TRIGGERING SYNCHRONIZATION OF MULTIPLE USB 3.0 DEVICES USING ISOCHRONOUS TIMESTAMP PACKET

LIM CHONG HAN

UNIVERSITI SAINS MALAYSIA

2016

TRIGGERING SYNCHRONIZATION OF MULTIPLE USB 3.0 DEVICES USING ISOCHRONOUS TIMESTAMP PACKET

by

LIM CHONG HAN

Thesis submitted in fulfilment of the requirements
for the degree of
Master of Science

May 2016

ACKNOWLEDGEMENT

Along the way of working out this project, there are a lot of people helping me to complete this project. With their guidance and encouragement, I managed to finish the project within the duration given.

First and foremost, I would like to express my deep gratitude to my project supervisor and advisor, Dr. Bakhtiar Affendi bin Rosdi. He had given the great effort in guiding and helping me in completion of this project and thesis. His professional advises and comments on this project had contributed a lot to the successfulness of my project. His patience and guidance from time to time of conducting scientific investigations related to this project are much appreciated.

Correspondingly, my special thanks and heartiest appreciation to Mr. Jeff Kellam, Mr. Yap Chee Fai, Mr. Ng Chow Keng and Madam Azian Wahab who are the researchers from National Instrument Malaysia, for their professional advices and comments on this project.

Besides that, I would like to extend my thanks to National Instruments Company for providing me all the facilities throughout my research including measurement devices.

Additionally, a heartfelt thanks to Collaborative Research in Engineering Science & Technology (CREST) for providing funding for this project with the grant account number of 304/P-ELECT/6050266/C121.

Last but not least, I would like to thanks my family and friends in encouraging me and given me mentally support during the completion of the project.

TABLE OF CONTENTS

APPENDICES

LIST OF TABLES

Page

LIST OF FIGURES

Page

LIST OF ABBREVIATIONS

ATE	Automated Test Equipment
CAST	Corsair Avionics System Tester
CBIV	Current Bus Interval Value
FLCS	Flight Control System
GPIB	General Purpose Interface Bus
HID	Human Interface Devices
IEEE	Institute of Electrical and Electronics Engineers
ITP	Isochronous Timestamp Packet
LAN	Local Area Network
LXI	LAN eXtension for Instrumentation
PC	Personal Computer
PCI	Peripheral Component Interface
PFI	Programmable Function Interface
PPM	Parts Per Million
PTP	Precision Time Protocol
PXI	PCI eXtensions for Instrumentation
SBIV	Specific Bus Interval Value
SOF	Start of Frame

T&M	Test and Measurement
USB	Universal Serial Bus
USB-IF	USB Implementer Forum
VI	Virtual Instrument
VME	Versa Module Europa
VXI	VME eXtensions for Instrumentation

PENYEGERAKAN PEMICUAN BERBILANG PERANTI USB 3.0 MENGGUNAKAN BUNGKUSAN CAP WAKTU SEMASA

ABSTRAK

Dalam pengujian dan pengukuran, penyegerakan berbilang peranti bagi penyelarasan pemerolehan data adalah sangat penting. Ini adalah kerana saluran yang terhad dalam satu peranti dan perlunya saluran isyarat bercampur. Kini, pelbagai teknik penyegerakan berbilang peranti telah dicadangkan seperti GPIB, VXI, PXI/PXIe, dan LXI. Tetapi, kosnya mahal dan memerlukan ruang besar dan kabel atau perkakasan tambahan untuk pemasangan. Oleh itu, USB 2.0 dengan harga yang murah dan senang untuk dipasang telah dipertingkatkan untuk mempunyai keupayaan penyegerakan. Terdapat beberapa kaedah penyegerakan dalam USB 2.0 seperti "software-timed trigger", "external hardware trigger" dan "USB-inSync". "Software-timed trigger" hanya diprogramkan dalam perisian dan mempunyai kelewatan di peringkat mikrosaat. Selain itu, kelewatan bagi kedua-dua "external hardware trigger" dan "USB-inSync" adalah sehingga peringkat nanosaat. Tetapi, mereka memerlukan perkakasan dan pemasangan tambahan untuk menyegerakkan berbilang peranti. Bagi mengatasi kelemahan teknik-teknik semasa, satu algoritma pemicuan berdasarkan teknologi USB 3.0 telah dicadangkan. Bungkusan Cap Waktu Semasa dalam USB 3.0 telah digunakan dalam algoritma pemicuan ini. Tanpa apa-apa tambahan perkakasan, algoritma pemicuan ini boleh memicu berbilang peranti USB 3.0 dalam masa yang sama dengan kelewatannya sekitar 280 nanosaat. Di samping itu, algoritma pemicuan ini mampu menyokong maksimum tiga hab USB 3.0 yang disambung berturut-turut dengan tidak menjejaskan prestasinya. Secara ringkasnya, algoritma pemicuan ini telah menambah baik kelewatan sebanyak 99.7% daripada "software-timed trigger" dan tanpa apa-apa tambahan perkakasan ia lebih senang dipasang berbanding dengan "external hardware trigger".

TRIGGERING SYNCHRONIZATION OF MULTIPLE USB 3.0 DEVICES USING ISOCHRONOUS TIMESTAMP PACKET

ABSTRACT

In test and measurement, synchronizing multiple devices to timely coordinate data acquisition is crucial. This is due to the limited number of channels on a single device and the need for mixed-signal channels. Currently, various techniques have been proposed for triggering and synchronization among multiple devices such as GPIB, VXI, PXI/PXIe and LXI. However, they are expensive and need large footprint and additional cable or hardware to set up. Due to that, with the low cost and easy-to-setup USB 2.0, it is enhanced to have synchronization ability. There are various synchronization methods in USB 2.0 such as software-timed trigger, external hardware trigger and USB-inSync. The software-timed trigger is purely implemented in software and has high trigger delay, which is up to microseconds. Besides that, for both external hardware trigger and USB-inSync, they can achieve up to nanoseconds of synchronization precision. However, both of those methods need extra hardware or setup to synchronize multiple devices. In order to overcome these drawbacks from the existing techniques, a trigger algorithm is proposed based on the study of USB 3.0 technology. The Isochronous Timestamp Packet (ITP) in USB 3.0 is used as the main component in the trigger algorithm. Without any hardware implementation, the trigger algorithm is able to trigger multiple USB 3.0 devices with a trigger delay around 280 ns. In addition, the trigger algorithm can support up to maximum three consecutive connected hubs without affecting the performance. In short, the trigger algorithm had improved the trigger delay from conventional software-timed trigger by 99.7% and without any hardware implementation it is easier to set up compared to external hardware trigger.

CHAPTER 1

INTRODUCTION

1.1 Background

Nowadays, detection and measurement on physical properties, electrical signals and medical systems in test and measurement (T&M) become very important. This is because improvement can be made from the measured data to achieve better output and provide better quality of end-product (WÅĆodarczyk et al., 2014). Analytical instruments and equipment such as data acquisition devices (LIU and ZHANG, 2013), sensors (Watanabe et al., 2011), product simulation system, analytical laboratory instruments and diagnostic equipment (NAGATA et al., 2012) are the examples that are used in T&M. T&M is widely employed in every industry including military, aerospace, telecommunication, automotive, medical, energy, semiconductor, consumer electronics and so on (Semancik, 2005). The purposes of T&M used by these industries are for their design, manufacture as well as deployment, and also to ensure the product's quality in order to increase the customer's satisfaction.

In T&M, data acquisition is a process to collect and gather data from testing and measuring instrument(s). Recently, due to the limited number of channels provided in a single device and the need for mixed-signal channels, synchronization of data acquisition in multiple devices becomes very important (Instruments, 2013a). Without the synchronization, data acquired from different channels or instruments will not be timely coordinated (Lenzen et al., 2010). Therefore, the test data become less accurate. To overcome this problem, triggering and synchronization methods are implemented into data acquisition instruments (Hirai and Satoh, 1980). In consequence, tests in different channels or instruments can be triggered at the same time.

Many types of methods have been proposed for triggering and synchronization among multiple devices. In 1972, General Purpose Interface Bus (GPIB), which was designed by Hewlett Packard, was brought forward to realize the clock synchronization and triggering of multiple instruments (Honglei et al., 2010). In 1987, a new standard of instrument architectures - Versa Module Europa (VME) eXtensions for Instrumentation (VXI) is introduced. This new architecture is to address the limitation of GPIB and further promote a standard not only for test and measurement, but for other industries as well (Greenberg, 2006).

Besides that, with the purpose to increase the rate of data transfer and to be more compatible with personal computer (PC), the peripheral component interface (PCI) extension for instrumentation (PXI) based on the PCI bus was proposed in 1997 and was launched in 1998 (Ullrich, 2007). Tight timing and synchronization is implemented into PXI to provide faster data transmission and higher precision of clock synchronization. Furthermore, followed by the ubiquity of internet technology, synchronization of multiple measurement devices through the network becomes possible. In 2004, the local area network (LAN) extension for instrumentation (LXI) bus based on LAN was proposed and was released in 2005 (Proft, 2007). LXI is based on the well-established Ethernet technology to network measurement devices and can embed multiple devices seamlessly into the available LAN in the lab or office. LXI applied Precision Time Protocol (IEEE-1588, 2008) which was defined in IEEE 1588 standard to synchronize clock among multiple devices.

On the other hand, due to easy-to-use and hot-plug-and-play of Universal Serial Bus (USB), it had been used into industrial application as a low cost data acquisition device. USB is introduced in 1995 and is maintained and developed by a non-profit organisation - USB Implementers Forum (USB-IF). Currently, there are five versions of USB introduced such as USB

1.0, USB 1.1, USB 2.0, USB 3.0 and the latest USB 3.1 (Axelson, 2015). The USB 2.0 is enhanced to have synchronization capability and the precision can be as high as up to picoseconds level (Foster et al., 2007).

In this project, a new algorithm is proposed to trigger multiple USB 3.0 devices at the same time. The proposed algorithm is designed based on USB 3.0 technology; therefore, it is only working in USB 3.0 and above. Additionally, the performance of the proposed algorithm is evaluated and discussed.

1.2 Problem Statement

In this modern age, many experiments require measurements of various physical quantities in one test (WÅĆodarczyk et al., 2014). There is the problem of limited channels in a single device to acquire multiple types of data. To solve this, multiple measurement devices are used together to acquire multiple data (Honglei et al., 2010). Due to that, data from multiple measurement devices need to be synchronized so that the data will be timely coordinated and processed at the same time. With data synchronization, data acquisition becomes more accurate and precise. Therefore, the quality of products or researches can be increased.

There are numbers of test and measurement buses are introduced to synchronize multiple instruments such as GPIB, VXI, PXI/PXIe and LXI. However, even though these buses provide high precision of synchronization, they are either expensive, take longer time to setup or require large footprint (Drenkow, 2005). Due to that, the cheap and easy-to-use USB becomes popular and is enhanced to support synchronization ability (Foster et al., 2007).

There are three trigger and synchronization methods introduced for USB such as USB-inSync (Foster et al., 2007), software-timed trigger and external trigger (Instruments, 2012b; Optics, 2014). USB-inSync provides up to picosecond levels of synchronization precision.

However, extra circuits and hardware are needed to implement this method and this increases the cost and time to produce it. Besides that, software-timed trigger uses software application to send a trigger signal to the device to trigger an event. Trigger signal is sent to each connected device one by one. Therefore, it has higher delay and low synchronization precision. On the other hand, the external trigger uses external signal such as power source to trigger an event in a device. However, this method needs to set up an extra circuit to supply the external trigger source to device.

Due to the drawbacks of each method in USB triggering and synchronization, a new algorithm based on USB 3.0 technology is proposed in this research to trigger multiple USB 3.0 devices at the same time. In this proposed algorithm, the new added Isochronous Timestamp Packet (ITP) in USB 3.0 is utilized to trigger multiple USB 3.0 devices at the same time.

1.3 Objectives

The objectives of this research are listed below:

1. To propose and develop an algorithm to synchronize multiple USB 3.0 devices to trigger them at the same time using Isochronous Timestamp Packet.
2. To evaluate the performance of the proposed algorithm by comparing it to previous synchronization algorithm used in USB and in various number of USB 3.0 hubs.

1.4 Scope of the Project

This research develops a new method of triggering and synchronization for USB devices. USB 3.0 technology is used for this research to create a new algorithm to synchronize multiple USB 3.0 devices so that they can start generate/acquire data at the same time. Both USB 3.0 host and device are used to test the proposed algorithm. The proposed algorithm is implemented in software only. A user interface host application is created using C Sharp (c#) programming language. The device firmware is created to communicate with host. The proposed algorithm is evaluated by comparing it to previous techniques used in USB such as software-timed trigger and external hardware trigger (Instruments, 2012b). Comparison is done based on trigger delay between each connected devices on the same host. Besides that, the performance of proposed algorithm is also tested in different USB hub depths.

1.5 Research Contribution

This research will contribute a new trigger algorithm for USB 3.0 system to trigger multiple connected USB 3.0 devices at the same time. The trigger algorithm does not need any implementation of extra circuit or hardware in USB 3.0 host, hub and device. It is implemented in software only which in both host application and device's firmware. Therefore, setup time and cost can be reduced.

1.6 Thesis Outline

The remaining chapters of this thesis are presented in four chapters and each chapter is summarized as follows.

Chapter 2 covers literature findings of current triggering and synchronization methods in test and measurement followed by the revision of USB 3.0 architecture.

Next, Chapter 3 discusses the methodology of the proposed algorithm in procedural steps from the concept of proposed algorithm to implementation of the proposed algorithm. In addition, the experimental setups for the performance assessment of the proposed algorithm are also explained in this chapter.

Meanwhile, Chapter 4 explains the results of the proposed algorithm. The performance of the proposed algorithm is evaluated by comparing it to existing synchronization methods used in USB. The performance of the proposed algorithm in multiple USB hub depths is also discussed.

Finally, Chapter 5 concludes this project and suggests the direction of future work.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter reviews the current studies and also the latest findings related to this research. First, the triggering and synchronization of each test and measurement bus will be explained briefly followed by comparison between them. Next, the existing triggering and synchronization methods used in USB 2.0 are explained. Then, the USB 3.0 bus, which will be focused on in this research is explained. Finally, the summary of this chapter will be given in last section.

2.2 Triggering and Synchronization in Test and Measurement

Synchronization is a process to coordinate two or more events, processes, or devices in time. These events from multiple equipment are synchronized by simultaneous triggering. Therefore, they can be started, stopped and processed at the same time. To do this, a common triggering control signal, which is a time-specific indication for a particular event to be performed, is sent to all involved equipment in a test and measurement system (Dinteman, 1993). START and STOP Trigger are usually the two components used to start or stop an event in a system. START Trigger has the ability to start a data acquisition system. In contrast, STOP Trigger has the ability to stop a data acquisition system.

Besides that, the synchronization is implemented into many instrumentation bus standards in test and measurement. There are four famous and widely used instrumentation bus standards for test and measurement which are General Purpose Interface Bus (GPIB), Versa Module Europa (VME) eXtensions for Instrumentation (VXI), Peripheral Component Interconnect (PCI) eXtensions for Instrumentation (PXI), Local Area Network (LAN) eXtension for Instrumentation (LXI) and each of them is explained briefly in the following sub-section.

2.2.1 General Purpose Interface Bus (GPIB)

GPIB is introduced in 1970s and has become the primary instrument control interface for about 30 years (Jia et al., 2009). In 2005, it is estimated that there are over 10, 000 different GPIB instruments available (Drenkow, 2005). GPIB is standardized by the Institute of Electrical and Electronic Engineers (IEEE) as IEEE 488 standard. The new version of IEEE 488.2 standard covers syntax in general and controller universal commands in specific (Mussmann, 1988). In addition, GPIB is an 8-bit full-duplex communications bus with 8 MByte/sec transfer rate. It can support up to 15 devices, which are connected together including the controller (Gilbert, 1982). The maximum length of cable in a normal bus network is either 2 meters times the number of devices on the bus or 20 meters, whichever is less (Gilbert, 1982).

Besides that, GPIB integrates synchronization ability to synchronize multiple instruments with its Group Execute Trigger (GET) (Drenkow, 2006). All the programmed instruments will be triggered to start an event when the GET line is pulled. Figure 2.1 shows the GET synchronization in GPIB. When the GET line is pulled, all the instruments connected to this GET line will be triggered. This method of GPIB gives a latency of approximately 5ns per meter of GPIB cable (Drenkow, 2006). Therefore, the shorter the wire connected between devices, the higher the precision of triggering.

On the other hand, external trigger, also known as hardware trigger, is also used in GPIB (Drenkow, 2006). An external trigger signal is supplied to each connected GPIB instrument one by one. Each GPIB instrument detects the signal; if there is a trigger signal, each instrument starts an event. The latency of this method is up to 5 ns, which depends on cable length

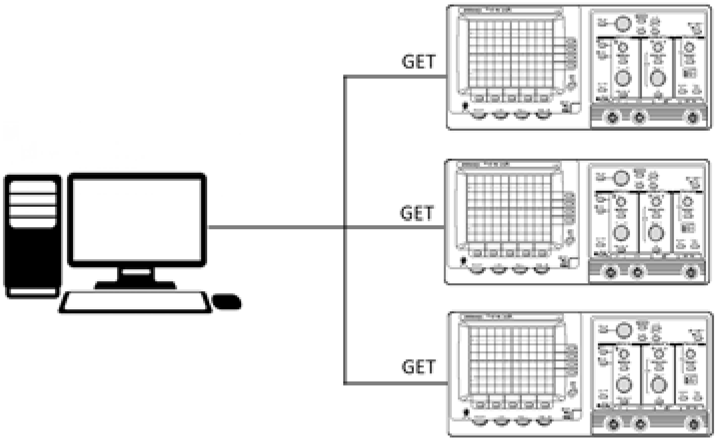

Figure 2.1: GET synchronization in GPIB

(5ns/meter). Besides that, there is another trigger method in GPIB which is software trigger. A trigger signal is sent by a PC to the instrument to trigger an event. However, it has higher latency than the previous two methods as the trigger signal travels from PC to instrument and takes up to one millisecond to each instrument.

GPIB is a short-range digital communication cable standard which provides a low bandwidth bus with only 8MByte/sec. It is brought forward to realize the clock synchronization and triggering of multiple instruments (Honglei et al., 2010). It can provide up to nanoseconds of synchronization between instruments. However, GPIB base instruments are rack and stack system; therefore, they use up much space to locate them, large footprint is needed to set up the test system. Besides that, the cost to build a GPIB system is very expensive compared to other I/O standards.

2.2.2 Versa Module Europa (VME) eXtensions for Instrumentation (VXI)

In 1987, a new standard of instrument architectures - VXI, is introduced to address the limitation of GPIB (Greenberg, 2006). VXI was proven to be an ideal solution for high channel count applications that require extremely tight synchronization between multiple instruments and mainstreams (Sarfi, 2006). For example, VXI is applied on Flight Control System (FLCS) in

F-16 (Roosendaal, 1996) and the US Air Force's A-7 Corsair Avionics System Tester (CAST) program (Singh, 1991).

VXI is based on VME bus, by adding extensions for instrumentation and maintains the modular systems approach of VME bus (Dettmer, 1989). VXI system adds additional clock bus and trigger bus to provide very tight timing and synchronization for instrumentation. VXI clock bus provides two system clocks and a clock synchronization signal - a 10 MHz clock (CLK10), a 100 MHz clock (CLK100) and a synchronization clock (SYNC100). All of these clocks are ECL signals and the CLK100 is always synchronous to CLK10. In addition, they are all sourced from Slot 0 (VXI Controller) and buffered on the backplane of the VXI chassis. On the other hand, CLK100 is guaranteed to be time-matched to within 2ns from slot 1 through 12 (Chipperfield, 1989) in VXI chassis. Besides that, SYNC100 is used to synchronize multiple modules to a given CLK100 rising edge.

There are two categories of synchronization in VXI (Emmert, 1998). The first category is to use the standard clock for performing measurements. The 10 MHz system clock can be used by each module card in the VXI chassis but it is not necessary to be used by the card. Therefore, many system or card manufacturers will produce their own internal clock to their card architecture. Besides that, the second category uses the VXI trigger line. In this category, the card must support this trigger's capability to use the trigger function (Chipperfield, 1989). Examples of the VXI trigger line protocol are STST (Start/Stop) protocol and SYNC trigger protocol.

The STST is simple as it runs the activity when trigger line is LOW, while it stops when trigger line is HIGH. This timing is very precise as the propagation delay between each slot is fixed at only 2 ns (Chipperfield, 1989). Therefore, the propagation delay can be easily calculated and eliminated. Figure 2.2 shows an example of synchronization using STST. Assume

A is the controller at slot 0 and it sends trigger signal to device B and C. B is located at slot 2 and C is at slot 12. The trigger signal takes 24 ns to reach C, so B needs to delay internally of 22 ns. Therefore, the event will only be triggered in both B and C at the same time after C received the trigger signal. On the other hand, the SYNC trigger protocol broadcast trigger does not require any acknowledgement from any acceptors. It uses a single pulse on one ECL trigger line and is asserted low to trigger all instruments at the same time.

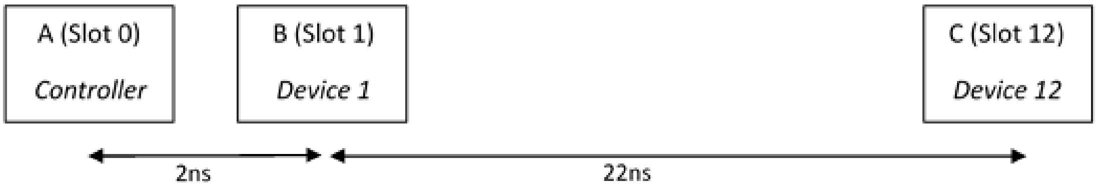

Figure 2.2: Example of synchronization of two devices using STST in VXI

VXI is a highly integrated and general-purpose ATE system. It provides high performance for higher count channel and solves the problem introduced in GPIB bus such as footprint, cost and cabling are reduced or eliminated and the hardware interoperability and software compatibility are improved (Pye, 1992). Besides that, it also improves the transfer rate to about 40 Mbyte/s in comparison to GPIB, which is only 8Mbyte/s (Schmalzel et al., 1992). Now, with the introduction of 64-bit VXI, the speed is improved up to 80 Mbyte/s. Additionally, with Two Edge VME in VXI, throughput is doubled to up to 160 Mbyte/s (Greenberg, 2006). On the other hand, it also provides very tight synchronization which is up to nanoseconds precision of synchronization. However, VXI is based on the older VME bus and it is not a part of modern computer architectures. Therefore, it cannot take complete advantage of the advances in modern PC technology and thus bring the benefits of mainstream software, lower cost, and high performance to the end user.