Journal of Computer Science & Computational Mathematics, Volume 6, Issue 3, September 2016
DOI: 10.20967/jcscm.2016.03.002

Optical Encryption Device for Software Protection

Abang Annuar Ehsan[1,2*], Mohd Ridzuan Mohd Arip[2], Mohd Quyyum Sohaimi[2],
Ibrahim Ghazi[2], Faieza Abd Aziz[2]

[1]Institute of Microengineering and Nanoelectronics, Universiti Kebangsaan Malaysia, 43600 Bangi, Selangor, Malaysia
[2]Departement of Mechanical and Manufacturing Engineering, Faculty of Engineering, Universiti Putra Malaysia, 43400 Serdang, Selangor, Malaysia

[*]Corresponding author email: aaehsan@nxphotonics.com

***Abstract*:** An optical-based hardware encryption device for software protection is proposed. Software piracy and illegal usage require that software operation is secured through a form of hardware protection. Electronic-based hardware keys such as dongle keys may be vulnerable to attacks through reverse engineered technology. A simple optical device can be included as part of the hardware key where it can be used as an encryption device. Software users will require specific hardware key with built optical device in order to operate the software. The LED activated optical device can produce encrypted signals via generated optical codes solely based on the geometrical feature of the optical device. A suitable form of optical code generating device is presented in the form of 1xN asymmetric waveguide coupler.

***Keywords*:** asymmetric waveguide, dongle, encryption, optical device, software protection**.**

1. Introduction

Software products piracy is the illicit proliferation with misuse of software programs, in the event for business enterprise or individual utilization. Sometimes, it may be impossible to safeguard software against non-authorized duplicating with the use of attainable software solutions. The possibility to illegally use a software without being registered is potentially high once there exists the possibility to start a program [1]. Various approaches to battle against unauthorized software product |found today can be fundamentally split into two main categories: software-based protection and hardware-based protection. Among the predominant hardware-based software, protection techniques are dependent on unique hardware known as “dongle”. The dongle is a small device which has a various physical forms such as USB, RS232 or LPT interface and fundamentally gives protection to applications from being illegally operated [2]. Hardware key or dongle for software protection has been developed persistently in the market for many years and has managed to basically protect the software against non-authorized copying and usage. Any application to work without this key requires an intruder or attacker to modify the software or to emulate the presence of device’s key [1].

Currently, methods employed by software attackers to break through the security established on a device’s key, are basically through software modification by having the key invisible for them which is accomplished by emulation of the software code. A dongle, which is gadget controlled and ensured by the security arrangement supplier can essentially build the level of security in a non-linear manner. This enables the software protection providers with the freedom to develop specific hardware only for the purposes of improving protection [1].

Most software developers and vendors contemplate that protection based on hardware key or dongle is very hard to disrupt. However, a weak spot on them which is the communication link between the dongle and the software application logic, has been identified. Communication protection can be enhanced by employing standard well-known methods like 3DES, RC2, AES, DES, Rijndael, etc [3, 4]. Nevertheless, these methods only safeguard low-level information exchange amongst dongle and application, but cannot provide protection when attacks are performed at a higher level [2]. The operating principle of dongle protection is through binding a particular software to a hardware dongle wherein the software cannot be operated unless the dongle is joined. Dongle is accessible in the business sector in various physical structures, where the most prevalent shape now is the USB dongle [5].

In addition to the hardware-based dongle protection methods, other forms of software protection include: (I) Serial-based protection: software serial numbers provided by retailers or software companies to allow to legally use their software. These doled out serial numbers can be utilized to actuate and operate the software [6]. (II) CD/DVD protection, where an original CD/DVD must be inserted, and software installed using it. If a copy version of the CD/DVD is detected, the software will not run [5]. (III) Online Internet Activation method whereby the user is required to perform verification with a server through internet connection prior to activating the software. However, this method that is viewed as an extremely resilient technique, requires loads of concern with respect to building a protected connection and recognizing server emulation [5]. (IV) A smart card that provides protection through encryption and decryption of key pairs to protect application sections and software licenses [6]. (V) Softlock scheme, a method that involves the use of hardware component qualities and applies a Triple Data Encryption Standard to protect software licenses containing hardware segment characters [6].

All of the methods above involve the used of electronics component in one form or another. However, either dongle-based or other forms of hardware solutions combined with software protection are susceptible to being attacked by hackers. Reverse engineering is one of the techniques used by an attacker to make software modifications. Reverse engineering is described as the procedure of examining a specific framework to perceive the framework's areas and connections in order to make representations of the original framework in another structure or at an alternate level of abstraction [7, 8]. Aggressors utilize two sorts of utilities for breaking software protection - debuggers and disassemblers [1, 7].

In this paper, we propose an optical solution to this problem where the hardware encryption key is constructed using an optical device. The proposed technique will employ planar optical waveguide device technology which generates unique and foolproof optical codes. The optical codes in a form of serial numbers constructed from a combination of optical signals are unique to every hardware dongle. Verification of the generated codes is done by the license key provided by the software provider. If the wrong optical code is generated, the software verification will fail. Only the manufacturer of the software knows what code combination is being generated by the dongle. If the code matches the information on the license key provided by the manufacturer to the user, then the software will operate. Without the license key sent by the manufacturer to the authorized user and the right optical code generated by the hardware dongle, the software will not operate.

Reverse engineering on the physical optical device is almost impossible as the device is constructed solely on geometrical features of a planar optical waveguide. Only the optical designer and manufacturer of the optical device will be able to determine the optical codes generated by the device. The concept of optical-based encryption device system and optical codes generated by an optical waveguide are presented. In addition, the design of the optical waveguide based on a simple asymmetrical planar waveguide is also illustrated.

2. Hardware-based Dongle Design

The current design for hardware based dongle is generally according to the standard USB design layout. The system layout for a dongle will incorporate a symmetric encryption engine and capacity for symmetric keys; a constant memory in which the software can read and compose; a unique serial number; a kind autonomous software supplier identification number which the merchant allows to each of its dongle clients; and an entrance password to open the dongle's functionality. The communication link and interaction between the copy-protected applications with the dongle will be established and progresses only if the dongle provides correct information to the software. This association between the software and the dongle happens through solicitations to the dongle API. Figure 1 explicitly demonstrates the API's center capacities, AES-encrypt and read-write memory [9].

Hardware-based dongles are normally based on two technologies: EEPROMs and ASIC-based or microcontroller-based solutions which provide more security than the former. Dongle manufacturer that utilizes these chips will regularly hid them in order to physically conceal their identity and the known technology from intruder. However, contents stored in EEPROM which are generally comprised of plain data, are easily readable via software. Hence, it is possible for a software attacker to emulate the process of getting the right to run the specific application with a simple software patch. This procedure of breaking the security depends on perusing the entire address space and recording it in emulator [1]. A protection device based on EEPROM and with parallel-port interface is shown in Figure 2.

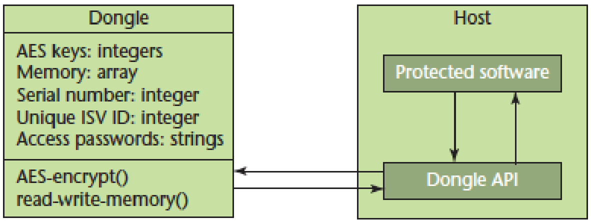


Figure 1. Dongle: dongle's interaction with the software [9]

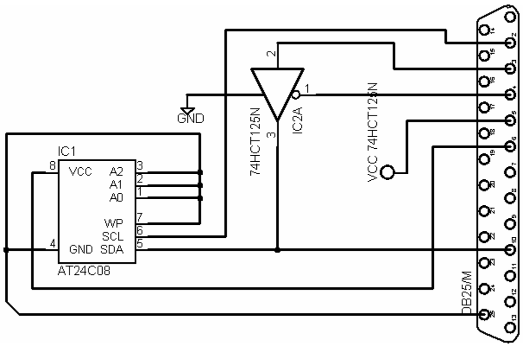


Figure 2. Schematic of dongle with parallel interface [1]

The second hardware-based design is the ASIC-based or microcontroller-based arrangements known as "astute" gadgets in light of the fact that their capacity is to convey encryption. A typical ASIC-based or microcontroller-based arrangements typically are made out of three major segments in particular; serial port, Max232, and microcontroller. The serial port on the PC will be opened by the protected software when this port sends its key to the Max232 which changes over signals from an RS-232 serial port to signals appropriate for use in TTL compatible digital logic circuits. At last, the microcontroller gadget that gets the key will send encoded information to the protected software by means of max232 under software control [8].

3. Optical Encryption Device Design

A new hardware encryption device based on optical device is presented. When a customer purchases a software, a dedicated hardware key or dongle is given. This hardware key in the form of a USB dongle is unique as it contains a specific design of an optical device or planar waveguide coupler.

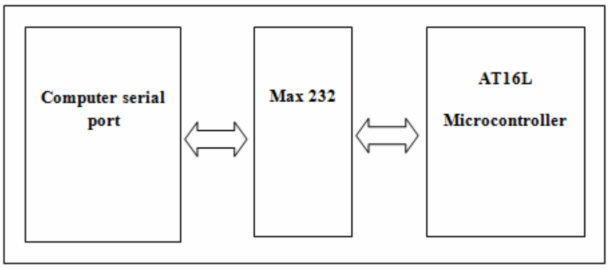


Figure 3. System block diagram for microcontroller-based hardware dongle [8]

Software manufacturer will send a specific license key used by the software to communicate with the dongle's microcontroller unit or MCU to enable an LED source built inside the dongle. Light emitted from the LED source will propagate through the optical device and a certain optical codes will be generated. An array of detectors will detect the light signal coming from the optical device. The optical device based on a waveguide coupler or splitter will be the encryption device which allows a certain optical signal to pass through. The decoding or decryption of the coded signals is done by the detector array. Signals from the detectors are then processed by the MCU and conveyed back to the software for code verification. If the application software detects the correct code, then the user may operate the software. If an attempt to change the dongle with a different set, different optical codes will still be generated. However, the software will identify that a wrong set of codes have been generated and given. Hence, the user may not be able to operate or run the software. Figure 4 illustrates the concept of hardware-based encryption system utilizing an optical code generating device.

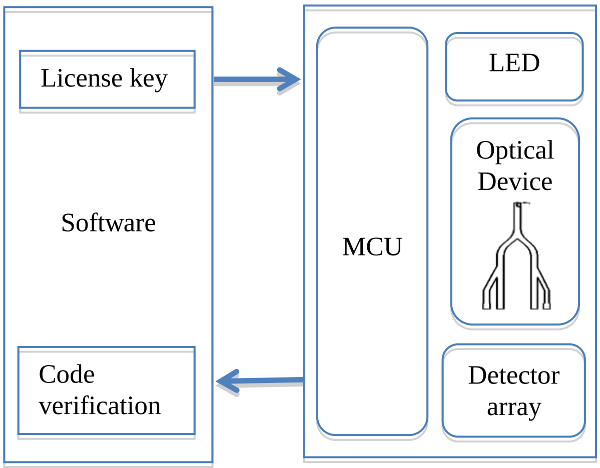
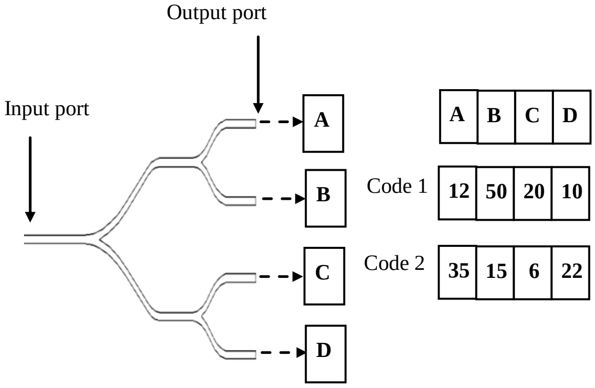


Figure 4. Block diagram of an optical-based encryption system based on optical device

The optical code generating device is based on a simple optical waveguide coupler or splitter device. A similar system using an optical device for code generation has been presented by Ehsan [10] designed mainly for a portable access security system. However, the new proposed system presented here is unique as it can be used by the software manufacturer to protect their software from an illegal operation.

The proposed system will employ similar concept as in [10] where a basic coding-decoding of the light signal transmitted from an LED or laser, is "coded" utilizing a 1xN optical waveguide coupler, where N is the number of output ports. Figure 5 shows a waveguide coupler device which can generate the optical codes. Two examples of the optical codes are given represented by code 1 and code 2. Each of the numbers in this code series represents the output power (in percentages) of the 1x4 waveguide coupler. For example, in code 1, the output power is represented by the number 12 which is 12% out off the input power. Similarly, 50, 20 and 10 each represents the percentage taken off the input power. At the point when the light source or LED is initiated, light beams will go from the LED and into the input port of the device. The optical device will split the information optical power into N-distinct optical force values. Each of the output port of the waveguide will be connected to a photodetector [11].



	A	B	C	D
Code 1	12	50	20	10
Code 2	35	15	6	22

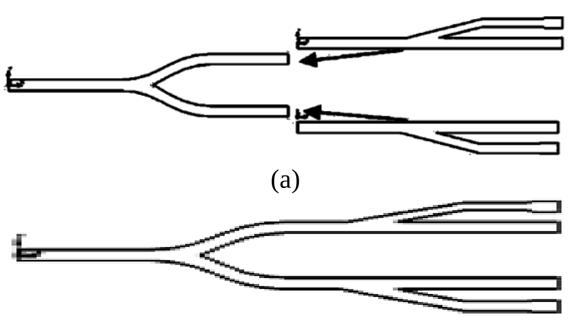
Figure 5. Waveguide coupler device for producing 4-digit code

The proposed design of the hardware-based encryption system is more secure and foolproof compared to the electronic-based system. It is well known that light signal cannot be easily copied as opposed to an electronic signal. In addition, any attempt to reverse engineering the optical part is almost impossible as the optical device has a very specific characteristics known only by the manufacturer. Knowledge of the actual waveguide design must be known in order to make duplicates of the code generating device. It is impossible from the naked eye to see what the splitting ratios of the code generating device as the optical device will have no indication of its splitting properties even when they are opened up. Furthermore, this proposed system will utilize simple detection scheme where photodetectors detect the optical signals and the converted electronic signals will be analyzed using a simple programming in the MCU.

4. Optical Waveguide Design

The optical waveguide design for the optical code generation will be based on a simple asymmetric waveguide coupler.

The waveguide design is based on a 1xN asymmetric waveguide coupler which has been developed by Ehsan [12], where N is the number of the output ports of the waveguide. The asymmetric coupler design enables different output coupling or splitting ratios to be realized. In the example in Figure 6, a 1x4 asymmetric waveguide coupler is shown. Here, a simple 1x2 Y-branch splitter is cascaded with two 1x2 asymmetric couplers. The Y-junction splitter will provide a 50% power splitting at the output port. In order to complete the 1x4 coupler structure, two asymmetric 1x2 couplers are inserted. The 1x2 asymmetric coupler is a device which allows simple optical power tap. Variation of optical power can be achieved using the 1x2 asymmetric coupler using a concept known as tap-off ratio (TOFR) [13]. This design can be easily produced by changing the geometrical size of the tap line or branch as shown by Ehsan *et .al.* [11,13]. The code generating device based on the 1x2 and 1x4 asymmetric coupler designs has been developed theoretically and experimentally by Ehsan *et. al.* [11,13]. However, the method of manufacturing for the proposed optical device suitable for this new application as an encryption device for software protection is still under research and development.


(a)

(b)

Figure 6. 1x4 asymmetric waveguide coupler design (a) 1x2 Y-branch coupler before joining with two asymmetric 1x2 couplers (b) 1x4 coupler with asymmetric branches [11,13].

5. Conclusion

This paper presents a new concept of software protection using an optical-based hardware encryption device. The hardware key or dongle can be designed with an additional foolproof feature including an optical device which can generate optical codes. Simple optical design based on a 1xN asymmetric waveguide coupler is a suitable optical device which allows light signals to be split according to a very specific ratio. This method enables optical codes to be generated easily.

References

[1] J. Jozwiak, I. Liber, K. Marczak, "A Hardware-Based Software Protection Systems", in *Proc. International Multi-Conference on Computing in the Global Information Technology* (ICCGI'07), French Caribbean, March 2007, pp. 254-261.

[2] A. Liutkevicius, A. Vrubliauskas, E. Kazanavicius, "Assessment of Dongle-based Software Copy Protection Combined with Additional Protection Methods", *Electronics & Electrical Engineering*, vol. 112, no. 6, pp.111-116, December 2011.

[3] J. Toldinas, V. Štuikys, G. Ziberkas, D. Naunikas, "Power Awareness Experiment for Crypto Service–Based Algorithms", *Electronics and Electrical Engineering*, vol. 101, no. 5, pp. 57-62, October 2010.

[4] J. Toldinas, V. Stuikys, R. Damasevicius, G. Ziberkas, M. Banionis, "Energy Efficiency Comparison with Cipher Strength of AES and Rijndael Cryptographic Algorithms in Mobile Devices", *Electronics and Electrical Engineering*, vol. 108, no. 2, pp. 11-14, April 2011.

[5] M. Usama, M. Sobh, "Software Copy Protection and Licensing based on XrML and PKCS#11", in *2011 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing* (PacRim), Victoria, Canada, August 2011, pp. 856-861.

[6] J.Y. Huang, I.H. Li, I.E. Liao, "A Software Licensing Authorization Scheme Based on Hardware Component Identifiers", *2014 IEEE Int. Conference on Information Science, Electronics and Electrical Engineering* (ISEEE), Sapporo, Japan, April 2014, pp. 1673-1676.

[7] J. Jozwiak, K. Marczak, "A Hardware-Based Software Protection Systems – Analysis of Security Dongles with Time Meters", *IEEE 2nd International Conference on Dependability of Computer Systems* ((DepCoS-RELCOMEX'07, Poland, June 2007, pp. 254-261.

[8] E. J. Mohammed Ahmed, E.E. Mohammed Ali, "Design of a Microcontroller-based Circuit for Software Protection", *International Journal of Computer Science and Information Technology & Security* (IJCSITS), vol. 3, no.1, pp. 149-154, February 2013.

[9] U. Piazzalunga, P. Salvaneschi, F. Balducci, P. Jacomuzzi, C. Moroncelli, "Security Strength Measurement for Dongle-Protected Software", *IEEE Security & Privacy*, vol. 6, no. 6, pp. 32-40, November 2007.

[10] A.A. Ehsan, S. Shaari, M.K. Abd Rahman, "Portable Optical Security Card System", PI 20071163 Malaysia Patent filing, 19 July 2007.

[11] A.A. Ehsan, S. Shaari, M.K. Abd Rahman and K.M.R. Kee Zainal Abidin "Hollow Optical Waveguide Coupler for Portable Access Card System Application", *Journal of Optical Communications*, vol. 30, no.2, pp. 67-73, Jun 2009

[12] A.A. Ehsan, S. Shaari, M.K. Abd Rahman, "Device for combining and splitting optical signals and methods for fabricating the same", PI 2009 4950 Malaysia Patent filling, 23 November 2009.

[13] A.A. Ehsan, S. Shaari, M.K. Abd Rahman, "Metal-Based 1×2 and 1×4 Asymmetric Plastic Optical Fiber Couplers for Optical Code Generating Devices", *Progress In Electromagnetics Research*, PIER 101, pp. 1-16, 2010.