

논문 2019-1-6 <http://dx.doi.org/10.29056/jsav.2019.06.06>

사물인터넷 디바이스의 집적회로 목적물과 소스코드의 유사성 분석 및 동일성

김도현*, 이규대**†

Similarity Evaluation and Analysis of Source Code Materials for SOC System in IoT Devices

Do-Hyeun Kim*, Kyu-Tae Lee**†

요 약

사물인터넷 디바이스의 소형화, 저전력화 요구는 프로그램을 단일 칩으로 구현하는 SOC 기술로 구현되고 있다. 불법 복제에 의한 저작권 분쟁은 반도체 칩에서도 증가하고 있으며, 디자인하우스의 칩 구현에서의 분쟁과 소스코드의 도용에 의한 칩 구현에 발생하고 있다. 그러나 최종 칩 구현은 디자인하우스에서 제작되기 때문에 저작권의 보호범위에서 어려움이 있다. 본 연구에서는 사물인터넷 디바이스의 집적회로에서 HDL 언어로 작성된 소스코드의 분쟁에서, 유사성을 판단하기 위한 분석방법과 유사성 판단의 기준을 설정하는 항목에 대해 다루었다. 특히 동일한 시방서를 기준으로 제작된 칩의 경우 동일한 구성과 코드 형태를 포함해야 하는 제작특성에서 유사성의 판단영역을 구분하는 내용에 대해서도 다룬다.

Abstract

The needs for small size and low power consumption of information devices is being implemented with SOC technology that implements the program on a single chip in Internet of Thing. Copyright disputes due to piracy are increasing in semiconductor chips as well, arising from disputes in the chip implementation of the design house and chip implementation by the illegal use of the source code. However, since the final chip implementation is made in the design house, it is difficult to protect the copyright. In this paper, we deal with the analysis method for extracting similarity and the criteria for setting similarity judgment in the dispute of source code written in HDL language. Especially, the chip which is manufactured based on the same specification will be divided into the same configuration and the code type.

한글키워드 : SOC, 칩설계, 하드웨어언어, 불법복제, 유사성
keywords : SOC, chip design, HDL, illegal copy, similarity

* 국립 제주대학교 컴퓨터공학과
** 국립공주대학교 정보통신공학부
† 교신저자: 이규대(email: ktleee@kongju.ac.kr)
접수일자: 2019.06.01. 심사완료: 2019.06.15.
게재확정: 2019.06.20.

1. 서 론

사물인터넷에서 디바이스 제작이 소형화, 저전력 요구에 의해, 생산 및 개발자는 시스템의

기능이 확정되고, 대량생산의 가능성이 있을 경우, 일반 프로세서를 사용하는 방식보다 효율적인 SoC(system On chip)을 사용한 시스템을 선호한다[1]. 이 방법은 프로세서 방식과 달리 한번 제작되면, 기능의 수정이 불가능하지만, 소형, 저전력 시스템 구성이 가능한 장점이 있다.

일반적인 시스템 개발과정은 블록도 설계, 하드웨어 설계, 응용프로그램 설계 및 코딩 등으로 진행되지만, SoC 방법은 한번 제작하면 수정이 불가능하고, 개인적인 작업이 불가능한 제작공정이 있다[2]. 따라서 개발자는 설계검증이 완료된 중간 코드 형태의 HDL 언어로 작성된 프로그램 소스코드 까지만 작업을 한다. 집적회로의 개발은 칩을 제작하는 과정에서의 중간 코드를 개발하는 것과 같다. 개념적인 설계로 제작하고자 하는 칩의 기능을 정하고, c 언어와 같은 고급언어로 시뮬레이션이 가능한 정도로 구현하고, 출력이나, 처리시간 등을 점검한다. 여기까지 확인이 되면, HDL 언어를 사용하여 물리적인 로직으로 구현이 가능하도록 코딩작업을 수행하고, 하드웨어의 처리 비트에 적합한 정수형 수정 작업을 수행한다. 여기까지 정상적인 처리시간과 출력이 확인되면, 칩을 제작하는데, 이 과정은 가용한 로직게이트의 상태에 따라 정해지는 것으로 주로 전문설계업체에서 담당한다. 이것은 PCB 의 패턴을 만드는 것과 같아서, 칩 생산업체에 넘어서면, 칩의 핀 개수, 핀의 기능 및 배치가 상이하거나, 유사할 수 있다.

SoC 관련 칩에 대한 분쟁이 발생하면, 칩의 형태만 확인가능하기 때문에, 내부의 프로그램 상태보다 칩의 크기, 핀의 개수, 각 핀의 배치 및 기능만으로 유사성을 의심할 가능성이 있다[3]. 소스코드가 확인되지 않은 상태에서 외형적인 모습은 칩의 제작사나 칩 제작 전문가가 같을 경우에는 내부 소스코드가 상이한 경우도, 유사할 가능성이 있다.

따라서 이러한 분쟁은 칩의 형태보다는 분쟁 당사자가 개발에 적용한 검증단계에서의 소스코드가 확보되었을 때 정확한 분석 및 판단이 가능한 특징을 갖는다.

본 연구에서는 SoC 칩을 사용하는 정보기기의 시스템에서 분쟁이 발생한 상황에 대해 양측의 자료를 분석하고, 유사성을 도출하는 방법의 객관적인 절차를 다룬다. 또한 양측의 개발 칩이 동일한 사양서(specification)에 의해 제작된 경우에 대한 사례를 통해, 분석방법의 객관성을 검증한다.

2. 감정목적물의 특징

2.1 집적회로 개발 구조

집적회로는 FPGA (Field Programmable gate Array), ASIC(Application specific integrated circuit), SOC(System On chip) 등으로 구분할 수 있으며, FPGA 는 프로그램 수정이 가능한 구조로 되어있어, 개발단계 또는 소량의 시스템 구현에 활용되고 있다, FPGA로 구현된 시스템이 대량생산 요구가 있으면, ASIC 으로 변환하게 되는데, 비용도 많이 들고, 개발시간도 길지만 한번 제작하면, 저전력, 소형화, 저가격의 공급이 가능한 장점이 있다. FPGA 나 ASIC 에 프로세서를 함께 포함하는 칩이 제작되는 경우 SOC 로 부르며, 다양한 인터페이스 회로를 포함하여 하나의 칩으로 제작하는 기술이다.

집적회로의 설계는 블록 다이어그램으로 구성되는 기능 블록의 연결로 완성되는데, 각 기능에 대한 프로그램을 구성하는 과정에서 사용되는 프로그램언어는 HDL(Hardware Description Language) 이고, VHDL 또는 verilog 등이 사용된다. HDL 프로그램이 완성되면, 합성과정을

하는데, synthesis 라고하며, 이 과정을 통해 gate level netlist가 완성된다[2][4]. 이 자료로 칩을 제작하는 단계로 진행된다. 이때 플립플롭, combinational logic 등의 레지스터레벨의 작업이 함께 합성되면 RTL(register transfer level) 절차를 포함하여 합성된다. 이후는 검증의 단계로, 일단 칩이 만들어지면 수정이 불가능하고, 수정 후 재 작업하면 제작비용이 추가되기 때문에 여러 번의 정확한 검증을 수행한다. 검증이 완료되면, 칩을 제작하는 설비를 보유한 디자인하우스에서 후속작업으로 완성된다[2].

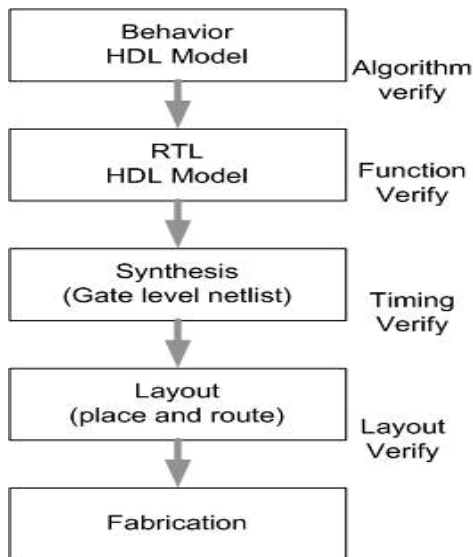


그림 1. 칩 설계 흐름도
Fig. 1. chip design flow

2.2 목적물 분석

감정목적물은 가정용 전기선(전력선)을 사용하여 통신을 가능하게 하는 PLC(Power Line Communication) 통신규약을 칩(chip)으로 구현한 프로그램이다[5][6]. 전기선 상에서 통신을 하는 방법은 그림 2와 같이 PLC coupler 와 전력선 통신 모델을 통해 데이터 통신이 가능하게 하는

방법이다. 전력선은 60Hz의 교류전원이고, 잡음이 심한 통신선으로 이곳에 데이터를 패킷으로 전송하는 기술은 잡음을 최소화하면서, 고속의 정보 데이터를 올려야하는 기술이 요구된다. 모든 가정에 전력선이 이미 설치되어있기 때문에 이 기술의 사용은 추가로 통신선을 배선하지 않아도, 홈네트워크 등의 다양한 서비스가 가능한 통신망으로 활용성이 우수한 기술이다.

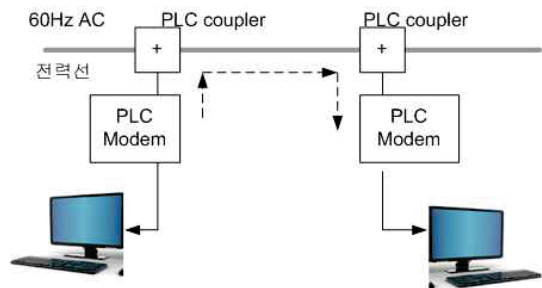


그림 2. 전력선 통신의 구성도
Fig. 2. Power Line communication flow

기술적인 통신방법은 그림 3과 같이 인터넷 통신 규약인 TCP/IP 의 7 레이어 구조에서 PHY와 MAC의 기능을 사용하여, PHY 층에서 전력선에 연결되도록 구현하는 방식이다.

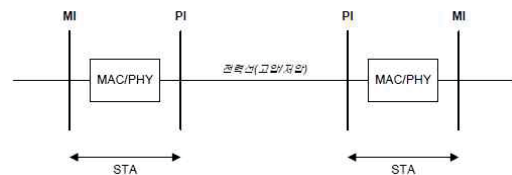


그림 3. PLC 참조 모델
Fig. 3. Reference model of PLC

PI(PHY Interface)는 STA(Station)와 함께 전력선에 연결되는 물리적(PHY) 인터페이스라고 하며, MI(MAC Interface)는 상위의 링크계층과 STA의 MAC 계층 사이를 정의하는 논리적 인

터페이스로 구분한다. 이 규격에 정의된 MAC와 PHY를 포함하는 장치를 STA(station)라 지칭한다. 그림 4와 같이 송신단과 수신단의 통신 프로토콜은 LINK-MAC-PHY 경로로 송신되고, PHY-MAC-LINK 순서로 수신된다.

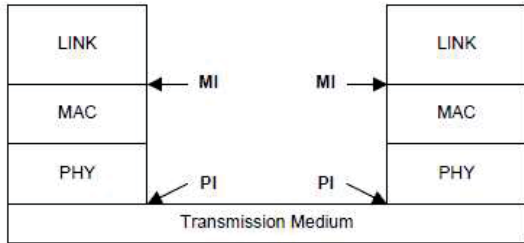


그림 4. PLC 프로토콜 참조모델
Fig. 4. PLC protocol model

송신단에서는 변조방식으로 DMT 심벌을 전송하는데, 송신과정은 그림 5와 같다. 순서적으로 송신측은 전송 데이터를 암호화(encryption), CRC, scramble, FEC interleaver, psk 변조, IFFT, DAC 과정으로 변환하여 전력선으로 출력된다.

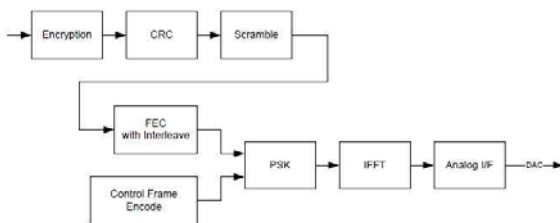


그림 5. PHY DMT 송신기 블록도
Fig. 5. DMT transmit block

이상과 같이 전력선 통신 표준방식인 PLC 기능이 구현되기 위해서는 위의 각 블록에 해당되는 프로그램이 작성되어야 하는 특징이 있다 [7][9].

3. 감정목적물의 비교항목

사물인터넷에서 전력선통신의 주요 기술은 송신단과 수신단에 MAC과 PHY의 신호변환을 하는 기능 블록을 소프트웨어나 하드웨어로 구성하는 것이다. 사용 환경에 따라 처리 속도를 고려하여 프로그램으로 구성하거나 하드웨어 칩 형태로 제작된다. 또한 통신 기술의 일반화를 위해 국제 표준으로 정한 규격을 기반으로 통신시스템을 구성한다.

목적물은 국제 표준 규격으로 정한 내용을 참조하여 시방서와 같은 표준안을 기초로 제작한 것으로 되어 있다[5][6]. 따라서 규격에 부합하는 기능블록이 구현되어야 표준안의 내용이 구현되는 것이므로, 통신규격에서 요구하는 기능 블록들이 포함되어 있는지 여부를 확인하는 방법으로 PLC 기능과 관련된 프로그램으로 제작된 것인지 확인할 수 있다.

목적물의 사실관계문서에서 고소인의 칩은 PHY 부분에 해당되는 소프트웨어를 기초로 제작되었다. 따라서 감정은 표준안의 PHY 부분을 중심으로 하는 기능 블록을 대상으로 분석한다.

표준안의 PHY를 구성하는 주요기능은 그림 5와 같이 6개의 핵심 블록으로 구성되어 있다.

- ① 암호화(encryption) - AES 와 DES 방식으로 정보데이터를 암호화 하는 부분
- ② CRC- 전송데이터의 오류 검출 검사가 가능한 점검코드를 추가하는 부분
- ③ 스크램블(scramble) - 입력데이터를 임의의 부호로 변환하는 부분
- ④ FEC with interleave - 통신오류 발생 시 정정하는 방식으로, 컨볼루션 부호, 리드솔로몬 부호, 다이버시티 매핑 등이 적용되는 부분
- ⑤ PSK - 위상편이방식으로 디지털 데이터 변조 부분

- ⑥ FFT/IFFT - 직교주파수 성분으로 분리하고, 복원하는 신호처리 부분

목적물은 위의 기능부분들이 연결되어, 전력선 통신 모뎀을 구현하고, 이를 칩으로 제작하기 위한 프로그램 소스코드로 구성된다.

4. 비교항목 유사성

각 기능 블록별 유사성을 비교하기 위해서는 양측의 목적물에서 소스코드 항목별 동작설명이나, 세부적인 기능 자료가 포함되어야 하나, 일반적으로 자료제시가 부족하여, 주석 문을 사용한 설명이 있는 경우와 직접적인 코드만 작성된 파일들만 제공되고 있다. 따라서 감정인의 전문성에 의거한 판단으로 특정 기능 블록으로 분류되는 경우는 1:1의 비교 분석이 수행될 수 있으나, 특정 기능블럭 코드로 분류되지 못하는 파일의 경우 원본을 기준으로 하는 1:N의 비교방식으로 유사성을 분석한다.

대부분의 프로그램은 verilog 언어로 작성되어 있으나, 피고소인의 C 언어와 병행되어 있다. 또한 PLC의 기능 구현에 적합한 내용이 포함되어 있는가를 판단해야하므로, 전력선 통신을 구현하는 주요 블록으로 제시된, 암호화, CRC, 스크램블러, 인터리버, PSK, FFT 등의 블록단위 비교를 수행한다. DMT 송신기를 구성하는 블록들이 확인되면, PLC 기능의 프로그램으로 판단하며, 또한 고소인과 피고소인의 파일비교에서 유사성을 도출한다.

유사도 산출은 고소인의 소스코드가 피고소인의 소스코드에 어느 정도 포함되었는지를 도출하는 것으로, 유사도 분석방법에는 원본기준과 비교본 기준 방법이 사용된다. 본 연구의 사례에서는 원본의 소스코드가 비교본에 어느 정도 포함

되었는지 여부와 이 결과에 의한 불법 도용 여부를 판별하는 사항으로, 원본기준방법으로 유사도를 도출한다.

고소인은 PLC 각 부분의 기능에 해당되는 파일의 종류를 적시하고 있으나, 피고인의 자료는 파일에 대한 설명 자료가 제공되지 않아, 해당파일이 어떤 기능의 부분인지 특정하기 어렵게 되어 있다. 파일 내에 주석으로 표기된 기능설명이 있는 경우는 대등한 비교분석이 수행된다. 그러나 특정되지 않는 파일의 경우 그림 6의 다대다(N:N) 비교방법이 가능한 exEyes 비교도구를 사용한다. 이 도구는 저작권위원회에서 보유하고 있는 파일간의 유사도 검출을 위해 제작된 것으로 다양한 선택방법으로 유사성 비교 기준을 정하고, 유사도를 분석할 수 있는 도구이다. exEyes를 사용한 비교수행 결과는 30% 이상의 유사성을 갖는 파일에 대해 세부 분석을 수행한다. 이는 30% 이하의 유사성을 보이는 결과는 유사성 판별에 의미를 두기 어렵기 때문이다. 그러나 이방법은 전체 파일에서 가능성이 있는 부분을 확인하는 1차적인 것으로, 선별된 파일에 대해서는 파일별로 육안 분석을 수행하여, 유사라인과 동일라인의 유의미성을 세부적으로 판단하여야 한다. 분석도구는 유사성판별에서 참고자료를 제고하는 역할이며, 최종적인 유사성을 판단은 직접적인 육안판독으로 도출하여야 한다.



그림 6. exEyes v5.0 실행화면
Fig. 6. exEyes v5.0 window

기능블럭의 특징이 확인된 파일은 원본과 1:1 비교가 가능한 상태이므로, 직접적인 파일비교가 가능한 도구를 사용한다. beyond compare 는 파일단위, 라인단위 그리고, 이진화일형태의 실행파일 비교가 가능한 도구 목적으로 사용된다[8]. 도구의 비교화면은 다음 그림 7과 같다.

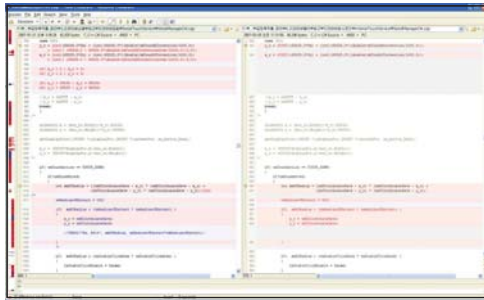


그림 7. Beyond Compare 실행 화면
Fig. 7. Beyond Compare window

이상의 방법으로 목적물에 대한 유사성을 도출한 결과는 표 1과 같다.

원본 38개 파일(15467 라인)에 대한 원본비교 방식의 유사성 분석결과는 전체 12.7%의 유사성을 보여 독자적인 개발로 판단되었다.

표 1. 기능블럭별 분석 결과
Table 1. Result of function block comparison

	기능블럭	파일수	라인수	유사라인	유사성
1	암호화	9	4530	518	11.4%
2	CRC	4	886	없음	x
3	scambler	1	151	없음	x
4	FEC	4	3760	없음	x
5	PSK	5	3372	1451	43.0%
6	역푸리에	15	2768	없음	x
		38	15467	1969	12.7%

그러나 위상편이변조 블록에서 43.0%의 유사성이 발견되어 세부분석을 하였으나, 그림 8과 같이 verilog 프로그램의 일반 구문에 해당되는 부분을 제거하면, 유사성의 의미가 없는 것으로 판단되었다.

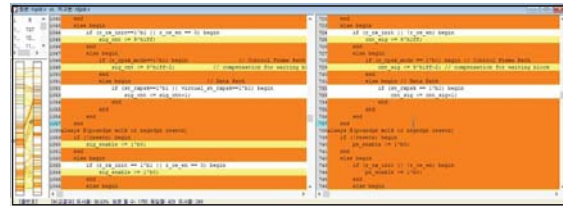


그림 8. exEyes 파일 분석 화면
Fig. 8. exEyes compare window

5. 결론

사물인터넷에서 디바이스의 소형화와 저 전력화의 방법으로 프로그램을 집적회로로 구현하여, 하드웨어적인 칩 형태로 제작하는 기술이 적용되고 있다. 그러나 칩 제작과정도 프로그램 소스코드를 바탕으로 구현되는 것으로, 동일한 기능의 칩이 생산된 것으로 의심되어, 원 개발자의 저작권을 침해하는 분쟁이 발생하고 있다. 본 연구에서는 집적회로 제작에 사용되는 HDL 언어형태의 소스코드 분쟁 시 유사성을 분석하는 방법에 대해 연구하고, 사례를 통하여 유사도를 검출하는 방법을 제시하였다. 칩 제작과정에서도 기능별 소스코드가 작성되어야 하는 특성으로 개발단계의 소스코드를 확보하고, 양측의 기능별 비교를 수행하는 방법으로 유사성 도출이 가능하고, 분쟁해결에 활용 될 수 있는 것으로 확인하였다.

Acknowledgment

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(2018R1D1A1A09082919), and this work was supported by Institute for Information & communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT)(No.2018-0-01456, AutoMaTa: Autonomous Management framework based on artificial intelligent Technology for adaptive and disposable IoT). Any correspondence related to this paper should be addressed to Kyu-Tae Lee.

참 고 문 헌

- [1] Raj Kamal, "Embedded systems Architecture Programming and Design 2nd ed.", MacGraw Hill Companies, p.5, 2015. ISBN: 0-07-049470-3
- [2] <http://blogspot.designonchip.com/2009/10/rtl-engineer.html>
- [3] Kyu-Tae Lee, Hyun-Chang Lee, and Jang-Geun Ki, "Establishment of the Subtitle on Materials for Evaluating Intellectual Ownership", International Journal of Signal Processing, Image Processing and Pattern Recognition, Vol.10 No. 9, pp.79-88, Sep. 2017. <http://dx.doi.org/10.14257/ijcip.2017.10.9.09>
- [4] M. M. Swift, B. N. Bershad, and H. M. Levy, "Improving the Reliability of Commodity Operating Systems", ACM Trans. on Computer Systems, Vol. 23, No. 1, pp.77-110, Sep. 2003. DOI: 10.1.1.107.2596
- [5] M. M. Swift, M. Annamalai, B. N. Bershad, and H. M. Levy, "Recovering Device Drivers", ACM Trans. on Computer Systems, Vol. 24, No. 4, pp.333-360, Nov. 2006. <http://u.cs.biu.ac.il/~wiseman/2os/bugs/swift1.pdf>
- [6] M. Rajagopalan, M. A. Hiltunen, T. Jim, and R. D. Schlichting, "System Call Monitoring Using Authenticated System Calls", IEEE Trans. on Dependable and Secure Computing, pp.216-229, Jul. 2006. DOI: 10.1109/TDSC.2006.41
- [7] T. Naughton, W. Bland, G. Vallee, C. Engelmann, and S. L. Scott, "Fault Injection Framework for system Resilience Evaluation", Proc. of the Resilience 2009, pp.23-28, Jun. 2009. <https://www.christian-engelmann.info/publications/naughton09fault.pdf>
- [8] Compare files and folders [Internet], 2018. http://www.scootersoftware.com/features.php?zz=features_focused
- [9] V.J. Mooney, D.M. Blough, "A hardware-software real-time operating system framework for SoCs", IEEE Design & Test of Computers, Vol. 19, No. 6, pp.44-51, Nov. 2002. DOI: 10.1109/MDT.2002.1047743

— 저 자 소 개 —



김도현(Do-Hyeun Kim)

2000.8 경북대 전자공학과(정보통신전공) 박사
2004.9~현재 국립 제주대학교 공과대학 컴
퓨터공학전공 교수.
<주관심분야> 사물인터넷, 예측 및 최적 제
어, 모바일 컴퓨팅, 임베디드 소프트웨어



이규태(Kyu-Tae Lee)

1991 고려대 전자공학과 박사
1992~현재 공주대학교 정보통신공학부 교수
<주관심분야> 신호처리, VLC, 저작권보호