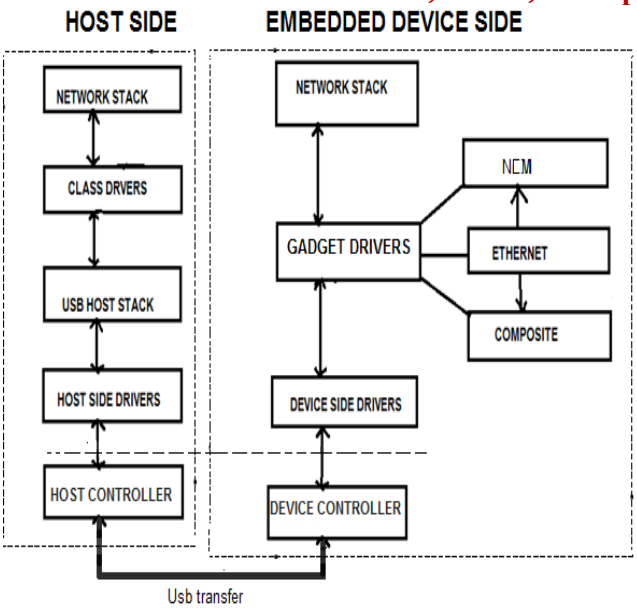

Fig 3. NCM function

3. Descriptors:

Descriptors[3] are the data structures that enable the host to learn about a device. Each descriptor contains information about the device as a whole or an element in the device. All USB devices must respond to requests for the standard USB descriptors. The device must store the contents of its descriptors and respond to requests for the descriptors. In NCM we have two different descriptors one is communication interface descriptors and data interface descriptors. The Data Interface of an NCM networking function shall have two alternate settings. The first alternate setting (the default interface setting, alternate setting 0) shall include no endpoints and therefore no net-working traffic can be exchanged when the default interface setting is selected. The second alternate set-ting (alternate setting 1) is used for normal operation, and shall include one bulk IN endpoint and one bulk OUT endpoint.The interface descriptors for alternate settings 0 and 1 shall have bInterfaceSubClass set to 0, and bInterface-Protocol set to 01h.

III.AGGREGATION TECHNIQUE

While writing the driver programming for the device and host, dynamically allocate a buffer for NCM transfer block in the wrap function. Each frame comes from the network stack, one after another aligned by using network control model protocol. This section explains how the Ethernet frames are aggregating. Network control model allows device and host efficiently transfer one or more Ethernet frames using single usb transfer. The usb transfer is formatted as NTB. It begins with network transfer header this identifies the transfer as an NTB and provides information to receivers.NTH points to head of list of network datagram pointers and each network datagram pointer(NDP) points multiple Ethernet frames encapsulated with in the NTB. Datagram is collection of bytes forming a single item of information. Figure shows us the aggregation mechanism. The maximum size of an Ethernet frame datagram can be dynamically adjusted by the host using the Set-MaxDatagramSize request. Commonly, Ethernet frames are 1514 bytes or less in length (not including the CRC), but for many applications a larger maximum frame size is needed . Hosts can discover the maximum Ethernet frame size supported by a device from the value wMaxSegmentSize in the Ethernet Networking Functional Descriptor, and shall not select a size larger than the device can support, nor shall it send a frame larger than the device can support. Host or function may append CRCs to datagrams. These four-byte CRCs are not included when deter-mining the “maximum segment size”, but they are counted when specifying the datagram size in the NDP. For example, if the maximum datagram size is currently 1514, and the NDP header indicates that CRCs are being appended, then the maximum wDatagramLength value for any datagram in that NDP is 1518.

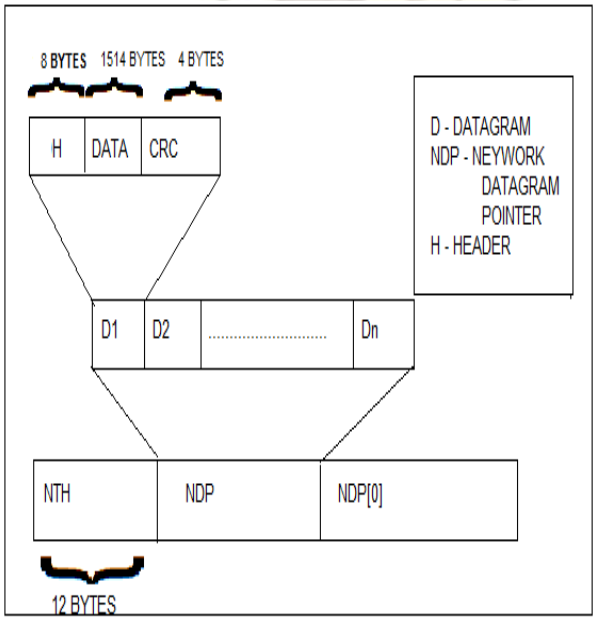

Fig 4. Aggregation mechanism

Any NCM Datagram pointer entry with an offset field of zero or with a length field of zero, or with both offset and length field set to zero, shall be treated as a Null entry. Receivers shall process datagram pointer entries sequentially from the first entry in the NTB. The first Null entry shall be interpreted as meaning that all following NCM Datagram Pointer Entries in the NDP are to be ignored. Transmitters are allowed to send a properly formatted NTB containing an NDP whose datagram pointer entries are all zero. Receivers shall ignore such NTBs. It is an error for a transmitter to send an NDP with non-Null NCM Datagram Pointer Entries following the first Null. Receivers MAY process datagrams up to the first Null NCM Datagram Pointer Entry, and MAY ignore the remaining non-Null entries in the NDP.

IV.SYSTEM TESTING

After the connections completed between host and device, the ultimate generated codes are compiled and downloaded to the target system for testing. In the testing, PC is used as Host which is connected with USB cable to the device respectively. The RS232 cable is used to connect the host from device. When the IP address of host is inputted by ping command, ifconfig and iperf commands the results show that the how the data transfer between embedded device and host in the system console. The higher data rate will be obtained and the down link speed is high then compare to uplink speed. The test results on system console appears that the communications from both sides are successful.

V. CONCLUSION

In this paper Aggregation of multiple Ethernet frames for higher data rate through universal serial bus efficiently is possible by using the network control model technique. By using this specification also reduce overhead on universal serial bus and increase the processor performance. In test results we will get much data rate. Mainly this paper will be useful in the wireless modem like future generation networks. We can achieve data rate up to 100 Mbps as down link speed and 50 Mbps as up link speed. Our future work will be the routing mechanism for the Embedded device to network connection device.

ACKNOWLEDGEMENT

We thank to our principal, Prof. K .Rajasekhar Rao, for providing necessary facilities towards carrying out this work. We acknowledge the diligent efforts of our Head of the Department Dr.S.Balaji in assisting us towards implementation of this idea.

REFERENCES

[1]. Universal serial bus specification2.0,,USB complete fourth edition.“THE DEVELOPERS GUIDE” by John Axelson.

[2]. Ethernet control model specification by CDCECM verson1.0...1999.

[3]. Types of descriptors form the “Device Class Definition for Human Interface Devices (HID)”.. Firmware Specification—6/27/01.

[4]. ARM-PROCESSORS is based on the ARMv7 architecture www.arm.com/products/processors/cortex-a/cortex-a8.php .

[5]. USB data pump® Embedded host &On to go software.

[6]. Encapsulation of multiple Ethernet frames for higher data rate by Ethernet emulation model..FEB 2012 from IJAST.