A message transmission system with lightweight encryption as a project in a Master subject

Jiménez, Carlos J.;Baena, Carmen; Valencia, Manuel
Dpto. de Tecnología Electrónica, Universidad de Sevilla /
Instituto de Microelectrónica de Sevilla (CSIC)
Sevilla, España
cjesus@us.es

Fernández, Juan M.; Moreno, Alejandro
Universidad de Sevilla
Sevilla, España

***Abstract*—Master subjects should ideally be very practical, to allow students to apply the knowledge they have acquired to the solving of specific problems. This paper proposes the design of a secure communications system using an SPI bus as a Master subject. The system designed uses a stream cipher to encrypt and decrypt data and allows transmission of random length messages. It also uses CRCs to check message integrity.**

Keywords— Secure communications; serial bus; data encryption; Master's course teaching.

I. INTRODUCTION

This work relates to the "Digital System Design with FPGAs" course which forms part of the Masters Degree in Products and Installations offered by the Escuela Politécnica Superior (E.P.S. - Advanced Polytechnic School) at the University of Seville [1]. This particular master degree is mainly aimed at graduates from the five degree courses organized by the E.P.S. (Industrial Electronic Engineering, Electrical Engineering, Mechanical Engineering, Industrial Chemical Engineering and Industrial Design Engineering), but it is also open to students with other degrees.

The master program has as its principal objective to train professionals capable of drawing on scientific concepts and techniques to address and solve engineering problems, taking scientific methodology and analytical, synthetic and deductive reasoning as their point of departure. The program is also intended to train professionals seeking to extend their skills and experience into new, interdisciplinary fields of activity through specialization in the interpretation, evaluation and selection of engineering alternatives.

As specific objectives, by the end of the course students should be capable of:

- Designing and managing the life cycle of products, industrial plant and electronic systems in accordance with sustainability criteria.

- Modelling and simulating products, industrial plant and electronic systems.

- Creating, developing and implementing innovative solutions for products, industrial plant and electronic systems.

- Optimizing installations and products to improve their efficiency.

- Executing entrepreneurial projects in the products and plant sector based on innovation and new business ideas.

The Master's degree is structured as a core block, to be taken by all the students, and three specialization blocks from which each student chooses one subject. All blocks carry 24 credits.

Core Block. The subjects included in this block are designed to provide students with advanced training in computer-based modelling and simulation. As part of the master degree common to all the students, the material covered is obligatory. The block has six subjects, with basic instrumental content aimed at introducing students to the field of Product and Industrial Plant Design and Development. Particular importance is attached to the scientific and technological principles underlying design processes.

Product Design and Development Block. Students wishing to earn their master degree in this specialization must complete the whole block. The material covered looks at the aesthetic aspects of industrial design, provides students with an overview of design and manufacturing methods and introduces them to the materials used in product design and development.

Industrial Plant Block. Students wishing to earn their master degree in this specialization must complete the whole block. The content covers electronic, automation, thermal, energy, hydraulic, chemo-environmental and electrical installations.

Industrial Electronic System Design and Application Block. Students wishing to earn their master degree in this specialization must complete the whole block. The block provides specialized training in the electronic systems and equipment used in industry. The course content examines both the renovation and adaptation of industries to the new technologies and the design, application, integration and

978-1-4799-6002-6/14/$31.00 ©2014 IEEE

development of electronic automation and industrial control products.

Students taking the master degree in Product Design and Industrial Installations complete their training with in-company placements (3 credits) and an End of Course Project (9 credits).

The project proposed in this paper corresponds to the specialization block "Industrial Electronic System Design and Application". The subjects included in this block are: Digital System Design with FPGAs, Intelligent System Design for Data Processing, Industrial Network Design and Management, Instrumentation with Sensor Arrays, Robotics, Intelligence and Perception, and Embedded Computers for Real Time Systems. All subjects carry four ECTS credits.

The basic objective of the subject dealt with here is to train students in design methodologies using VHDL language and FPGA devices, thereby compensating for the lack of training in electronic design with hardware description languages offered in the old 3-year Industrial Engineering degree courses and in current 4-year Industrial Electronic Engineering degrees. The subject, which is eminently practical in its approach, includes theory classes, laboratory practice and a final project. The following sections detail the content and objectives of the Digital System Design with FGPAs subject, briefly introduce encrypted communications using stream ciphers and then describe in depth the work carried out by the students and the results obtained.

II. Digital system design with FPGAs

The subject main objective is to present the methodology for designing complex digital electronic systems with programmable devices using hardware description languages and RT level synthesis tools. This methodology covers not only all the stages of the design process (from the original abstract concept through to the implementation and integration of devices in a system), but also all aspects to do with design verification and testing (from the functional level through to implementation testing).

The subject is taught using applications which illustrate all levels of design in as practical a manner as possible. The applications of greatest interest in this regard are those that can be used in industrial contexts, for example in process control, secure communications, etc., although other applications may also be considered suitable for their modern design or functional characteristics.

The designs are implemented in FPGA technologies, which make it possible to explore the proposed methodology in depth and also quickly to prototype complex digital systems.

The subject is taught in ten two-hour sessions. One part of each session (ideally less than an hour) is dedicated to theory instruction, the other part to practical hands-on training. In the last session, students present and defend the project on which they have worked during the course.

The subject content is organized into four major thematic blocks:

Block 1: Introduction to digital system design, device technologies, design methodology, CAD tools and FPGA technologies.

Block 2: VHDL hardware description language, the use of HDLs in digital design, basic VHDL language constructions and functional simulation.

Block 3: VHDL language for synthesis, the main limitations to synthesis, combinational circuit descriptions, synchronous sequential circuit descriptions, and timing considerations.

Block 4: Design of a medium complexity system. Presentation of the project to be carried out: the problem to be solved, the objectives to be fulfilled, the means and the results.

III. Stream cipher -encrypted communications

Appropriate data transmission security is indispensable in present day digital communications. Cryptography is the science of protecting transmitted data from disclosure to non-authorized persons and providing the techniques, mechanisms and tools to ensure secure private communications in open networks. In the future, all information flowing through networks will almost certainly need to be encrypted and decrypted, and to guarantee secure data transfer it will therefore be necessary to incorporate designs with cryptographic functions into the different data transmitting devices.

The most important factors affecting the protection of transmitted data are: confidentiality (secret information being transmitted to and from authorized persons), authenticity (information being transmitted from a person belonging to an authorized group) and integrity (data must be protected from malicious attacks).

Cryptographic algorithms fall into two categories: symmetric-key algorithms based on a secret key, and public-key algorithms based on two keys, one of which is secret and the other one is public. In both mechanisms, the same algorithms are generally used for both encrypting and decrypting data. Circuits implementing symmetric-key algorithms consume far fewer resources than circuits implementing public-key algorithms. This work focuses on symmetric-key cryptography. There are two types of symmetric key ciphers: block ciphers and stream ciphers. Block ciphers encrypt fixed-length blocks of data, while stream ciphers encrypt bit-groups of varying lengths. This work will use stream ciphers, with hardware implementations based on linear feedback shift registers (LFSRs). Although both types of cipher pose the problem of transferring the key from the transmitter to the receiver, stream ciphers are faster and their simple architecture facilitates the transfer of a large volume of data. This makes them ideal for low complexity systems and situations in which power consumption is severely restricted.

With this in mind, the European Union launched a project known as eSTREAM [2], [3] to select proposals for stream ciphers in both software and hardware. One of the three finalists in the hardware category was the Trivium cipher [4], and it was this solution which was used in this project.

A. Specifications of the Trivium cipher

The Trivium stream cipher is a synchronous circuit with an 80-bit key *K* and an 80-bit initialization vector *IV*, capable of generating a keystream of up to 2^{64} bits. The ciphertext message is obtained by XORing between the keystream and the plaintext message (Figure 1). Trivium's architecture comprises a 288-bit cyclic shift register with combinational logic (AND, OR, XOR gates) to provide feedback. This shift register is in turn divided into three sub shift registers of differing lengths: 93 bits, 83 bits and 111 bits respectively. Figure 2 illustrates the Trivium structure, with its three shift registers, and shows how both the feedback bits and the output bit are generated.

The Trivium encryption process begins with the initialization of the cipher's internal state. During this phase, secret key *K* and initialization vector *IV* are loaded and 4 x 288 clock cycles are run to update the cipher's internal state before it starts generating a valid keystream.

B. Communications protocol

The project objective is to establish communication between two independent, autonomous systems, one acting as a transmitter and the other as a receiver, using an encrypted data transmission.

The communications protocol adopted must meet the following requirements: it must be simple, because the intention is only to send messages from a master to a slave, and it must also allow serial transmission of random length messages. Since the application is intended for use over short distances, the communications mechanism may be synchronous: that is to say, the transmitter can generate the clock signal used by the receiver.

From the different communications mechanisms which meet these requirements, an SPI protocol was selected [5].

This protocol transmits data in series in both directions between a master device and a slave device. Data is synchronized and transmitted using 4 signals:

SCK: clock shared by both master and slave and generated by the master.

MOSI: MasterOut-SlaveIn data line.

MISO: MasterIn-SlaveOut data line.

SS: (Slave Select) Used to notify the slave that communication is about to begin.

This type of communication can also be conducted with several slaves. In this case, the master module will have an SS line for each slave and the line enabled will depend on the slave device with which the master wishes to communicate.

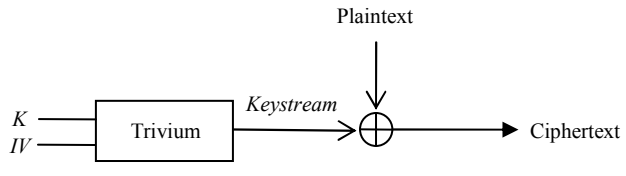

Fig. 1: Schematic representation of Trivium stream cipher.

Data transmission is clock-synchronized in such a manner that one bit is transmitted in each cycle. It is possible to send bits synchronized with the clock's rising/falling edge (polarity) and also to choose the edge the slave sample the data (transmission phase). Transmission synchronization is configured using a packet of bits sent by the master and recognized by the slave.

This communications protocol generally requires very simple hardware, with a single clock signal, but is nevertheless capable of transmitting a random length message. It is therefore very suitable for short distance communications.

The next section describes how this protocol was adapted to meet the specific requirements of the system being designed in the project.

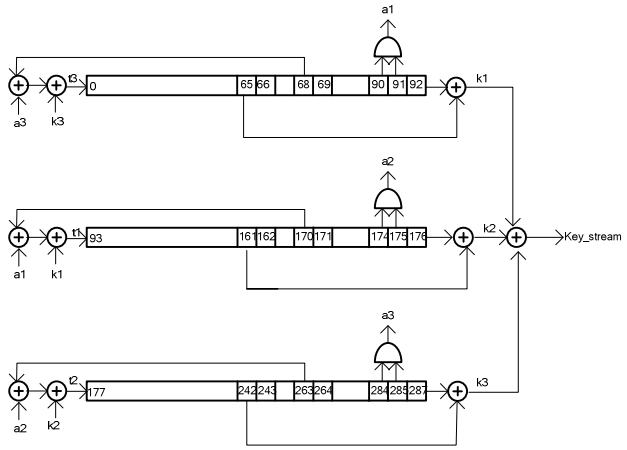

Fig. 2: Schematic representation of the Trivium stream cipher.

IV. Description of the Project Work

The project involved the serial transmission of messages encrypted using the Trivium stream cipher with an SPI protocol. The system designed has two blocks: a transmitter module and a receiver module. Both blocks first have to load Trivium's key and initialization vector and run the cipher for 1152 clock cycles before being able to send and receive messages. Once initialization has been completed, the system works as follows: The transmitter with a message to send checks whether the receiver is available to receive that message, in which case it proceeds to encrypt and send the message. When the receiver receives the message it decrypts it and stores it. To make the project work more stimulating for students and to facilitate verification, both blocks employ seven-segment displays to show the messages. As the communications protocol is synchronous, the transmitter also generates the receiver's clock signal. To ensure data integrity during transmission, the CRC corresponding to the ciphertext is sent, also encrypted, to the receiver following transmission of the message. The system was set up experimentally using Xilinx evaluation boards. The functions to be performed by each block are described below.

The transmitter block has to perform the following functions:

- Generate the message to be sent.
- Display it in the seven-segment displays.
- Encrypt it with the Trivium stream cipher.
- Generate an error detection code.
- Send the message and the CRC by means of a synchronous serial communications protocol.

The receiver block has to perform the following functions:

- Receive and decrypt the message.
- Store it in registers and display it in the seven-segment displays.
- Check that the message has not been corrupted by comparing the CRCs generated and received.

The project is intended to help students fulfill several learning objectives:

- Work with a communications protocol which, although simple, permits synchronous serial transmission of variable length data.
- Work with encrypted data, and more specifically, familiarize themselves with stream ciphers functionality.
- Use CRCs to detect errors.
- Learn design and verification methodology through a practical case.
- Learn to use Xilinx tools.
- Experimentally set up and check a circuit.

A. Description of the transmitter blocks

The transmitter module includes the following blocks:

a. Random message generator

The messages to be sent are generated randomly and shown in four seven-segment displays. A four-digit limit is imposed because the evaluation board on which the circuit was to be implemented has four built-in displays. The messages, however, can be of variable length, and may thus be displayed with one, two, three, or all four digits. This represents a binary length of 4, 8, 12 or 16 bits.

Both the content and the length of the message are generated randomly. The generated message has to be stored in a 16-bit data register, and its length in a separate, 2-bit data length register. A new message is generated when a signal is asserted (by pressing a button on the board).

b. Seven-segment displays

A second block takes the data stored in the data register and shows it in the seven-segment display. This block uses the information stored in the data length register to switch on only those displays which correspond to the message being transmitted. The way the data is displayed depends on the type of board used. In this case, the display was configured for the Digilent Basys2 board.

c. Trivium stream cipher

To encrypt the messages a Trivium stream cipher was used. To function, this cipher has the following requirements:

- A key and an initialization vector, both of 80 bits. Both the key and the IV are written into the VHDL code.
- The key and the IV are loaded at the moment feedback is received, but valid data cannot be generated until 1152 clock cycles have passed.

d. CRC generation

Another block generates error detection codes. These codes are important both for detecting errors in transmission and also for detecting potential attacks on the system through malicious changes made to the messages. As the messages to be transmitted are very short in length, a CRC of only 4 bits was chosen. However, this mechanism can easily be upscaled to a greater number of bits if the system is to be implemented with longer messages.

e. Message transmission control

The transmitter has a message transmission control block to control the functioning of the Trivium cipher and generate the corresponding communications protocol signals. The message to be sent is taken from the data register, and its length from the data length register.

This same block will also control Trivium's clock signal, advancing one clock cycle for each packet of bits to be transmitted. If no data is to be sent, the clock signal should remain disabled.

Other functions:

- When a start input pulse is received, the block proceeds to encrypt and send the data (both message and CRC).
- It will have an asynchronous reset signal.
- The message to be transmitted will be the XOR of each bit of the plaintext and the bit generated by the Trivium cipher.

B. Description of the receiver blocks

The receiver module had the following blocks:

a. Seven-segment displays

This block is similar to the one in the transmitter module. It takes the data stored in the data register and shows it in the seven-segment displays. This block uses the information stored in the data length register to display only data corresponding to the message that has been received.

b. Trivium stream cipher

This block is used to decrypt data, although essentially it works in the same manner as it does in the transmitter. As in the transmitter, Trivium cipher functions are controlled from the control block.

c. CRC checking

Before the received message is accepted, the received CRC must be checked. If the CRC is correct, the message is valid.

d. Message reception control

The receiver will have a message reception control block to control the function of Trivium and interpret the communications protocol signals.

The clock signal will be that generated by the transmitter. It has to control the Trivium clock signal in such a manner that this is only generated when data needs to be decrypted.

Other functions:

- The block has to provide a signal indicating that it is ready to receive data.
- It will have an asynchronous reset signal.
- When data is received, it will be decrypted and stored in a data register. When this has been done, the block will store the length of the received message in a data length register.

C. Communications protocol

The simplest possible synchronous serial protocol was required for message communication. In this case, a simplified SPI protocol was used. Since the intended communication was exclusively from a transmitter to a single receiver, the options of transmitting messages to several receivers and transmitting data from the receiver to the transmitter were both eliminated. However, since the receiver has to initialize Trivium, it was decided to add a signal that would allow the transmitter to be notified whether the receiver was ready to receive messages. No clock edge configuration options were implemented for generating or receiving messages. Messages are generated with rising edge and shown in the receiver with falling edge.

To sum up, for this communications protocol the transmitter generates a clock signal, a signal indicating the start of the message transmission and a data signal whereas the receiver generates a signal indicating that it is ready to receive a message. Table 1 shows a schematic representation of these signals, together with the names used to identify them.

Transmitter		Receiver	
clk	System clock signal		
send	Signal to enable start of keystream When its value is “1”, it means that data is being transmitted.	ready	Signal to indicate that receiver is ready to receive data. When its value is “1”, it means that it can receive data.
data	Signal containing the sent message.		

TABLE 1: DESCRIPTION OF SIGNALS BETWEEN TRANSMITTER AND RECEIVER.

Figure 3 shows a timing diagram with the moments when the different signals are emitted during message transmission. The message transmission procedure is as follows: The transmitter checks that the *ready* signal is at “1”. If it is, it asserts the *send* signal and starting sending data. When it has sent the last piece of data it puts the *send* signal back to "0”. The receiver, while receiving data, puts the *ready* signal at “0”, thus indicating that it is receiving data. Once the data has been received, the *ready* signal returns to "1”.
:

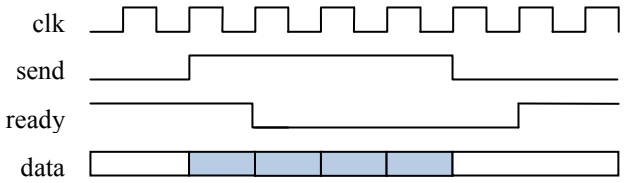

Fig. 3: Data transmission timing diagram.

D. Project execution

The work was carried out by students working in pairs. One person was responsible for designing the transmitter, the other for designing the receiver, although the design of blocks common to both modules was shared. Once the teacher had handed out the specifications, three monitoring sessions were held in which any emerging problems were solved, specification details not explained sufficiently well in the original document were clarified and verification tests were planned, both for the individual component designs and for the message transmission system as a whole.

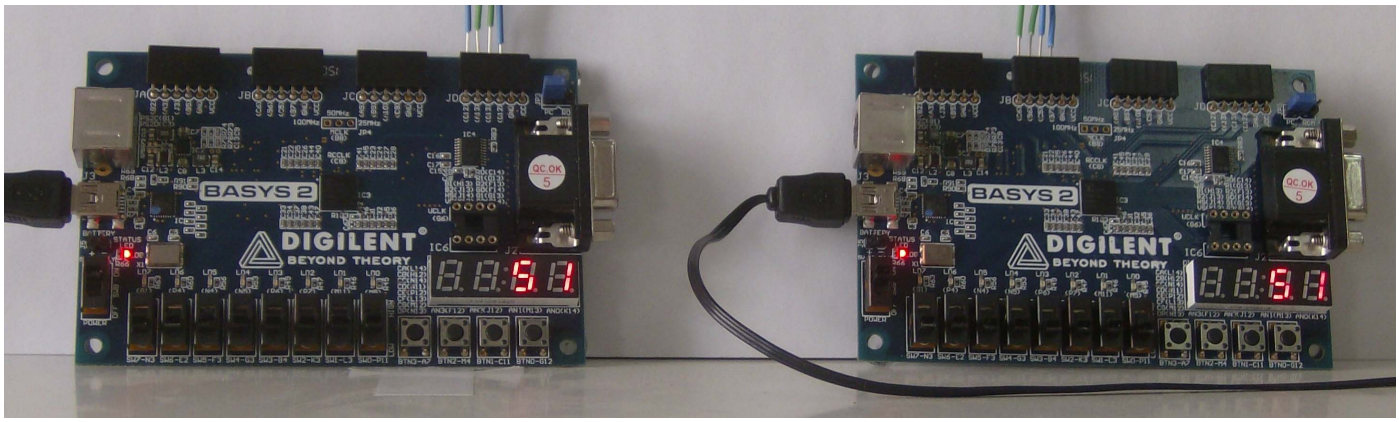

Fig. 4: Photograph of the experimental set-up.

To finish, a final session was held in which the designs were implemented on two Digilent Basys2 evaluation boards and the final test was carried out by interconnecting them to send and receive different messages. To verify that the system worked correctly, the messages that had been sent and received were checked in the seven-segment displays on each board. This mechanism, although it imposed severe limitations in terms of the number of bits in the message being transmitted, made it possible to experimentally check the functionality of the system in a very visual manner, without having to use additional laboratory equipment (more specifically, logic analyzers). Figure 4 shows a photograph of the set-up. The left hand board is that of the transmitter and the right hand board that of the receiver. The displays on both boards show the same value, meaning that the message transmitted corresponds to the message received.

ACKNOWLEDGEMENTS

This work was partially funded by the Spanish Government's Ministry of Science and Innovation, as part of the CITIES Project (TEC-2010-16870).

V. CONCLUSIONS

This paper has presented a work experience in a Master subject. It first identified the master's degree course of which the subject forms part, and then went on to describe the objectives and methodology corresponding to that subject. The project enabled students not only to familiarize themselves with an application comparable to a real world application and with commercially employed tools, but also to acquire a range of additional skills: they had to deal with a specifications sheet in which certain details were not clear in the original document and had to be clarified. They also acquired experience in teamwork, with each student assuming responsibility for a specific part of the design and with a need for continuous coordination to ensure successful data transmission.

The experience was considered very positive by both the teacher and the students involved. The system designed worked correctly at the first attempt, and all the proposed objectives were therefore achieved.

REFERENCES

[1] http://www.us.es/estudios/master/master_M066

[2] eSTREAM: ECRYPT Stream Cipher Project, http://www.ecrypt.eu.org/stream/

[3] New Stream Cipher Designs. The eSTREAM Finalists. Matthew Robshaw Olivier Billet (Eds.). Springer 2008.

[4] C. De Canniere y B. Preneel, "Trivium, A Stream Cipher Construction Inspired by Block Cipher Design Principles", eSTREAM, ECRYPT Stream Cipher Project. http://www.ecrypt.eu.org/stream/papersdir/2006/021.pdf

[5] Motorola Inc., "SPI Block Guide V03.06," February 2003.