

IIoT를 위한 IEEE 802.1AS MAC 설계 및 구현

윤기하*, 김근용*, 김량수*, 손병희*, 유학*

Design and Implementation of IEEE 802.1AS MAC for IIoT

Gi-Ha Yoon*, Geun-Yong Kim*, Ryang-Soo Kim*, Byung-Hee Son*, Hark Yoo*

요약

본 논문에서는 산업용 사물인터넷 장치를 제어하고 모니터링하기 위한 차세대 네트워크 기술로써 최근 활발히 연구되고 있는 TSN(Time-Sensitive Network)의 IEEE 802.1AS 고정밀 시각동기 기능을 지원하는 MAC 설계 및 구현에 관한 내용을 기술한다. 본 논문에서 기술하는 IEEE 802.1AS 시각동기 지원 MAC은 ALTERA SoC FPGA기반으로 이더넷 MAC과 Time-of-Day 모듈로 구성되며, SoC FPGA에 내장된 ARM 프로세서에서 해당 하드웨어 블록을 제어할 수 있도록 설계하였다. 또한, Linux OS(ubuntu 16.04 LTS)에서 IEEE 802.1AS 시각동기 지원 MAC 디바이스 드라이버 소프트웨어를 구현하였으며, PTP(Precision Time Protocol)에 대한 오픈소스 프로그램인 LinuxPTP를 통해 시각동기 정확도 성능을 측정하였다. 본 논문에서 구현한 MAC의 시각동기 성능은 최대 1982ns, 최소 -2991ns 시각오차 범위에서 rms 기준 125ns으로 측정되었다.

Key Words : TSN, IEEE 802.1AS, Time synchronization, MAC, PTP, LinuxPTP, SoC FPGA

ABSTRACT

In this paper, we describe the design and implementation of IEEE 802.1AS high-precision time synchronization capable MAC for TSN(Time-Sensitive Network), which is being actively studied as a next generation network technology for controlling and monitoring industrial IoT devices. The proposed MAC supporting IEEE 802.1AS Time Synchronization is composed of Ethernet MAC and Time-of-Day module based on ALTERA SoC FPGA and is implemented to control corresponding hardware block with the ARM processor embedded in SoC FPGA. In addition, we implemented IEEE 802.1AS time synchronization capable MAC device driver on Linux OS (ubuntu 16.04 LTS) to evaluate time synchronization accuracy performance using LinuxPTP, an open source program for PTP(Precision Time Protocol). The experiment results show that the time synchronization offset of the proposed MAC is rms 125ns at range from 1982ns to -2991ns.

1. 서론

1989년, WWW(World Wide Web)의 등장은 인터넷이 양적, 질적으로 급격히 고도화되는 계기가 되어, 이를 이용한 소비자 중심의 다양한 콘텐츠 및 서비스

가 크게 발전하였다. 이후 스마트폰의 등장을 필두로 기술의 영역 확장과 융합을 바탕으로 한 IoT(Internet of Things)관련 기술들이 쏟아져 나와, 개인소비 관련 분야 뿐만 아니라 전 산업분야로 확대되면서 Industrial-IoT(IIoT)로 발전되고 있다^[1]. 각종 산업

* 본 연구는 한국전자통신연구원 연구운영비지원사업의 일환으로 수행되었음. [18ZK1130, 저지연/실시간 네트워킹 엣지형 단말 플랫폼 핵심기술 개발]

♦ First and Corresponding Author : (ORCID:0000-0002-0436-6312)Electronics and Telecommunications Research Institute, gya@etri.re.kr, 정회원

* (ORCID:0000-0002-0436-6312)Electronics and Telecommunications Research Institute, {gykim, rskim, bhson, harkyoo}@etri.re.kr
논문번호 : 201806-0-112-SE, Received May 11, 2018; Revised July 6, 2018; Accepted July 6, 2018

분야에 광범위하게 적용되고 있는 IIoT를 위해서는 제품과 산업기계 등과 같은 사물과 사물 간 통신이 필요하며, 이를 위해 다수의 장치를 유연하게 연결하기 위한 확장성 및 장치 제어의 정밀성을 위한 저지연, 고정밀 네트워크 기술이 요구된다^[2]. 인터넷 기술과 다양한 산업분야가 경계 없이 융합됨에 따라, 네트워크로 전송되는 데이터에는 융합기술의 특성에 따른 높은 중요도를 갖는 정보와 일반 사용자용 서비스 정보가 혼재되는 양상을 보이고 있다^[1]. 네트워크에 다양한 정보가 혼재된 경우, 보다 중요한 정보를 특정 시간 내 전송하기 위한 TSN(Time-Sensitive Network, 실시간 네트워크) 기술이 최근 각광을 받고 있다^[3].

TSN은 장치 간 데이터 전달 시간 최소화를 통한 보다 빠른 모니터링 및 시각동기화에 의한 정밀 제어 등, 다수의 종단 장치로부터 정밀 시각단위로 데이터를 수집하고 제어를 할 수 있는 IEEE 국제 표준 기술로 IEEE 802.1 working group의 Time-Sensitive Networking task group에 의해 표준화가 진행되고 있다^[2,3]. TSN은 다양한 정보가 혼재된 이더넷 네트워크에서 중요도에 따른 패킷의 스케줄링을 비롯하여 트래픽 처리 및 관리에 관한 표준기술의 집합으로 고가용, 고신뢰 네트워크 환경을 제공한다. 이러한 고가용, 고신뢰 네트워크를 제공하기 위한 TSN은 IEEE 802.1AS기반의 고정밀 시각동기 기술을 기반으로 하고 있다.

본 논문에서는 IEEE 802.1AS 표준의 시각동기 프로토콜(PTP; Precision Time Protocol)에서 요구하는 Hardware timestamp기능을 포함한 SoC FPGA기반 MAC(Media Access Control) 장치와 이를 Linux OS 환경에서 구동하기 위한 디바이스 드라이버 소프트웨어를 설계 및 구현하였으며, PTP를 구현한 오픈소스 소프트웨어인 LinuxPTP를 통해 시각동기 정확도에 대한 성능을 검증하였다.

II. 관련 연구

2.1 Time Sensitive Network

실시간 네트워크는 다양한 산업 시스템의 저지연, 무손실 제어를 지원하기 위해 오디오/비디오 전문 기술인 AVB(Audio Video Bridging) 기술에서 확장된 산업 네트워크 기술이다^[4]. IEEE 802.1 Working group의 TSN task group에서는 시민감형 데이터의 즉각적인 전송을 위해 전송중인 비-시민감형 데이터의 전송을 중단하는 Frame preemption 기능이 포함된 IEEE 802.1Qbu, 시민감형 데이터의 전달 지연 방

지를 위한 큐 스케줄링 기능이 포함된 IEEE 802.1Qbv 등의 표준 기술을 제정하였다. 뿐만 아니라, 네트워크 전체 시민감형 데이터 전송 스케줄을 관리하는 IEEE 802.1Qcc 및 시민감 데이터의 주기적인 Queueing 및 Forwarding과 관련된 IEEE 802.1Qch 등 10개의 표준 작업이 활발히 진행 중이다. 표 1은 현재 표준화 작업이 이뤄지고 있는 TSN 관련 표준 기술을 나열한 것이다.

TSN은 표 1에 언급된 표준 기술들을 활용해 시민감 네트워크 데이터의 종단 간 전송 지연 시간을 보장한다^[5]. IEEE 802.1에 포함되는 다양한 TSN 표준 기술들은 실시간 통신 솔루션에 필요한 세 가지 기본 핵심 요소 범주로 그룹화 할 수 있다.

표 1. TSN 관련 표준 기술
Table 1. Standard technologies for TSN

Project designation	Summary
IEEE 802.1AS-rev	Time synchronization enhancements
IEEE 802.1CB	Frame replication and elimination
IEEE 802.1Qcc	SRP enhancements and performance improvements
IEEE 802.1Qch	Cyclic queueing and forwarding
IEEE 802.1Qci	Per stream filtering and policing
IEEE 802.1CM	Profile for fronthaul
IEEE 802.1Qcp	802.1Q YANG data model
IEEE 802.1CQ	Local address protocol
IEEE 802.1Qcr	Asynchronous traffic shaping
IEEE 802.1CS	LRP (Link-local Registration Protocol)

2.1.1 시각동기화

실시간 네트워크에 참여하는 모든 장치는 공통된 시각으로 동기화되어야 한다. 이론적으로는 GPS(Global Positioning System) 시각을 이용해 실시간 통신에 참여하는 모든 장치의 시각을 동기화하는 방법이 있다. 하지만 GPS 시각 기반의 시각동기화 기술은 구현 비용이 높고 GPS 신호가 수신되지 않는 음영지역에서의 동작이 불가능하다는 단점이 있다. 이를 극복하기 위해 이더넷 프레임에 이용해 시각동기화 정보를 공유하는 방식으로 장치 간 시각을 동기화하는 IEEE 1588 PTP가 사용된다.

2.1.2 스케줄 및 트래픽 처리

실시간 네트워크에 참여하는 모든 장치는 통신 패킷 처리 및 전달 시, 동일한 규칙을 준수해야 한다. IEEE 802.1Q 표준에서 정의된 IEEE 802.1Q VLAN(Virtual LAN) Tag 및 VLAN Priority를 이용

하여 단일 네트워크 내에서 트래픽 클래스와 우선순위가 다른 다양한 데이터의 공존이 가능하다. 기존의 IEEE 802.1Q 환경에서 시민감 데이터와 기존 데이터 간 구분은 가능하지만 우선순위가 높은 데이터의 전송시간에 대해 절대적으로 보장해주지는 못한다. 그 이유는 스위치 포트 중, 하나의 이더넷 프레임 전송을 시작한 경우 우선순위가 가장 높은 프레임조차도 전송이 완료될 때까지 스위치 버퍼 내에서 대기하는 버퍼링 현상 때문이다. 이와 같은 단점을 극복하기 위해 TSN은 IEEE 802.1Qbv 표준에서 시각 인식 스케줄러 기술의 적용을 통해 시민감 데이터 전송을 위한 대역폭 및 시간 슬롯을 사전에 미리 예약하는 기술을 제시하였으며, 이 기술을 통해 종단 간의 전송 지연 시간을 보장할 수 있다.

2.1.3 통신 경로 예약

실시간 네트워크에 참여하는 모든 장치는 동일한 규칙에 따라 경로를 선택하고 대역폭 및 시간 슬롯을 예약할 수 있어야 한다. 이를 위해 TSN에서는 IEEE 802.1Qcc 기반의 관리 기술을 적용한다. IEEE 802.1Qcc는 대규모 TSN의 대역폭 및 시간 슬롯 관리에 대한 인터페이스와 프로토콜을 정의하고 있다. IEEE 802.1Qcc는 소프트웨어 정의 네트워크(SDN, Software-Defined Network)의 개념을 도입하여 중앙 집권적인 자원(회선) 관리 방식을 사용한다.

2.2 IEEE 802.1AS

IEEE 802.1AS 표준은 IEEE 1588 PTP 표준을 IEEE 802.1Q 기반의 가상 브릿지 LAN(Local Area Network)에서 사용하기 위한 프로파일을 정의한다. IEEE 802.1AS 표준은 네트워크에 연결된 각 장치 간 시각을 동기화하는 프로토콜에 대해 기술한다³⁾. IEEE 802.1AS에서는 시각동기를 수행하기 위해 각 네트워크 장치의 기준 클럭 정보를 ANNOUNCE 메시지를 통해 공유하고 해당 정보를 바탕으로 BMCA(Best Master Clock Algorithm)를 통해 동기의 기준시각을 제공할 Master 장치와 Slave 장치를 결정한다. 장치 간 시각 동기 프로토콜의 각 패킷은 이벤트(Event) 메시지와 일반(General) 메시지로 구분되며 이벤트 메시지의 송·수신 시, 별도로 구성된 시각발생장치로부터 생성되는 Egress 및 Ingress timestamp 값을 바탕으로 Master와 Slave 간 시각을 동기화하고 기준 클럭에 의한 오차를 보정한다. 시각 오차는 각 계층 간 데이터 전달에 대한 지연시간과 기준 클럭의 지터(Jitter) 등, 시각 값 생성에 존재할 수

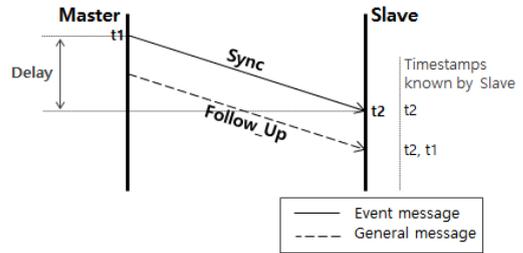


그림 1. IEEE 802.1AS 시각 동기
Fig. 1. Time synchronization of IEEE 802.1AS

있는 불확실한 요인(Uncertainty)이 반영된다^{3,6)}. IEEE 802.1AS는 IEEE 1588v2와 동일하게 Sync 메시지와 Follow_Up 메시지를 이용하여 Slave가 Master의 시각으로 동기를 맞춘다. 그림 1은 IEEE 802.1AS 프로토콜에서 시각 동기(Time synchronization) 절차를 표현한 그림이다. 그림 1과 같이 Master는 시각 동기를 위해 Sync 메시지를 전송하고 이어서 Follow_Up 메시지를 전송한다.

Master는 이벤트 메시지만 Sync 전송 시 Egress timestamp를 생성 및 저장하고 Slave는 Sync가 수신될 때 Ingress timestamp를 생성 및 저장한다. Master는 Sync 메시지를 전송한 직후, 일반 메시지만 Follow_Up에 Sync 메시지 전송 시 생성된 Egress timestamp 값을 담아 Slave로 전송한다. 이러한 절차를 통해 얻어진 Egress timestamp(t_1)와 Ingress timestamp(t_2)의 차(t_2-t_1)로부터 Slave 장치의 시각 값을 보정할 수 있다. IEEE 802.1AS에서 Master와 Slave 장치는 Peer-to-peer로 연결되며, 이에 따른 전송 지연(Propagation delay) 측정방법은 그림 2와 같은 절차를 통해 얻어진 Timestamp 값들을 바탕으로 구할 수 있다.

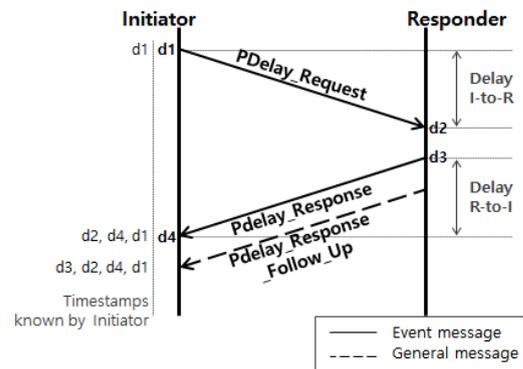


그림 2. IEEE 802.1AS 전송 지연
Fig. 2. Propagation delay measurement of IEEE 802.1AS

그림 2와 같이 전송지연을 알고자하는 장치 (Initiator)에서 요청(PDelay_Request; Peer Delay Request) 메시지를 전송하면 연결된 상대 장치 (Responder)가 응답(PDelay_Response; Peer Delay Response)하는 절차를 통해 전송지연에 필요한 Timestamp 값을 생성 및 수집한다. Responder의 응답은 이벤트 메시지인 PDelay_Response와 일반 메시지인 PDelay_Response_Follow_Up을 쌍으로 시각 동기 절차와 유사한 2-step 방식으로 진행된다. 다만, Sync 메시지와는 다르게 PDelay_Response 메시지의 패킷정보에 PDelay_Request 수신 시 생성된 Ingress timestamp(d2)가 포함되어야 한다. PDelay_Response 메시지가 전송될 때 생성되는 Egress timestamp(d3)는 해당 메시지 전송 직 후 전송 되는 PDelay_Response_Follow_Up 메시지에 포함되어 Initiator 측으로 전달한다. 그림 1과 그림 2의 절차로부터 얻어진 Timestamp 값은 식 (1), (2)와 같이 전송 지연(Delay_p)과 시각오차(Offset)를 얻으며, 이 값들로 Slave에서 시작동기 및 시작동기 오차 값을 확인할 수 있다.

$$Delay_p = \frac{(d4 - d3) + (d2 - d1)}{2} \quad (1)$$

$$Offset = t2 - t1 - Delay_p \quad (2)$$

III. IEEE 802.1AS 시각동기 지원 MAC 설계 및 구현

본 논문에서는 ALTERA사의 SoC FPGA를 기반으로 IEEE 802.1AS 시각동기를 위한 Hardware timestamp 기능을 지원하는 IEEE 802.1AS 시각동기 지원 MAC을 설계 및 구현하였다. ALTERA사의 SoC FPGA는 Quartus Prime 설계 툴을 통해 내장된 프로세서와 FPGA(Field-Programmable Gate Array) Fabric에 구현한 하드웨어 간 인터페이스를 보다 직관적으로 설계할 수 있는 개발환경을 제공하는 장점이 있다. SoC FPGA는 FPGA 칩에 Hard Macro로 내장된 ARM 프로세서 및 관련 주변장치블록(프로세서 시스템)이 포함된 형태를 의미하며, ARM 프로세서와 FPGA는 칩 내부에 AXI(Advanced eXtensible Interface) 버스에 의해 연결성을 확보하였다. 이에 따라, FPGA에 구성된 하드웨어 기능블록을 매우 인접한 위치의 ARM 프로세서가 보다 효율적으로 접근, 제어할 수 있다.

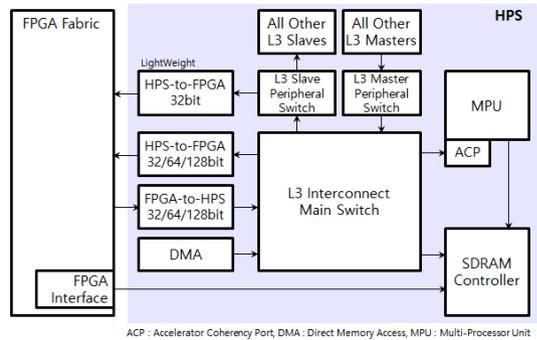


그림 3. ALTERA SoC FPGA의 HPS to FPGA 인터페이스
Fig. 3. HPS to FPGA interface of ALTERA SoC FPGA

본 논문에서 활용한 ALTERA사의 SoC FPGA는 그림 3과 같이 최대 925MHz로 동작할 수 있는 Dual-core ARM Cortex-A9 기반의 프로세서 시스템 (HPS; Hard Processor System)이 Hard Macro로 구성 되어있다⁷⁾. 그림 3과 같이 ALTERA사 SoC FPGA는 FPGA 영역에서 구현된 하드웨어 블록을 HPS-to-FPGA AXI, FPGA-to-HPS AXI 및 HPS-to-FPGA Lightweight AXI 버스 인터페이스를 통해 제어할 수 있다.

이러한 구조를 활용하여 IEEE 802.1AS 시각동기 지원 MAC은 FPGA 영역에서 그림 4와 같이 설계하였으며, 이를 HPS로 구성된 ARM 프로세서 기반의 LinuxOS에서 제어할 수 있도록 IEEE 802.1AS 시각동기 지원 MAC 드라이버 소프트웨어를 구현하였다.

그림 4의 Triple-Speed Ethernet MAC(이하 TSE MAC), DMA(Direct Memory Access), Time-of-Day 블록은 ALTERA사의 합성소프트웨어인 Quartus Prime에서 제공되는 IP 라이브러리를 활용 하였다⁸⁾. TSE MAC의 Hardware timestamping 기능을

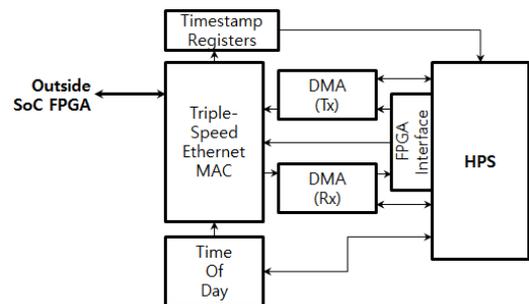


그림 4. IEEE 802.1AS 시각동기를 지원하는 Ethernet MAC 구조
Fig. 4. Ethernet MAC Structure of supporting IEEE 802.1AS Time Synchronization

활성화하고 Time-of-Day 블록을 추가하여, Time-of-Day 블록으로 부터 얻을 수 있는 기준 시각 값으로부터 Timestamp가 생성될 수 있도록 설계하였다. 이와 같이 구성하면 TSE MAC은 Time-of-Day에서 발생하는 기준 시각 값을 기초로 패킷의 전송 또는 수신 시, Egress 및 Ingress timestamp를 생성한다. Time-of-Day블록의 기준 클럭은 해당블록에 시각 값을 생성하기 위한 전용 클럭과 이 전용 클럭의 Period, Jitter 등 각종 정보로부터 얻어지며, 본 논문에서는 50MHz의 공용 발진기(Oscillator)로부터 PLL(Phase Locked Loop)에 의해 분주된 125MHz 클럭을 기반으로 기준 시각 값이 생성되도록 설계하였다. 이러한 구조에서 Egress 및 Ingress timestamp는 MII(Media Independent Interface)기준으로 패킷이 전송 또는 수신 시작시점에서 생성되며 각각 Physical layer 장치 (PHY)로부터 제공되는 클럭에 동기되어 출력된다. 본 논문에서 설계한 IEEE 802.1AS 시각동기 지원 MAC과 PHY 간 인터페이스는 RGMII (Reduced Gigabit-MII)로 구성되며 해당 구성의 Timestamp 생성 시점은 그림 5와 같이 시뮬레이션으로부터 확인할 수 있다. 그림 5는 ModelSim 10.1c 시뮬레이터를 통해 얻어진 RTL(Register Transfer Level) 시뮬레이션 결과이다.

TSE MAC으로부터 생성되는 Timestamp는 IEEE 802.1AS 표준에서 정의한 96비트의 Extended-Timestamp 규격으로 제공되며 해당 값으로부터 초 단위영역인 상위 16비트, 나노초의 소수점 영역인 하위 16비트를 제거하여 상위계층에서 사용되는 초 단위 32비트, 나노-초 단위 32비트로 인터페이스를 구성하여, AXI 버스를 통해 프로세서에서 값을 읽을 수 있도록 설계하였다.

IEEE 802.1AS 시각동기 지원 MAC은 ALTERA 사의 SoC FPGA에 해당하는 Cyclone V SE(5CSEMA4) 디바이스에 합성소프트웨어인 Quartus Prime 16.0 standard edition을 이용하여 구현하였다. 그림 4와 같은 구조로 설계한 IEEE 802.1AS 시각동기 지원 MAC의 합성결과는 표 2와 같다.

표 2. IEEE 802.1AS 시각동기 지원 MAC 합성 결과
Table 2. Synthesis result of IEEE 802.1AS Time-Synchronization supporting MAC

	Used	Available	Ratio(%)
Logic utilization(in ALMs)	10,613	15,880	67
Total registers	17,846	-	-
Total pins	208	314	68
Total block memory bits	348,287	2,764,800	13
Total PLLs	2	5	40
Total DLLs	1	4	25

Target Device - 5CSEMA4U23C6
Quartus Prime SE(16.0.0 build 211)

표 2와 같이 IEEE 802.1AS 시각동기 지원 MAC은 SoC FPGA(5CSEMA4)에서 전체 FPGA 허용자원 중, 일반 회로부분 67%, 메모리 13% 수준으로 구현 되는 것을 확인할 수 있다. 이렇게 구현된 IEEE 802.1AS 시각동기 지원 MAC은 HPS의 주변장치로 활용될 수 있으며, Linux OS(ubuntu 16.04 LTS) 기반에서 구현한 IEEE 802.1AS 시각동기 지원 MAC 디바이스 드라이버로 구동할 수 있도록 구현하였다.

본 논문에서 구현한 IEEE 802.1AS 시각동기 지원 MAC을 구동하기 위한 디바이스 드라이버는 TSE MAC의 패킷 전송 및 수신 등의 기초 기능을 비롯한 IEEE 802.1AS 표준에서 정의한 PTP의 요구기능을 지원한다. PTP를 위한 요구기능은 Time-of-Day 블록의 현재 시각 값 접근을 비롯하여 Time-of-Day 시각 값 수정 및 보정 기능, Egress 및 Ingress timestamp 획득 기능이다. 본 논문에서 구현한 IEEE 802.1AS 시각동기 지원 MAC의 시각동기 정확도 성능은 LinuxPTP 소프트웨어를 통해 측정하였다. LinuxPTP는 오픈소스 프로젝트로부터 IEEE 표준에 따른 PTP를 구현한 linux용 범용소프트웨어이다. 이 소프트웨어는 linux 커널 정보를 기반으로 범용의 PTP환경을 제공하여 개발한 시각동기 장치와 디바이스 드라이버를 통해 장치의 시각동기 성능을 확인할 수 있다^[9]. 해당 소프트웨어를 통한 Master-Slave 결정 후, Slave장치의

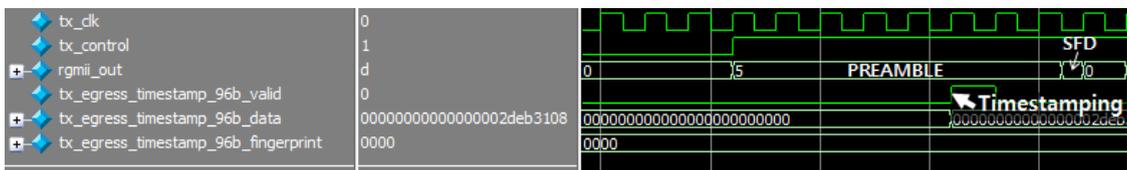


그림 5. ALTERA SoC FPGA 내, TSE MAC의 Timestamp 생성 시점
Fig. 5. Timestamp generation point of Triple-Speed Ethernet MAC on ALTERA SoC FPGA

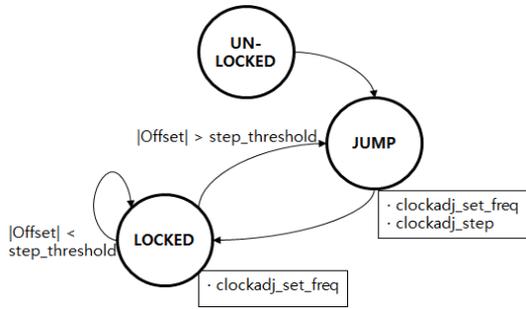


그림 6. LinuxPTP의 시각동기 FSM
Fig. 6. Time synchronization FSM of LinuxPTP

주요 시각동기 절차는 그림 6과 같다.

그림 6과 같이 LinuxPTP는 Master-Slave의 결정 후, JUMP 상태로 변경하며, 해당 상태에서는 계산된 주파수 보정값(Fadj:Frequency adjustment value)과 시각오차(Offset)을 기능함수인 clockadj_set_freq와 clockadj_step에서 각각 보정 및 수정하고 LOCKED 상태로 천이한다. LOCKED 상태에서는 주파수 보정 값에만 의존하여 Master와 Slave 간 시각오차를 보정한다. LinuxPTP에서 시각동기 오차보정을 위한 값은 식 (3), (4)와 같은 계산에 의해 얻어진다.

$$Offset = IngressTimestamp(@Sync) - EgressTimestamp(@Sync) \quad (3)$$

$$Fadj = Offset + drift \quad (4)$$

식 (3)과 같이 시각오차는 PTP 시각동기 절차 (Time synchronization)에 의해 Master에서 Sync 메시지를 보낼 때 생성되는 EgressTimestamp(@Sync)와 Slave에서 Sync 메시지를 받을 때 생성되는 IngressTimestamp(@Sync) 값의 차로 얻는다. 주파수 보정 값은 현재 시각오차와 시각오차 변화추이(drift)의 합으로 구할 수 있으며, LinuxPTP는 시각 값 보정 동작에 의한 시각오차 급증현상을 줄이기 위해 k_i , k_p 비율을 연산분리와 가중치(weight)를 적용하여 식 (5)와 같이 계산하고 시각오차에 대한 변화추이는 식 (6)과 같은 계산을 통해 얻어진다⁹⁾.

$$Fadj_{LinuxPTP} = (k_p \cdot Offset \cdot weight) + (k_i + Offset + weight) + drift_{(n)} \quad (5)$$

$$drift_{(n)} = (10^9 - drift_{(n-1)}) \cdot \frac{Offset_variation}{Sync_interval_Local}, \quad drift_{(0)} = 0.0 \quad (6)$$

식 (6)에 표기된 Sync_interval_{Local}은 Slave 장치의 IngressTimestamp(@Sync) 값에 의해 얻어지고, Offset_variation은 현재의 시각오차와 이전 시각동기의 시각오차 변화량을 의미한다. 이와 같이 LinuxPTP에서 얻어진 주파수 보정값과 시각오차 값은 디바이스 드라이버에 구현된 동작 모델에 따라 Time-of-Day 블록의 시각 값에 1 클럭 또는 수 클럭에 걸쳐 반영 되도록 구현하였다.

IV. IEEE 802.1AS 시각동기 지원 MAC 기능 시험

본 논문에서 설계 및 구현한 IEEE 802.1AS 시각동기 지원 MAC의 시각동기 정확도에 대한 성능을 검증하기 위해 Master 장치는 하드웨어 Timestamp 기능을 지원하는 네트워크 장치인 Intel Ethernet Server Adapter I350가 장착된 PC를 사용하였고, Slave 장치로 SoC FPGA기반의 IEEE 802.1AS 시각동기 지원 MAC를 사용하였다. 그림 7은 IEEE 802.1AS 시각동기 지원 MAC의 시각동기 성능을 측정하기 위한 장치의 구성을 도식화한 것이다.

ALTERA사 SoC FPGA의 FPGA 영역에 구현된 MAC의 구동을 위한 Physical layer는 그림 7과 같이 TEXAS INSTRUMENTS사의 PHY 시험보드를 FPGA의 GPIO(General Purpose Input/Output)에 Wire connection으로 연결하여 구성하였으며, Master Node가 되는 PC의 NIC(Network Interface Controller)와 PHY 시험보드는 Category.5 규격의 UTP(Unshielded Twist Pair) 케이블을 이용하여 연결하였다. IEEE 802.1AS 시각동기 지원 MAC 장치와 Intel I350 장치는 100Mbps, Full-duplex로 연결(Link)되도록 설정하였다. 이러한 환경에서 LinuxPTP를 이용하여 시각동기 성능을 측정하였다. 이와 같은 시험구성에서 소프트웨어기반의 PTP 시각동기 성능은 그림 8과 같이 측정되었다. 소프트웨어 방식의 시각

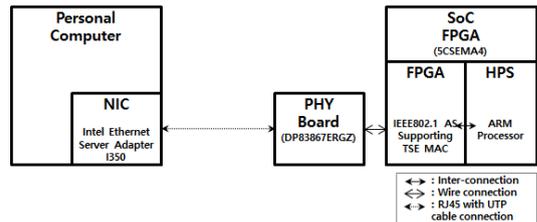


그림 7. IEEE 802.1AS 시각동기 지원 MAC 기능 시험 구성
Fig. 7. Test environment of IEEE 802.1AS Time-synchronization supporting MAC

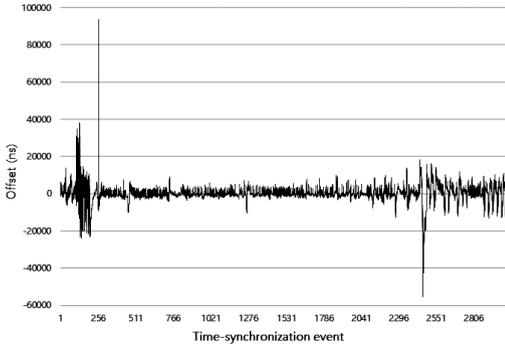


그림 8. 소프트웨어기반 PTP 시각 동기 성능 시험 결과
Fig. 8. Software timestamp based PTP Time-Synchronization test result

동기 과정에서는 Timestamp를 IP 패킷으로 주고 받는다. FPGA Fabric에 구현된 하드웨어 시각동기 지원 블록을 사용하지 않은 소프트웨어 시각동기는 그림 8과 같이 최대 93679ns, 최소 -55274ns 범위 내에서 rms(root mean square) 기준 5156ns 수준의 성능으로 확인되었다.

본 논문에서 구현한 IEEE 802.1AS 시각동기 지원 MAC의 시각동기 성능은 그림 9, 10과 같이 측정되었다. IEEE 802.1AS 시각동기 지원 MAC은 LinuxPTP가 실행될 때, Time-of-Day 블록의 기준 시각 값을 0으로 초기화하도록 구현되어 있다. 따라서, 최초 LinuxPTP가 동작을 시작하여 UNLOCKED 상태에서 JUMP 상태로 천이하면 Master와 Slave 간 매우 큰 시각오차가 발생된다. 그림 9, 10의 그래프는 시각동기 이벤트가 매 1초 마다 수행되는 환경에서 Master와 Slave 장치 간 시각동기 오차를 나타낸 것으로, 가로축은 시각동기 주기 수(Time-synchronization event) 나타내며 세로축은 시각동기 오차(Offset)를 나타낸다. 그림 9는 LinuxPTP의 UNLOCKED, JUMP 상태를 지나 LOCKED 상태에 진입한 직후, 초기 18번째 시각동기까지에 대한 시각동기 오차를 나타낸다. 그림 9에서 보이는 수십 us의 오차는 최초 LinuxPTP가 실행될 때, Slave의 Time-of-Day 기준 시각 값이 0으로 초기화되도록 구현되어, Master와 Slave의 시각 오차가 매우 크기 때문에 발생하는 것으로 그림 6에서 보는 바와 같이 JUMP에서 LOCKED 상태로 천이 시 오차 보정에 사용되는 clockadj_step 함수의 보정 정확도 한계에 의해 기인한 것이다. 이후, 그림 9에 나타난 Offset의 변화후이는 JUMP 상태에서 시각 값 보정 이후에 생긴 97,905ns값의 offset을 LOCKED 상태에서 0ns에 수렴하며 보정하는 양상을

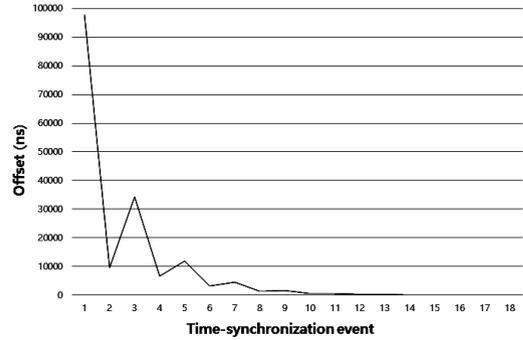


그림 9. LinuxPTP를 통한 시각동기 성능 시험 결과 (1~18 시각동기 구간)
Fig. 9. Result of PTP Time-Synchronization test on LinuxPTP (1~18 Time-synchronization event)

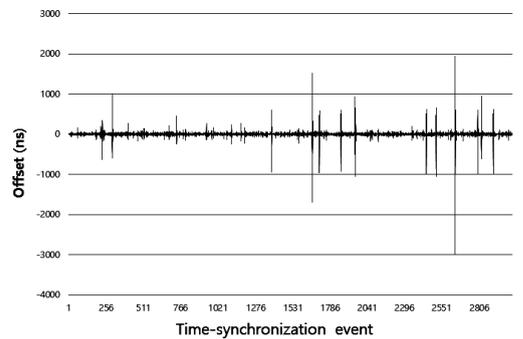


그림 10. LinuxPTP를 통한 시각동기 성능 시험 결과 (19 ~ 3052 시각동기 구간)
Fig. 10. Result of PTP Time-Synchronization test on LinuxPTP (19~3052 Time-synchronization event)

보인다. 그림 10은 그림 9의 시각동기에 이어서 총 3052회 동안의 시각동기 오차수준을 나타낸다. 그림 10에서 보는 바와 같이 Master와 Slave 간 시각동기 오차는 최대 1928ns, 최소 -2981ns 범위 내에서 rms 기준 125ns 수준을 보였으며, 큰 Offset 값에서부터 시각보정이 수행되는 그림 9의 초기 구간을 포함한 전체 구간에 대한 시각동기는 rms기준 1909ns 오차 수준의 시각동기 정확도 성능을 보였다.

그림 11은 예측된 시각오차에 대하여 -2000ns 부터 2000ns 범위 내에서 전체 3052회 중 3044회에 대한 시각오차의 분포를 나타낸 것이다. 이 분포 그래프에서 제외된 시각오차는 그림 9에서 보인 초기 보정 구간과 그림 10에서 보인 2660번째 시각오차 -2981ns 값이다. 그림 11에 의해 확인된 시각오차 분포는 -50ns부터 50ns 범위 내에 전체 시각오차의 83.22%가 예측되는 것으로 확인되었다. 이와 같은 시각동기 성능은 다양한 산업분야에서 실시간 네트워크의

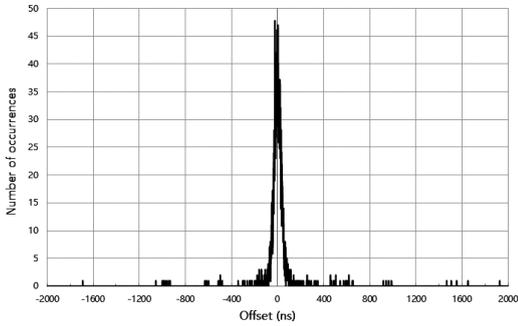


그림 11. PTP 시각동기 시험의 시각오차 발생 분포
Fig. 11. Offset occurrences distribution of PTP Time-Synchronization test

요소기술로 적용할 수 있을 것으로 보인다. 표 3은 5가지 산업분야에서 요구되는 시각동기 성능을 나타낸다^[10]. 본 논문에서 구현한 IEEE 802.1AS 시각동기 지원MAC의 시각동기 정밀도는 표 3에서 보는바와 같이 일반적인 산업 응용분야에 적용할 수 있는 성능을 보였다.

그림 12는 본 논문에서 구현한 IEEE 802.1AS 시각동기 지원 MAC의 하드웨어 시각동기 기능 적용 유/무에 따른 시각동기 성능을 시각오차 범위별로 구분하여 그래프로 나타낸 것이다. 그림 12에서 보이는 것과 같이 Timestamp 방식에 따른 시각동기 성능 비교에서 소프트웨어 시각동기는 $1000ns \leq |Offset|$

표 3. 산업 응용분야의 일반적인 시각동기 요구사항^[10]
Table 3. Typical requirements of Industrial applications^[10]

Industrial Application	Synch. Accuracy(μs)
Condition monitoring	1
Process automation	1000
Machine tool	0.5
Packaging machines	5
Printing machines	0.25

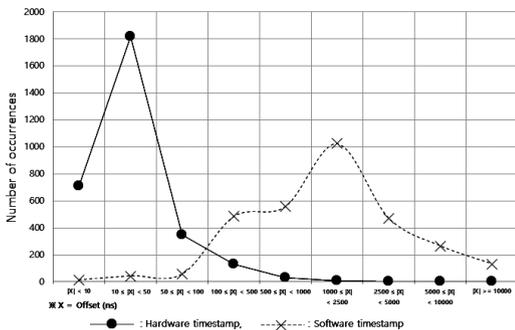


그림 12. PTP 시각동기 성능 비교
Fig. 12. Comparison of PTP Time-Synchronization

$<2500ns$ 구간, 하드웨어 시각동기는 $10ns \leq |Offset| < 50ns$ 구간에 밀집되는 특성을 보였다.

그림 9에서 보이는 큰 값의 Offset은 LinuxPTP의 JUMP 상태에 대한 디바이스 드라이버의 동작에서 clockadj_step과 clockadj_set_freq 기능에서 요청되는 디바이스 드라이버와 Time-of-Day 하드웨어 블록 간 접근에 관련된 인터페이스(AXI)를 공통으로 사용하는 다른 인접장치 제어에 따른 동작지연이 반영된 것으로 보인다. 또한, 그림 10에서 보이는 100ns 이상의 Offset은 clockadj_set_freq에 의한 보정값과 Time-of-Day 하드웨어 블록에서 기준시각 값 생성에 기준이 되는 발진장치(Reference clock for Time-of-Day)의 이상동작에 의한 것으로 추정된다^[6]. 이와 같은 오류성 동작은 Time-of-Day 하드웨어 블록에 독립된 정밀 발진장치 적용 및 Offset 수정 및 보정에 대한 디바이스 드라이버에 개선된 알고리즘 개발 및 적용으로 보완할 수 있을 것이다.

V. 결 론

IoT 기술이 각종 산업분야에 접목되면서 다양한 산업용 장치 간 통신기술이 요구되고 있다. 이러한 IIoT 환경에서의 산업장치 간 통신을 위해서는 다수의 장치를 유연하게 연결하기 위한 확장성 및 장치 제어의 정밀성을 위한 저지연, 고정밀 네트워크 기술이 요구되며, 따라서 높은 정밀도를 갖는 장치 간 시각동기화 기술이 활발히 연구되고 있다. 본 논문에서는 정밀 시각동기를 위한 IEEE 802.1AS 시각동기 지원 MAC을 ALTERA사의 SoC FPGA기반으로 설계 및 구현하고 시각동기 정확도에 대한 성능을 측정하였다. IEEE 802.1AS 시각동기 지원 MAC은 SoC FPGA HPS의 주변장치로 구현하였으며, Linux OS에서 구동할 수 있도록 디바이스 드라이버 소프트웨어를 개발하였다. 또한, ARM 프로세서 기반으로 구동되는 ubuntu 16.04 LTS 버전의 Linux OS에서 IEEE 802.1AS 시각동기 지원 MAC이 올바르게 동작되는 것을 확인하였다. IEEE 802.1AS 시각동기 지원 MAC의 시각동기 성능 측정을 위해 오픈소스 소프트웨어인 LinuxPTP를 이용하였으며, 최대 1929 ns, 최소 -2991 ns 오차 범위로 rms기준 125ns 오차 수준의 시각동기 성능을 보이는 것을 확인하였다. 본 논문에서 구현한 IEEE 802.1AS 시각동기 지원 MAC은 IIoT분야에서 요구되는 높은 정밀도 및 신뢰성을 요구하는 실시간 네트워크의 게이트웨이 및 엣지 컴퓨팅 플랫폼 등 다양한 서비스를 수용할 수 있는 종단장치로

활용될 수 있을 것이다.

References

- [1] H. Kim and S. K. Hwang, "Past, present, and future of IoT," *Electron. Telecommun. Trends*, vol. 33, no. 2, pp. 1-9, 2018.
- [2] Y. S. Choi, "Industrial internet of things(IIoT) market outlook and technology trends," *The Mag. IEEE.*, vol. 44, no. 5, pp. 43-49, May 2017.
- [3] IEEE 802.1AS, "*IEEE Standard for Local and Metropolitan Area networks - Timing and Synchronization for Time-Sensitive Applications in Bridged Local Area Network*," IEEE Std. 802.1AS-REV Draft 7.0, Mar. 2018.
- [4] Y. C. Ryoo, J. T. Song, J. D. Ryoo, T. S. Cheung, J. S. Ko, J. W. Youn, and S. M. Kim, "Ultra-low latency infra technology," *Electron. Telecommun. Trends*, vol. 32, no. 1, pp. 13-24, Feb. 2017.
- [5] S. Kehrer, O. Kleineberg, and D. Heffernan, "A comparison of fault-tolerance concepts for IEEE 802.1 time sensitive networks(TSN)," in *Proc. 2014 IEEE ETFA*, pp. 1-8, Barcelona, Spain, Sept. 2014.
- [6] P. Chopp and A. Hamoui, "Analysis of clock-jitter effects in continuous-time $\Delta\Sigma$ modulators using discrete-time models," *IEEE Trans. Cir. Syst. Regular Papers*, vol. 56, no. 6, pp. 1134-1145, Jun. 2009.
- [7] Altera, *Overview of the Design Guidelines for Cyclone V SoC FPGAs and Arria V SoC FPGAs(2017)*, Retrieved Jul. 6, 2018, from <http://www.altera.com>
- [8] G. H. Yoon, B. H. Son, H. Yoo, and G. Y. Kim, "Design and implementation of IEEE 802.1AS MAC for IIoT," in *Proc. Symp. KICS*, pp. 1216-1217, Gang-Won, Korea, Jan. 2018.
- [9] R. Cochran, *The Linux PTP Project(2018)*, Retrieved Jul. 6, 2018, from <http://linuxptp.sourceforge.net>
- [10] S. Nsaibi and A. Mishra, "Experimental

evaluation of TSN timing performance," *The J. Ind. Netw. and IoT*, no. 106, pp. 33-36, May 2018.

윤 기 하 (Gi-Ha Yoon)



2012년 2월 : 목포대학교 정보통신공학과 졸업

2017년 2월 : 전남대학교 전자컴퓨터공학과 석사

2017년 2월~현재 : 한국전자통신연구원 연구원

<관심분야> 디지털시스템설계,

저지연/실시간 네트워크 기술, FPGA, ASIC

김 근 용 (Geun-Yong Kim)



2004년 2월 : 광운대학교 전자공학부 졸업

2006년 2월 : 광주과학기술원 정보통신공학과 석사

2017년 2월 : 광주과학기술원 전기전자컴퓨터공학과 박사

2006년 2월~현재 : 한국전자통신연구원 호남권연구센터 선임연구원

<관심분야> 저지연/실시간 네트워크 기술, IoT 소프트웨어 플랫폼, 엣지 컴퓨팅 기술

김 량 수 (Ryang-Soo Kim)



2010년 2월 : 충남대학교 정보통신공학과 졸업

2012년 2월 : 광주과학기술원 정보통신공학과 석사

2017년 8월 : 광주과학기술원 전기전자컴퓨터공학과 박사

2017년 9월~현재 : 한국전자통신연구원 연구원

<관심분야> 저지연/실시간 네트워크 기술, HVDC 변환 제어플랫폼 기술, 무선 통신 기술

손 병 희 (Byung-Hee Son)



2009년 2월 : 중앙대학교 전자
전기공학부 졸업
2011년 2월 : 중앙대학교 전자
전기공학부 석사
2015년 8월 : 중앙대학교 전자
전기공학과 박사
2016년 1월~현재 : 한국전자통신

연구원 호남권연구센터 선임연구원

<관심분야> 저지연/실시간 네트워크, IIoT, 엣지 컴
퓨팅 기술

유 학 (Hark Yoo)



1998년 2월 : 연세대학교 전기
공학과 공학사
2000년 2월 : 한국과학기술원 공
학석사
2005년 8월 : 한국과학기술원 공
학박사
2005년 2월~현재 : 한국전자통신

연구원 호남권연구센터 책임연구원, 산업융합네트워
크연구실 실장

<관심분야> 저지연/실시간 네트워크 기술, HVDC 변
환 제어플랫폼 기술, 엣지 컴퓨팅 기술