Ročník 2018 electroscope Číslo I

Design of System-on-Chip (SoC) with Embedded Cryptographic Module for Internet-of-Things (IoT)

M. Kerndl[1], P. Šteffan[1]
[1] Brno University of Technology
Faculty of Electrical Engineering and Communication
Technická 3058/10, 616 00 Brno, Czech Republic

E-mail : xkernd00@vutbr.cz, steffan@feec.vutbr.cz

Abstract:
This paper proposes the design of specific System-on-chip (SoC) that can be seen on Fig.1, with basic and most used cryptographic functions such as SHA-256, RSA and AES, as well as SHA-3 candidates in form of X11 hash algorithm embedded as modules for faster processing of these functions. This article deals with Dynamic Routing that is implemented in X11 algorithm for advanced output of the hash function. The platform also includes the most common communication standards, such as UART, SPI, I2C, CAN, GPIO and can be connected to Wi-Fi, Bluetooth, Ethernet modules etc. The design and verification is done using the Virtual Platform methodology. This proposed System-on-chip (SoC) design aims to provide an example of software development and system architecture based on Virtual Platform that can be deployed to real hardware platform.

INTRODUCTION

Recent developments in Internet-of-Things (IoT) shows that the applications are getting more complex and hardware requirements are still getting higher. The key building block for these applications are Systems-on-Chip (SoC), that integrates a variety of system functions such as audio, video, modem, connectivity, imaging, etc. Typically these system functions are implemented using a mix of software and hardware [1].
Typically the most important building blocks in SoC is processor that can provide flexibility and upgradeability [1]. In other hand, there are cryptographic functions, usually used in scenarios such as connecting, security, verifying etc.
This article deals with cryptographic hash functions, that can be implemented on low-level register-transfer design (RTL) and can be dynamically routed, to achieve specific order of connected hash functions as well as more specific output. This method can be used to verify, sign or compute appropriate number of new hash values, based on the order and number of particular hash functions used. This design was created by Virtual platform method, that needs to build and connect blocks, typically written in VHDL language.

SYSTEM-ON-CHIP

Virtual platform

This method of design requires to model each component separately first, than the modeled components needs to be interconnected by bus or connections. Verification of design is next step and can be done by deploying the Virtual Platform on a host hardware and connect the IoT nodes by communication modules [2]. SoC Virtual Platform is great cost reduction and quality improvement due to time-to market requirements.
As proposed in article [3] the comparison of running such software on real hardware and SoC Virtual Platform shows the same results. The flexible development is possible without actual hardware and the software can be used immediately on the target hardware in the future. Similar design was proposed in article [4] and [5], this work is based on its proposal.

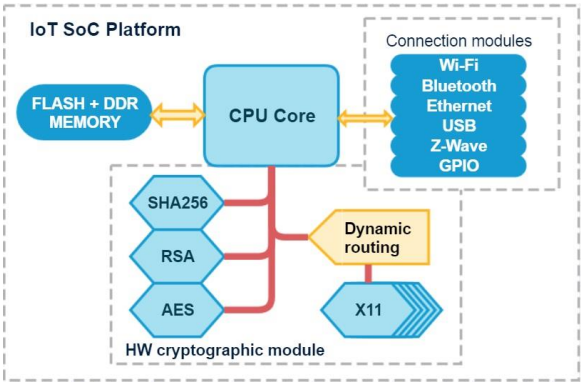

Fig. 1: Block diagram of proposed IoT SoC Platform design

HARDWARE CRYPTOGRAPHIC MODULE

The Fig.1 also shows the Hardware Cryptographic Module that is the main idea of this design. Modeled functions are connected via bus connection. The only

difference is X11 function that is passed through Dynamic Routing module both on the input and output of the partial functions. The Dynamic Routing means that the X11 algorithm (which consist of 11 different hash algorithms) can be re-routed and the inputs/outputs of partial hashing functions can be used to get new specific hash function output. This method is used to get more advanced (or specific) hash output value. Internal structure of dynamically routed multi-hash algorithm is shown on Fig.2.

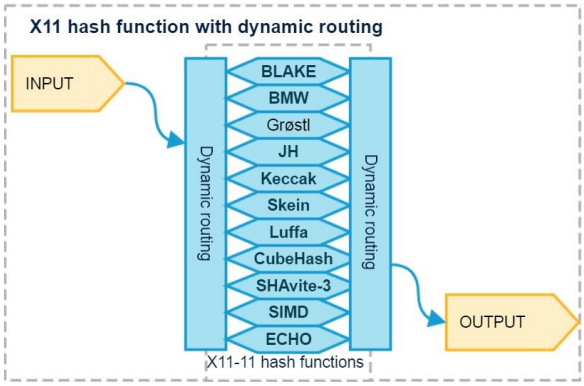

Fig. 2: Proposed Dynamic routing for multi-hash function (X11 in this case) with specific output.

As seen on Fig.2 the multi-algorithm hash function is composed by each particular hash function as a block with input and output. The input/output of each block can be routed to another particular hash function in any order so the output is determined by order, selected hash function and usage count of this selection. The specific output of partial hash function is passed to input of next hash function as per selection by Dynamic Routing. Main output is than sent to CPU Core block via internal bus. All used algorithms are "Secure Hash Algorithm-3" (SHA-3) candidates in National Institute of Standards and Technology (NIST) open competition.

Used cryptographic functions

The SHA-256, RSA and AES was chosen due to their often usage in applications. The X11 was chosen, because it is multi-algorithm hash function composed by SHA-3 candidates, and is considered as very secure and fast hash function. Three individual hash functions of X11 was selected as example of VHDL implementation of Dynamic routing. Selected hash algorithms JH, Keccak and Skein was implemented in VHDL and verified to work individually by their test-bench.

Dynamic routing

Proposed Dynamic routing is solved by usage of Multiplexer and Demultiplexer (13 to 1 in this case) that are connected via input/output ports "X" as can be seen on Fig.3. The selection is then made by writing binary combination on "Sel" pins on Multiplexer and Demultiplexer, that corresponds to multiplexing output pins ("A"-"M"). This selection causes the routing of selected hash functions in corresponding order.

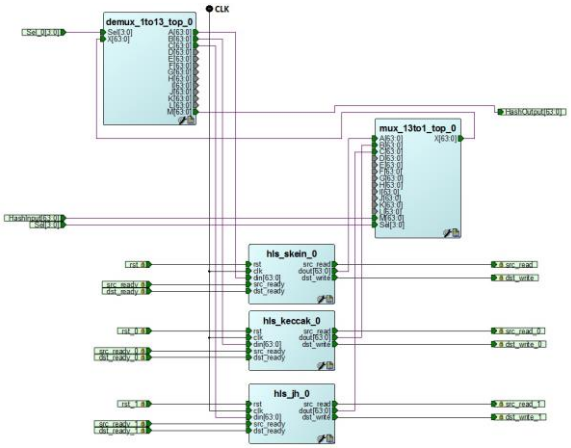

Fig. 3: Dynamic routing solved by Multiplexer and Demultiplexer with three example hash functions.

The components are driven by master "CLK" clock wire, that is connected to each component. Hash input ("M" in this case) must be selected on start-up, and routed to first hash function block. Hash function blocks are driven by master "CLK" signal, and needs reset by appropriate "RST" pin.
The block will start counting the hash function for data on "din" when all "handshake" signals are correct ("src_ready", "dst_ready") by rising edge on "src_read". when the computation is completed the output data are sent to "dout" pin by "dst_write" rising edge signal. For better control over the middle state errors, it is possible to add RAM memory block in between the Multiplexer and Demultiplexer and write/read the hash input/output from specific memory address. Unconnected Multiplexer/Demultiplexer pins ("D"-"L") will be used for the rest of X11 particular hash functions (SHA-3 candidates).

CONCLUSION

In this paper we proposed Dynamic routing realized by basic components, such as Multiplexer and Demultiplexer. This routing was used to dynamically connect inputs/outputs of particular SHA-3 candidate hash functions to achieve more advanced hash output. So the original information can be hashed or verified in more complex way. The SoC can calculate hash output for given input, based on order of particular functions. This can increase security, because the hashed information needs to be provided (or can be kept in secret) with exact order of used functions, to be verifiable by other users. This system can also be used in many scenarios, such as building decentralized distributed database also known as "Blockchain" etc.

ACKNOWLEDGEMENT

The work was supported by the Brno University of Technology project no. FEKT-S-14-2300: "New types of electronic circuits and sensors for specific applications".

REFERENCES

[1] TANURHAN, Yankin and Pieter VAN DER WOLF. 2013. Processors as SoC building blocks. *2013 IFIP/IEEE 21st International Conference on Very Large Scale Integration (VLSI-SoC)*. IEEE, , 286-287.

[2] EL-MOURSY, Magdy A., Ayman SHEIRAH, Mona SAFAR and Ashraf SALEM. 2014. Efficient embedded SoC hardware/software codesign using virtual platform. *2014 9th International Design and Test Symposium (IDT)*. IEEE, 2014(9), 36-38.

[3] LEE, Hyoung-Ro, Chi-Ho LIN, Ki-Hyuk PARK, Won-Jong KIM and Han-Jin CHO. 2017. Development of SoC virtual platform for IoT terminals based on OneM2M. *2017 International SoC Design Conference (ISOCC)*. IEEE, 2017(5), 320-321.

[4] VARGHESE, Nelson Vithayathil, Won Jong KIM, Shin Seok KANG and Hyo Seung LEE. 2018. Design and verification of secure IoT hub based on virtual SoC platform. *2018 International Conference on Electronics, Information, and Communication (ICEIC)*. IEEE, 2018(5), 1-3.

[5] YOON, Sungjae, Kihyuk PARK, Wonjong KIM and Hanjin CHO. 2018. Power estimation of cryptographic modules using virtual SoC platform. *2018 International Conference on Electronics, Information, and Communication (ICEIC)*. IEEE, , 1-3.