Software Engineering
2018; 6(2: 56-62
http://www.sciencepublishinggroup.com/j/se
doi: 10.11648/j.se.20180602.14
ISSN: 2376-8029 (Print); ISSN: 2376-8037 (Online)

Development of a System for Token Validation in a Prepayment Energy Meter from Wireless Recharge Source without a Server

Henry Erialuode Amhenrior, Joy Omoavowere Emagbetere

Department of Electrical/Electronic Engineering, Faculty of Engineering, University of Benin, Benin City, Nigeria

Email address:
henrino2003@yahoo.com (H. E. Amhenrior), omoavowere.emagbetere@uniben.edu (J. O. Emagbetere)

To cite this article:
Henry Erialuode Amhenrior, Joy Omoavowere Emagbetere. Development of a System for Token Validation in a Prepayment Energy Meter from Wireless Recharge Source without a Server. *Software Engineering*. Vol. 6, No. 2, 2018, pp. 56-62.
doi: 10.11648/j.se.20180602.14

Received: July 8, 2018; **Accepted**: July 19, 2018; **Published**: August 15, 2018

Abstract: This paper presents a method of Token Validation of a wireless recharge token in Prepayment Energy meter without recourse to any server. Most proposal on wireless recharging have suggested token validation at the server in the Supply Authority's office. The need for Prepayment Energy Meter to be able to validate token from a wireless source without the help of an intermediary system is highly desirable. The Prepayment Meter is realized with an ADE7755, Atmega328 and Atmega2560 in its circuitry among other components. The ADE7755 gives the load pulses that are measured and recorded by Atmega328P. Atmega2560 manages the units according to consumption. It also controls and monitors the activities of the meter and receives token for recharge from a GSM Short Message Service (SMS) platform using SIM900 as the gateway. It also comprise of a Liquid Crystal Display (LCD) for displaying unit balance and other information. The microcontrollers are programmed in C++ language with a Data Encryption Standard (DES) built in a Labview environment used in the token validation algorithm incorporated in Atmega2560. The results obtained show a mean signaling time for SMS recharging of 20.50s and 100% success rate in wireless recharging showing correct validation of recharge token by the meter. With the results obtained, token recharge from wireless sources, especially the SMS can be validated without recourse to a server anywhere.

Keywords: Token Validation, SMS, Prepayment Meter and DES

1. Introduction

Over the years, the measurement of electricity has evolved through the use of various energy meters ranging from meters that works on liquid movement which is like a sand watch through various electromechanical energy meters and then the Prepaid Meters [1-3]. The Prepayment meters are electronic in nature and therefore have some level of intelligence. A Prepayment Energy Meter enables the utility companies to collect electricity bills from consumers (consumers paying for their energy) before consuming energy. Prepayment meter does not just have Automated Meter Reading ability, but it also has the ability of prepaid recharging as well as sharing information on customer's consumption with the utility companies [4]. This ability of recharging includes token validation from any source. The most prevalent type of recharge system especially in Nigeria is the token recharge system. Server validation of token from wireless sources such as the SMS has been advocated for by some researchers. A prepayment meter should be able to also validate recharge token within its system and circuitry even if it is from wireless sources and this is the objective which this paper seeks to achieve.

2. Literature Review

There has not been enough literature on the subject of token validation from wireless sources. However, Jebashanthini et al. and also Bharat with Lokhande at various times have proposed a metering system which comprises of a PIC Microcontroller which uses a 4x3 matrix keypad for recharging the meter. The later used ZigBee technology for

communication while the former used a GSM modem [5-6]. In this system, recharge tokens are physically entered into the meter through the keypad. ZigBee and the GSM were used to transmit the tokens to the Electricity Board (EB) for the validation of the recharge token. The validation of the token before acceptance by the meter was done outside the meter. Similarly, Hiware et al. presented a system that comprises an energy metering IC, 8051 Microcontroller and communicates using GSM modem with the server. The system incorporated both the prepaid and the postpaid billing method. In the prepaid operations, consumers buys the scratch card that contains the recharge token. This is loaded by sending an SMS which contains the meter ID and the token through the keypad of the wireless meter to the Supply Authority's central server. The central server checks the validity of the token and sends a coded Short Message Service (SMS) to the meter containing the number of credit balance that will be recharged on the meter and the meter will be subsequently credited with this number of units [7].

Recently, reference [8] developed an SMS protocol-link between the modem (M20) and a microcontroller (PIC18F2550) to realize the above process in a simulated environment. According to them, the protocol main rules for any message sent from the Utility Server to the meter include @<reply number> <IPEMID> <Task Index> <Extra data@, where the IPEMID is the meter serial number. The task index were six in number showing what each message is requested to do e.g. task index "6" means recharge Energy Meter. The last digit of the 12-digit energy recharge voucher determines the amount of energy unit to be credited as recharge unit to the energy meter. However, there was no direct communication between the consumer and the meter in this system. The token validation of this system is in a server before crediting the meter.

The foregoing shows that the validation of recharge token sent to a meter through SMS and other wireless means is clearly not within the meter rather, it is in a server. The meter only receives the number of units already decoded at the server to increment the unit balance. Again, with this system, there are at least two-hops for SMS transmission in token recharging. The implication of this is that in the event of server failure or failure in any of these hops' links, the meter cannot be recharged with the already purchased token. Also, this typically is vulnerable to security threat by hacker.

In this work, the Data Encryption Standard (DES) built in a Labview environment was used in the token validation algorithm. The DES is a mirror image key type block encryption and decryption system published by the National Institute of Standards and Technology (NIST) as FIPS 46 in the US Federal Register [9]. It is based on a cipher known as the Feistel block cipher developed by the IBM cryptography researcher. It operates only on 64 bit blocks of data at a time.

At the encryption site, DES takes a 64-bit plaintext and creates a 64-bit ciphertext. At the decryption site, DES takes a 64-bit ciphertext and creates a 64-bit block of plaintext. The same 56-bit cipher key is used for both encryption and decryption [9-11]. However, the cipher key is normally presented as a 64-bit key in which 8 extra bits are the parity bits, which are dropped before the actual key-generation process. Therefore, DES expects two inputs - the plaintext/ciphertext to be encrypted/decrypted and the secret key as shown in Figure 1.

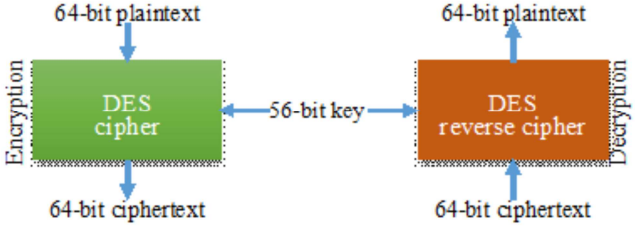

Figure 1. *DES Encryption and Decryption Overview.*

The encryption process is made of two permutations (P-boxes), the initial and final permutations, which are invertible [9]. Once a plain-text message is received to be encrypted, it is arranged into 64 bit blocks required for input. If the number of bits in the message is not evenly divisible by 64, then the last block will be padded. Multiple permutations and substitutions are incorporated throughout in order to increase the difficulty of performing a cryptanalysis on the cipher [12].

During encryption/decryption operations, DES firstly performs an initial permutation on the entire 64 bit block of data and then passed into the Round. There are 16 rounds in the DES encryption/decryption process and each of the rounds are identical and the effects of increasing their number is twofold - the algorithms security is increased and its temporal efficiency decreased [12]. Each round consists of the mixing, swapping and expansion operations. At the end of the 16th round, the 32 bit R16 and L16 output quantities are swapped and then concatenated to create the pre-output. This concatenation is permuted using a function which is the exact inverse of the initial permutation. The output of this final permutation is the 64 bit ciphertext in the case of encryption operation or plaintext in the case of decryption operation.

3. Materials and Method

The methodology used in this work is in two phases namely the hardware and the software implementation. The Token validation system software called in as a subroutine from the meter main operational and control algorithm is embedded in the hardware. The token validation system block diagram is as shown in Figure 2.

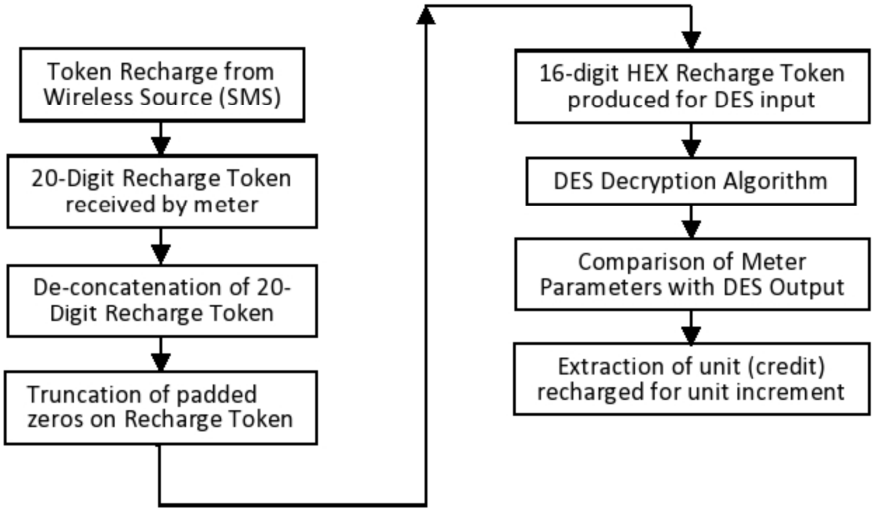

Figure 2. *Block Diagram of Token Validation System.*

3.1. Hardware Implementation

The hardware components of the design consist of the energy meter section and the wireless source of token for recharge linked by GSM network coverage.

3.1.1. The Energy Measurement IC (ADE7755) Connections

The ADE7755 IC is one of the most important components in this work. This chip does the energy measurement. Figure 3 shows the connections.

The very important inputs of ADE7755 are the transducer inputs. Pins 5 and 6 are the analog current input Pins of the IC and they are fully differential inputs. The signal from the current transducer are fed into the IC through a current limiting resistor of 10k each with C_6 and C_8 as filtering (decoupling) capacitors for the current signal channel. Similarly, the voltage ratio is fed into the IC pin 8 through a simple voltage divider network of R_{12} and R_{13} which measures the voltage across R_{13} based on voltage divider with C_9 as a filtering (decoupling) capacitor for this voltage signal channel which is also fully differential. The ADE7755 output frequency (CF) in PIN 22 is connected to the optocoupler for noise isolation before connecting to the interrupt 0 (PIN 4) of the microcontroller (Atmega328P).

3.1.2. Interfacing Between the Microcontrollers and the GSM Modem

The Atmega328P is dedicated to monitoring pulses generated from the ADE7755 for measuring the energy consumption through the meter. These pulses are received in the controller pin 4 (INT 0). This controller keeps count of these pulses and updates the Atmega2560 controller every seconds at its request. Atmega2560 pin 6 (INT4) is connected to pin 15 (PCINT1) of Atmega328P and this is used to request for update from it. Pin 14 (PCINT0) of Atmega328P is connected to pin 63 (RXD3) of Atmega2560 and this is used to obtain the pulse readings for records and other operations. Figure 4 shows the interfacing between the Microcontrollers and SIM900. The SIM900 is interfaced to the port H 0 (PIN 12) and 1 (PIN13) of the microcontroller which are the Received Data (RXD2) and Transmit Data (TXD2) pins. The receive data of the controller is connected to the transmit data of SIM900 while the transmit data of the controller is connected to the receive data of SIM900.

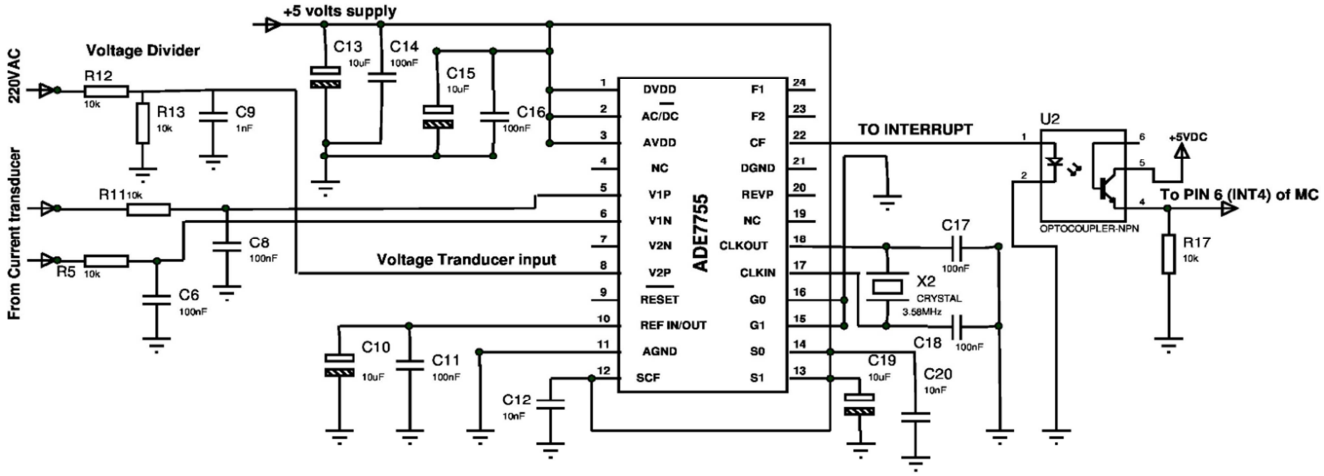

Figure 3. *ADE7755 connections.*

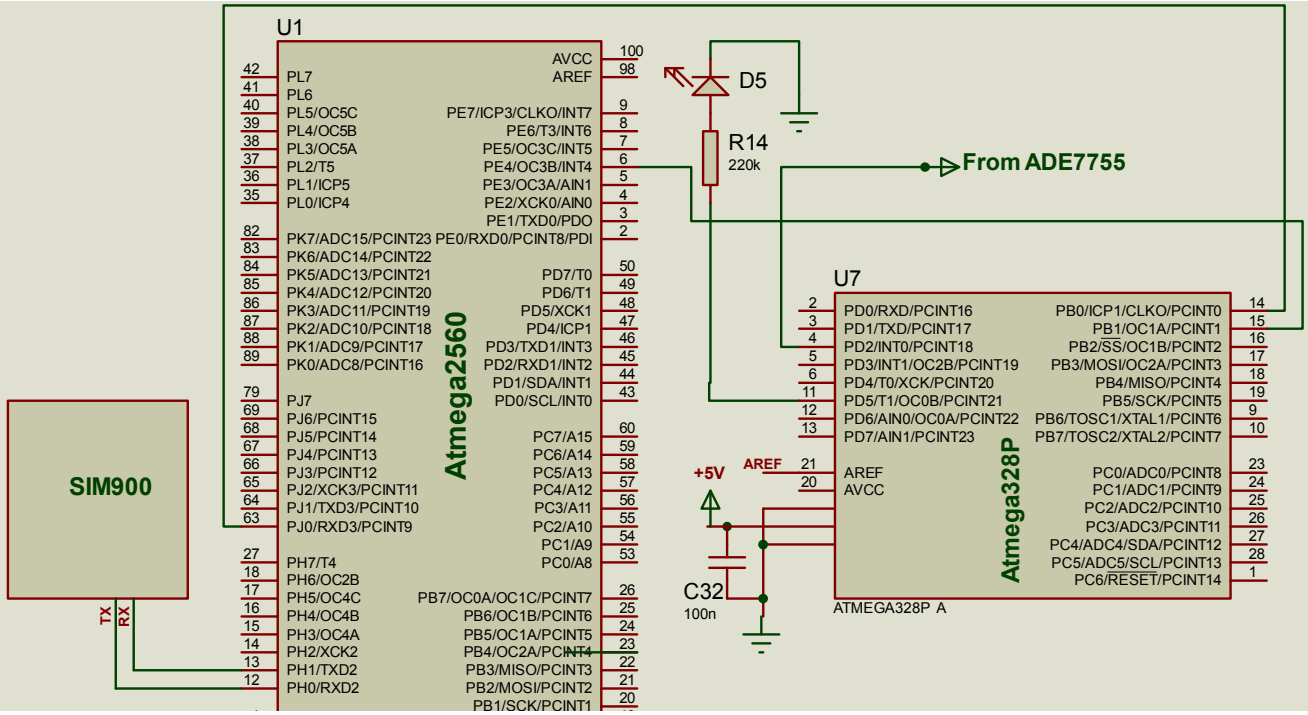

Figure 4. *Interfacing between the Microcontrollers and SIM900.*

3.2. Software Implementation

This implementation comprises of microcontroller programming in C++ language for the meter to work as designed and the DES token validation software and algorithm development.

3.2.1. Token Validation Algorithm

The Atmega2560 which contains the token validation algorithm is programmed in C++ language and calls in the DES token validation subroutine when request for token recharge is received.

The token validation algorithm used for this program is based on DES decryption. The DES as used in this work is designed in Labview environment and compiled as DLL file and called into the main program. The DES decryption uses a 64 bit or 8 bytes key for its encryption as stated in section 2.1. In DES, the input data and the key must be a 64 bit data each giving a corresponding 64 bit or 8 bytes encrypted information output [13]. The output of the algorithm comprises of the Meter ID or serial number, the Token ID, the Unit (in KWh) which consists of the whole number part and the decimal part. The output frame format is as shown in Figure 5.

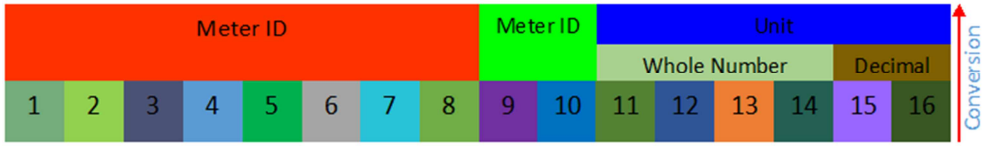

Figure 5. *DES Token Decryption output frame format [15].*

The Meter ID is the unique identifier of the meter. It contains 8 digit only for example 0103 FFAB, with a digit having character value between 0-F. The meter ID takes the first 8 digit positions or first 4 bytes of the output data format and it is in hexadecimal number system. It is validated at the hexadecimal level of DES output during token validation. The Token ID is used for the identification of all tokens generated [14]. It is a numeric (decimal) value and it increments on each token generated. It resets itself when the maximum number is reached. The Token ID takes the next two digit positions in the output data format. Each numeric digit of the Token ID is mapped from 2 digits (characters) of hexadecimal DES decryption output before validation. The Unit in Kilowatt-hour as output consists of two parts namely the whole number part and the decimal part. The Whole Number Part usually in numeric values takes the next 4 digit positions in the output data format while the Decimal Part also in numeric values takes the next 2 digit positions. Each numeric digit of the Unit is mapped from two digits (characters) of the hexadecimal DES decryption output before validation in the token validation process.

The token validation process starts when the meter receives the 20 digit numeric number recharge token through any of the recharging avenues especially the SMS source. It de-concatenates the 20 digits into four 5-digits numeric numbers. The padded zeros are discarded wherever it appears at the beginning of each de-concatenated 5-digit

numeric numbers. Starting from the leftmost (Least Significant Digit of 1) each of these numeric 5-digit number is taken at a time and converted to a four 4-digit hexadecimal numbers. The results from these conversions are concatenated into 16 digit (character) hexadecimal token (ciphertext) which was the original output from the token generation algorithm. This token is then used as an input of the DES decryption process. Figure 6 shows the digit conversion/mapping process, while Figure 7 shows the graphical DES encryption/decryption algorithm used in this study. The decryption algorithm is the inverse of the LabView graphical design of DES token encryption algorithm. The output of the decryption process is a 16 digit (character) hexadecimal plaintext.

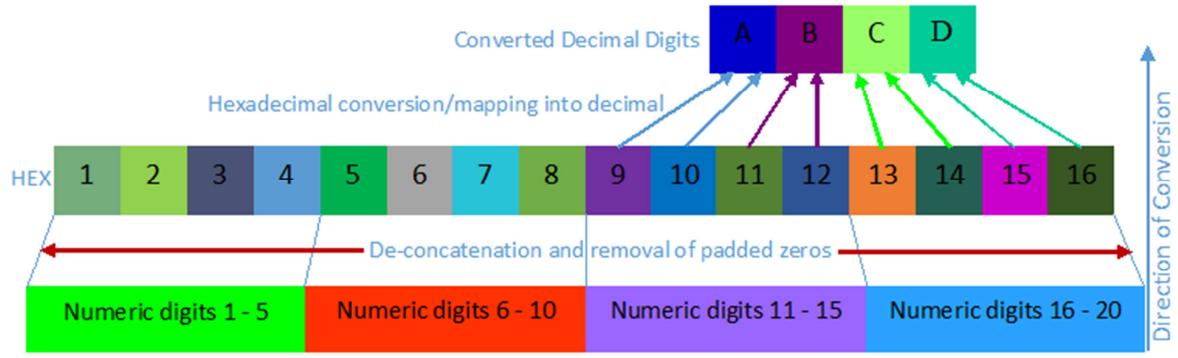

Figure 6. *Digit conversion/mapping of decimal to hexadecimal.*

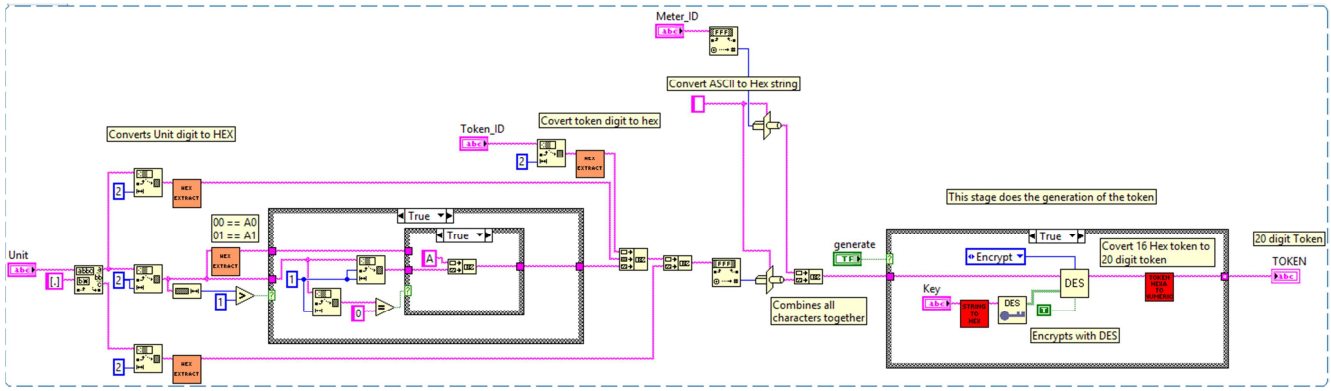

Figure 7. *DES Token Encryption/Decryption Graphical Algorithm.*

The meter takes the first 8 digits from position 1 to 8 of the decryption output frame and compares them with the Meter ID or Serial number to see if the token is meant for the meter and if this is correct, it proceeds with the remaining processes, otherwise the process is halted. It then maps the value in the next two digits in positions 9 and 10 to its decimal equivalent and then validate that the token ID has not been used. The value in the next four digits from positions 11 to 14 and 15 to 16 are converted by mapping each two digits of this 16 digit (character) hexadecimal output to its decimal equivalent to produce respectively the whole number part and the decimal number part of the Unit in Kilowatt-hour. This "numeric number value" of the unit is used to recharge the meter by incrementing the units in the meter by that value number. The token validation flow charts is as represented in Figure 8 while Figure 9 shows the SMS token recharge validation feedbacks.

3.2.2. Token Recharge System/Process Model

Parameters:

Unit balance before recharge U_B, Negative Units ($-U_B$), Borrowed Units B_U and Recharged Units R_U [15].

$$\text{Total Unit } TU = (UB + RU) - BU \quad (1)$$

Case 1: UB = -UB
Hence,

$$TU = (-UB + RU) - BU = RU - (BU + UB) \quad (2)$$

Case 2: UB = 0
Hence, TU = (0 + RU) - BU = RU – BU

4. Token Test

Having completed the development of the token validation system both in hardware and software, there was need to know how well the system performs with respect to the set objectives. In achieving this, two test were carried out in this research. Firstly, the token generation platform was used to generate token by providing all the required input and the generated token was loaded into the develop meter through SMS for validation. The result is as presented in Table 1.

Secondly, the time for a token recharge command sent in SMS to be received by the meter; execute the command and send a feedback SMS to the sender was measured and the

delay in each case recorded. In carrying out this test, the developed meter and a stop watch was used with MTN Nigeria Mobile Communications Network used at both ends. The command was sent twice and the delay in each case was observed and measured. The mean delay time was established and recorded. The result is as shown in Table 2.

Table 1. *Token Test.*

SN	Description	Numbers of Times	Numbers of Success	Numbers of Failure	% of Success	% of Failure
1	Token generation	40	40	0	100	0
2	Token validation	40	40	0	100	0

Table 2. *SMS Time Analysis.*

MSG COMMAND	TIME OF THE DAY SENT		TIME OF THE DAY RECEIVED		DELAY (SEC)		MEAN DELAY (SEC)
	1	2	1	2	1	2	
*02#	9:48	9:50	9:48	9:50	19	22	20.5

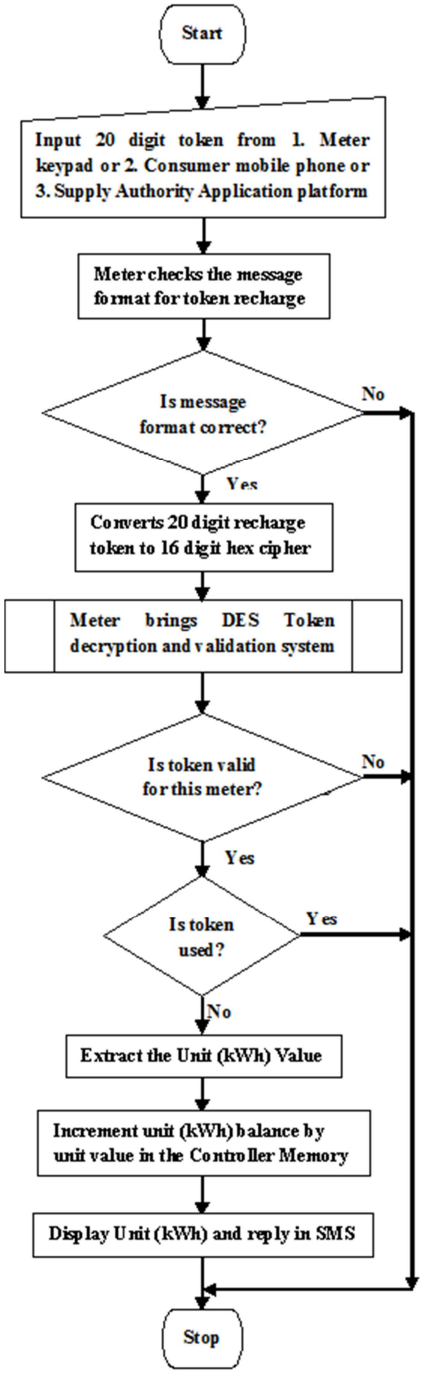

Figure 8. *Token Validation Flow Chart.*

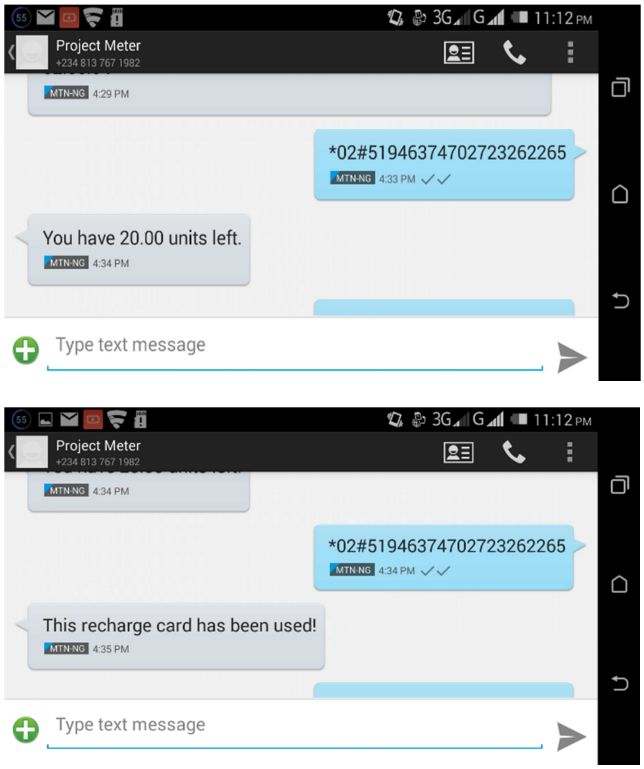

Figure 9. *Token Recharge Validation Success from SMS.*

5. Discussion

Several proposed wireless token recharge validation has been server based. Here, an algorithm that incorporates DES decryption process has been developed successfully for the validation of token from wireless source outside the system server. This algorithm is a combination of standardized cryptographic algorithm, which output is further converted and the results decomposed at various levels to compare with certain parameters as the execution progresses along the algorithm in the meter for extra security and the validated "unit" will be recharge to the meter.

From the tables above, recharge token validation rate is 100% and an SMS mean time of 20.50s for the token to be validated and obtain a feedback in SMS. Here, token validation especially from SMS as a wireless source is achieved in the meter as oppose to other models by several researchers proposing server validation of token.

6. Conclusion

In this work, the developed token validation algorithm is able to validate the token when sent wirelessly from mobile devices or from the supply authority platform and also when keyed-in from the keypad without recourse to the server anywhere, thereby satisfying the objective of this work. Through this system, consumers and operators alike can recharge their prepayment meters wirelessly and be sure it will be correctly validated and accepted in their meters. This is particularly helpful in time of exigencies when one will not be available for physical recharging of meters. Again, it helps to avoid the problem of delayed recharging and failure recharge due to server associated link failures in the server validation system as proposed by some researcher.

References

[1] Berhanu R., Ana V. M., Gómez I. M., Octavio R, José A. G., "Upgrading of Traditional Electric Meter into Wireless Electric Meter Using ZigBee Technology", In: Matías L. R., José M. F. A., Juan J. G. R., Josef L., Francisco J. B. O., and Antonio M. eds. 2011. IT Revolution- Third International ICST Conference. Spain: Springer Berlin Heidelberg. pp 84-94, 2012.

[2] Amit J. and Mohnish B., "A prepaid meter using mobile communication", International Journal of Engineering, Science and Technology, Vol. 3, No. 3, pp. 160-166, 2011.

[3] Tariq J, "Design and Implementation of a Wireless Automatic Meter Reading System", Proceedings of the World Congress on Engineering. London, Vol 1, U. K., July 2 - 4, 2008.

[4] Mejbaul H., Kamal H., Mortuza A., Rafiqul I., "Microcontroller Based Single Phase Digital Prepaid Energy Meter for Improved Metering and Billing System", International Journal of Power Electronics and Drive System IJPEDS), Vol. 1, No. 2 pp. 139-147, 2011.

[5] Jebashanthini M., Sweety A., Rini R. and Alfred K. A., "Advanced Prepaid Energy Metering System Using GSM", Methods Enriching Power & Energy Developments (Meped'13), Pp 1-5, 2013.

[6] Bharat I. and Lokhande M., "ZigBee Based Advanced Energy Prepaide Meter", International Journal of Innovations in Engineering and Technology (IJIET), Volume 3, Issue 3, Pp 109-112, 2014.

[7] Hiware R. B., Bhaskar P., Uttam B. and Nilesh K., "Advance Low Cost Electricity Billing System Using GSM", International Journal of Advanced Engineering Technology, Vol. IV/IV, Pp 51-53, 2013.

[8] Omijeh B. O., Ighalo G. I. and Anyasi F. I., "SMS- based Recharge Protocol for Prepaid Energy Billing System", International Journal of Engineering Innovation & Research, Volume 1, Issue 6, Pp 553-558, 2012.

[9] Lihaoxu, Chapter 06 Notes, CSC 5270 Data Encryption Standard. [Online], Wayne State University, 2015. Available at: <https://www.coursehero.com/file/12 439925/Chapter-06-Data-Encription-Standard/> [Accessed: 30 September 2016].

[10] Hamza Megahe "DES (Data Encryption Standard)" 2016. Accessed: <https:// www.cybrary.it/0p3n/des-data-encryption-standard/> [Accessed: 6 July, 2018].

[11] Sahin Okur, Youssef Ojeil, Michael Cuervo, Md. S. Rahaman, Dr. Chung-Yong Chan "Prepaid Energy System Senior Design II," Spring 2016, May 2, 2016. [Online] Available at: <http://www.eecs. ucf.edu/seniordesign/fa2015sp2016/g21/doc/prepaid%20energy%20sysem.pdf> [Accessed: 6 July, 2018].

[12] Sourav M., "The Data Encryption Standard (DES)", MA60031, Cryptography and Network Security. [Online via internal VLE], Indian Institute of Technology Kharagpur, 2015. Avalaible at: <http://www.facweb.iitkgp.ernet.in/~sourav/crypto.html> [Accessed: 23 January 2016].

[13] Reagan Mbitiru, Taha Selim Ustun "Using input-output correlations and a modified slide attack to compromise IEC 62055-41," IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC), November 2017.

[14] Kobus van den Berg, "The STS Prepayment StandardTID rollover in 2024 managing the change," AMEU Convention, Vanderbijlpark, 2016.

[15] Amhenrior H. E., "Analysis and Development of a GSM-Based Recharging and Monitoring of Energy Metering System", PhD Thesis, University of Benin, Benin City, Edo State, Nigeria, 2017.