ON-BOARD AUTOMATIC CERTIFICATING SYSTEM (ACS) FOR SOFTWARE DEFINED RADIO

Kazuyuki OKUIKE, Hironori UCHIKAWA, Kentaro IKEMOTO, Kenta UMEBAYASHI, Ryuji KOHNO (Yokohama National University, Yokohama, Kanagawa, Japan; kuike[gotz, ikemoto, ume, kohno]@kohnolab.dnj.ynu.ac.jp);

ABSTRACT

This article presents a novel mechanism called Automatic Certificating System (ACS) for Software Defined Radio (SDR). ACS is the system that realizes the role of the authorization authority both on-board and on-chip. This system approves each module unit changed by the software. Moreover, this article describes an approval procedure using ACS and gives an example of a modem module to illustrate the implementation of ACS. This procedure aims to realize approval on board, on chip and on line in real time. This procedure improves the flexibility of SDR compared with the FCC (Federal Communications Commission), and TIT (Tokyo Institute of Technology) proposed procedures. Furthermore, ACS reduces the amount of authentication between hardware and software. Also, ACS can adapt to future communication methods.

1. INTRODUCTION

Recently Software Defined Radio (SDR) [1][2] has been recognized as an important technology for wireless communications. Approval procedures are an essential issue for SDR. This article proposes Automatic Certificating System (ACS) that provides a highly flexible approval procedure for SDR.

Currently, every wireless communication device or system must obtain approval showing that it conforms to regulations regarding central frequency, frequency band, output power, and so on from the appropriate governmental authority before being manufactured and sold as a commercial device. However, SDR terminals use reprogrammable hardware and therefore have to obtain approval again if the software is modified.

Moreover an approval procedure for SDR is desired, which provides high security and high flexibility compared with the current approval procedure. Some flexibility is desired through on-board approval, on chip approval, on-line approval, and real-time approval.

We therefore propose ACS to satisfy these desires and show the possibility of its realization giving an example of a modem module.

The article is organized as follows. In Section 2, the Automatic Certificating System (ACS) concept and architecture is described. Section 3 outlines the Approval procedure in ACS with an example. Section 4 compares ACS with other schemes. Section 5 draws some conclusions.

2. AUTOMATIC CERTIFICATING SYSTEM (ACS)

2.1. ACS Concept

ACS certifies that reconfigurable hardware modules, which are defined by the SDR architectures [2], conform to the regulations. This is because SDR is defined by software and the system is changed by software. SDR therefore has to obtain approval when the software is changed.

The approval procedure using ACS differs from the conventional procedure. The ACS approves each reconfigurable hardware module, where as the conventional procedure approves each complete device or system. ACS can approve at a low load and in a short time, when only some of the modules are changed or updated. The ACS can therefore realize on-board, on-chip, on-line and real-time approval.

2.2. ACS Architecture

ACS is software for reconfigurable hardware, and can therefore be changed or updated. Fig.1 shows the ACS architecture. ACS consists of an I/O Interface, Data Generation Function, Database, and Certification Function.

The I/O Interface controls the input and output data that are generated at the Data Generation Function for certification. It also sends the input data to the reconfigurable hardware modules and receives the output data from the reconfigurable hardware modules. After that, it sends the output data to the Certification Function.

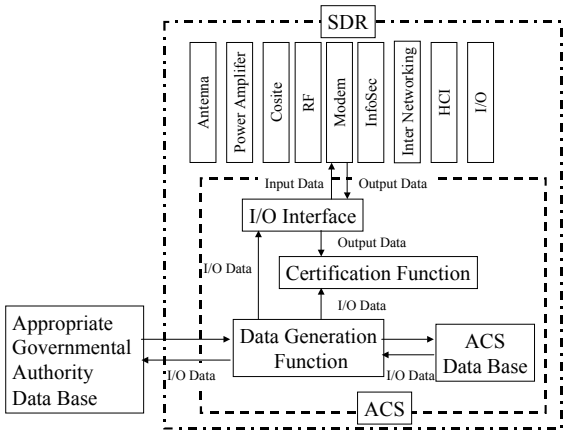

FIG 1. ACS Architecture

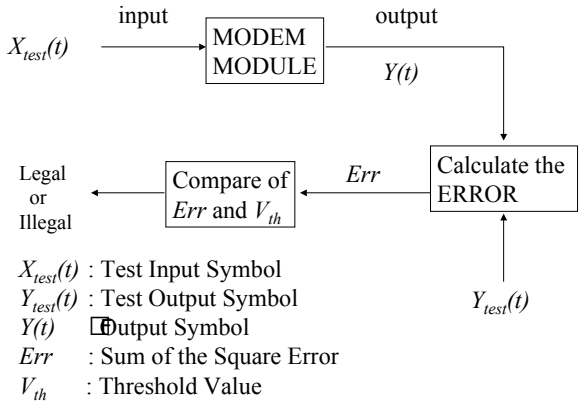

FIG 2. Approval Scheme

The data Generation Function generates the input and output data for certification from the ACS database or a database of an appropriate governmental authority. Generally an ACS database is used, but an appropriate governmental authority database is used if the new software downloaded is not stored in the ACS database.

The ACS database and the database of an appropriate governmental authority contain I/O data for certification. The ACS database exists within the SDR systems (e.g., SDR terminal, SDR base station, and so on), therefore the ACS database is smaller than a database from a governmental authority. The database of such a governmental authority contains all I/O data.

The Certification Function examines the changed module using the output data from the I/O Interface and I/O data from the Data Generation Function.

3. APPROVAL PROCEDURE WITH ACS

3.1. Approval Procedure

There are 2 types of approval procedure, namely the fixed type and reconfigurable type.

Fixed parts (e.g., Antenna Module, RF Module, Power Amplifier Module, and so on) are approved without ACS by the appropriate governmental authority using a spectrum analyzer or a network analyzer. Fixed parts can't be reconfigured, and are therefore approved only one time.

Reconfigurable parts (e.g., Modem Module, Info Sec Module, and so on) are approved using ACS when the parts are reconfigured. These parts are certified using a test symbol.

3.2. Approval procedure example for reconfigurable parts

3.2.1. Simulation Model

In this section the approval scheme is described using the modem module as an example. Firstly, the test I/O symbols are prepared from ACS data base or appropriate governmental authority data base. Secondly, the I/O Interface sends the test input symbol $X_{test}(t)$ to the modem module. The modem module transacts the input symbol $X_{test}(t)$, and outputs the symbol Y(t). Thirdly, the sum of the square error Err(t) is calculated between the output symbol Y(t) and test output symbol $Y_{test}(t)$.

$$Err(n) = \sum_{t=1}^{n} \{Y_{test}(t) - Y(t)\}^2 \quad (1)$$

The modem module is approved if the Err(n) of equation (1) is smaller than threshold value $V_{th}(n)$ of equation (2), and is not approved if Err(n) is larger. The threshold Value $V_{th}(n)$ is decided as equation (3). Ns is number of samples for one wave and n is the sample number. Noise is checked before beginning the approval procedure and assumed as an Additive White Gaussian Noise (AWGN) of modem module.

$$V_{th}(n) = 2\pi \frac{n}{N_s} + Noise \quad (2)$$

When M-PSK is $\cos(2 \cdot ft + \cdot_1)$ and N-PSK is $\cos(2 \cdot ft + \cdot_2)$, the sum of the square error err gives as equation (3).

$$err = \int_0^{2\pi} \{\cos(2\pi ft + \phi_1) - \cos(2\pi ft + \phi_2)\}^2 dt$$
$$= 2\pi + 2\pi \cos(\phi_1 - \phi_2) \quad (3)$$

Average of err of equation (3) err_{ave} is showed equation (4).

Therefore average of err for one sampling is $2 \cdot / N_s$ and we decided the V_{th} as equation (2).

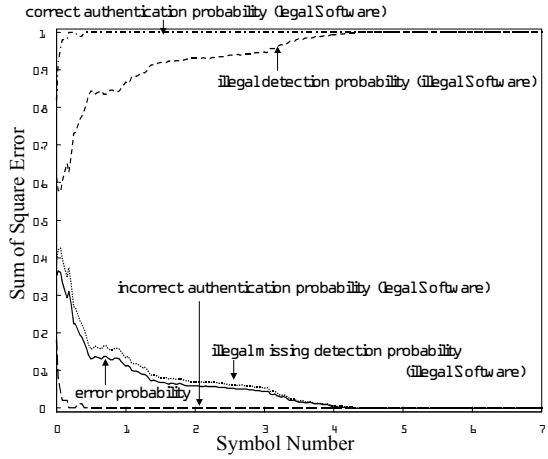

FIG 3. Simulation Result

$$err_{ave} = 2\pi + \frac{2\pi}{M \times N}\sum_{M}\sum_{N}\cos(\phi_1 - \phi_2)$$

$$= 2\pi + \frac{2\pi}{M \times N}\left\{\sum_{0}^{M/2}\sum_{N}\cos(\phi_1 - \phi_2) + \sum_{M/2}^{M}\sum_{N}\cos(\phi_1 - \phi_2)\right\}$$

$$= 2\pi + \frac{2\pi}{M \times N}\left\{\sum_{0}^{M/2}\sum_{N}\cos(\phi_1 - \phi_2) + \sum_{0}^{M/2}\sum_{N}\cos(\pi + \phi_1 - \phi_2)\right\}$$

$$= 2\pi + \frac{2\pi}{M \times N}\left\{\sum_{0}^{M/2}\sum_{N}\cos(\phi_1 - \phi_2) - \sum_{0}^{M/2}\sum_{N}\cos(\phi_1 - \phi_2)\right\}$$

$$= 2\pi. \quad (4)$$

In this simulation the following modulation schemes are used: BPSK, QPSK, 8PSK, 16PSK, 16QAM, FSK.

The simulation is evaluated using five probabilities, that are correct authentication probability, incorrect authentication probability, illegal detection probability, illegal missing detection probability, and error probability. Correct authentication probability is defined as the probability of correct authentication when the correct software is implemented. Incorrect authentication probability is defined as the probability of incorrect authentication when the correct software is implemented. Illegal detection probability is defined as the probability of detection when illegal software is implemented. Illegal missing detection probability is defined as the probability of missing the detection of illegal software.

Error probability is defined as the probability of the correct authentication and Illegal detection and shows total error probability.

3.2.2. Simulation Results

Fig.3 shows the result of the simulation. Correct authentication probability tends to 1 at some later time and incorrect authentication probability tends to 0 at some later time. This means that ACS can authorize the software completely when the correct software is implemented.

Second, the illegal detection probability tends to 1 at some later time and illegal missing detection probability tends to 0 at some later time. This means that ACS can authorize the software completely when illegal software is implemented.

TABLE 1
Comparison of ACS and other schemes

	FCC Proposal	TIT Proposal	ACS (our proposal)
Authorization Place	appropriate governmental authorities	appropriate governmental authorities	appropriate governmental authorities + Terminal
Authorization method	combination of HW and SW	separate HW and SW	combination of HW and SW
Authorization number (HW: M, SW: N)	M N	1	1
Prevent malicious or illegal software	e.g. label	e.g. label	Authorization
Role	Verification of integrity	calibration	Authorization
Realization method	HW or SW	HW	SW
Power consumption	Only during approval	always	Only during approval
Update	Change SW/HW	Data base	Change SW

4. COMPARISON WITH OTHER SCHEMES

This section presents a comparison of ACS with two other namely, FCC (Federal Communications Commission)[3], and TIT (Tokyo Institute of Technology) proposed schemes [4]. Table 1 shows this comparison.

This comparison describes that ACS has advantages at appropriate governmental authority, hardware maker and software maker have smaller load for authentication. ACS also can solve the integration of hardware and software. ACS can authenticate each module speedier in terminal. Furthermore ACS can adapt to future communication methods and repair the security holes and bugs because ACS can be changed by software.

5. CONCLUSIONS

In this article, we have proposed the concept and architecture of ACS and described the authorization procedure using ACS. Moreover we showed the realization using the example of a modem module. This example has showed that ACS can authorize the software completely in such a modem module. Furthermore we gave comparison of ACS with two other schemes and described the advantages of ACS. It is expected that ACS

can reduce the load of authorization and gives more flexibility.

6. REFERENCES

[1] J. Mitola, "The software radio architecture," *IEEE Commun. Mag.* , pp.26-38, May 1995.

[2] J. Mitola, "Software Radio Architecture: A Mathematical Perspective," *IEEE Journal on Selected Area in Communications,* Vol. 17, pp.514 - 538, April 1999.

[3] "Authorization and Use of Software Defined Radio: First Report and Order," *FCC. Washington, DC,* Sept.2001.

[4] Munkhtur Togooch, Kei Sakaguchi, Jun-ichi Takada, Kiyomichi Araki, "Automatic Calibration Unit (ACU) and ACU employed Authorization Procedure (AMAP) for SDR," *SDRF-02-I-0020-V0.00.*

[5] Lachlan B. Michael, Miodrag J. Mihaljevic, Shinichiro Haruyama, Ryuji Kohno, "A Framework for Secure Download for Software-Defined Radio," *IEEE Commun. Mag.* , pp.88-96, July 2002.

[6] Miodrag J. Mihaljevic, Lachlan B. Michael, Shinichiro Haruyama, Ryuji Kohno, "On Specific Security Requests for SDR Downloading," *Tech. Rep. IEICE, SR02-05(2002-4)* , pp.31-35, April 2002.

[7] J. Mitola, "Software Radio Architecture Evolution: Foundations, Technology Tradeoffs, and Architecture Implications," *IEICE TRANS. COMMUN.,* Vol. E83-B, No.6, pp.1165 - 1173, June 2000.