

# Application Platform and Token Generation Software for Prepayment Meter Administration in Electricity Distribution Companies

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)

\*Corresponding author

## To cite this article:

Henry Erialuode Amhenrior, Joy Omoavowere Emagbetere. Application Platform and Token Generation Software for Prepayment Meter Administration in Electricity Distribution Companies. *Machine Learning Research*. Vol. 3, No. 1, 2018, pp. 1-10.

doi: 10.11648/j.ml.20180301.11

**Received:** July 17, 2018; **Accepted:** August 6, 2018; **Published:** September 1, 2018

---

**Abstract:** In this paper, an Application Platform (AP) and Token Generation Software for prepayment meter administration was developed. This is a response to the need to develop a vending software and platform that can recharge a generated token seamlessly into the meter apart from the traditional keying-in of token into meter through keypad. Also, it is motivated by the need to reduce the cost of token generation infrastructure by having a software for token generation achieved with local material. It is a platform that has wireless capability to recharge meter after vending as well as other meter administration by the Electricity Distribution Companies. The system is made up of two main modules; the server and the PC. The server's hardware is made up of Atmega 328 microcontroller which is programmed in C++. It controls the intermediate communication of the outcome of the activities of the AP such as the generated token and the energy meter. The communication component between the Electricity Distribution Company is achieved by the uses of SIM900 which has a valid Subscriber Identity Module (SIM) that enables this communication in the existing Global System for Mobile Communication (GSM) network. The PC contains the developed Application Platform using Visual Studio 2010, and has the Graphical User Interface (GUI), the Token Generation Software and the Database which is developed in Microsoft Access 2013. The AP is comprised of the Token Generation, the Registration and the Records sub-platforms. The Software for the token generation is the Data Encryption Standard (DES) compiled in a LabView environment with extra level of complexity and security built on both the input and output of the DES. Three main input of the Application Platform are the Meter Number, the Cost and the Mobile phone Number. In generating token, the first two inputs are taken into the DES system at different levels of number system and the other inputs are sourced from the AP based on the information captured for the meter. The Server is interfaced with the PC on its USB port where it also sources its power. The results obtained show 100% of 20-digit token generation through the Application Platform and seamless recharge of vended token into the meter from the Platform.

**Keywords:** Application Platform, Token Generation Software, Server, User Interface and DES

---

## 1. Introduction

Token generation and meter information management are very important part of Prepayment Meter deployment and administration. This module comes in various forms and sizes. In Nigeria, the vending system (Application Platform) generates token and these token are not seamlessly recharged into the meters because they lack wireless capabilities, as well as direct communication with the energy meters

domiciling at the consumer's premise. Also in Indian tokens cannot be recharged into the meter from the purchase platform [1]. Most of these Vending Application Systems are very exorbitant and lacks locally sourced contents. In a report by Deloitte, a power utility company, the Madhya Gujarat Vij Company Limited pays Rs. 14 for each recharge token to the manufacturer in Indian. That as the time of the report, utility

companies relied on the vendors for token generation, a situation that call for concern by the utility company especially in the event of failure from the vendor at any point of their operation [1]. This brings a drawback to the use of such vending or token generation infrastructure, as this cost will eventually be passed by the utility company to the consumer.

The system can be made smarter by introducing some value added components such as the ability to recharge meters from the platform and using simple components that are less costly but highly efficient and good in encryption system to achieve a secured token generation and meter administration.

## 2. Literature Review

Token is a means of data transportation mechanism used in transferring purchased credit from the point of sale or vending (generation) to the prepayment meter [2]. One of the standard in token vending system is the Standard Transfer Specification (STS). The STS ensures the security of prepayment token systems and the interoperability of multivendor [3]. In STS standard, two token technology are defined namely, the disposable magnetic token and the numeric token [2, 4, 5, 6]. The former transfers the token data on a disposable magnetic stripe card while the latter transfers the token data as a numeric string of 20 digits which can be printed on a receipt or any paper [2, 4]. These token are made to be meter specific as part of the security to avoid recharge token theft [1].

A few researchers notably Omijeh, Ighalo and Anyasi; Shomuyiwa and Ilevbare have looked into this area of token generation through simulation method. However, the method used remain unclear in actual implementation. Omijeh, Ighalo and Anyasi developed an SMS protocol-link between the modem (M20) and a microcontroller (PIC18F2550) to realize wireless recharging in a simulated environment. They assumed a 12-digit energy recharge voucher produced by the Utility Company and using the 12<sup>th</sup> digit to determine the amount of energy to be credited as recharge unit to the energy meter. The activities where represented in an index form and the proposal did not discuss the ways of actual development of application platform and token generation [7]. Also, Shomuyiwa and Ilevbare in the design and implementation of their meter stated that they generated 16-digit energy token for the meter using a software application program. According to them, this software was developed using Delphi and C language [8]. They however did not show how this was achieved. MAOSCO Limited built a product they called MULTOS which they opined could help secure the prepayment meter system as it could be used in the Secure Transfer Specification (STS) Protocol especially in the generation of numeric codes and tokens in vending station for recharging meters [9].

The usefulness and importance of computers and network in our daily activities involves the sending of

information, and most transmitted information requires that security be added, otherwise it will be captured by people whom the information are not meant for. Encryption systems secures data from unwanted agents from having access to correct decoding of the information. It changes the data into a form which is secured through a secret key and this key is also required to decrypt the data. Many encryption types exists namely the Skipjack, the Data Encryption Standard (DES), Triple DES, the Advanced Encryption Standard among others [10]. The DES and the Triple DES are among the most popularly used cryptographic algorithm methods [11].

The Data Encryption Standard (DES) is a veritable encryption standard and this form the basis of the token generation software algorithm of this work. The DES is a replica key type block encryption and decryption system issued by the National Institute of Standards and Technology (NIST) as FIPS 46 in the US Federal Register [12]. It is a ciphered Feister block developed from the cryptographic work of IBM which works with only 64 bit blocks of data at a time.

During the encryption operations, it admits a 64-bit plaintext and produces a 64-bit encrypted (ciphertext) output. Similarly, during the decryption operations, it admits a 64-bit ciphertext and produces a 64-bit block of non-encrypted (plaintext) output. It uses a 56-bit cipher key for both operations namely, the encryption operations and the decryption operations [12-14]. The Encryption or cipher key is the cryptographic key that is employed in the course of changing information or message into a form that cannot be understood except those that have the particular cipher key for that information [11, 15]. It is usually added to manipulate the message to make the information confusing and non-decodable to unauthorized persons. Usually, the cipher key is a block of 64-bit data where 8 additional bits are used as the parity bits, and these are removed prior to the real key-generation operations. Hence, DES takes two inputs - the plaintext/ciphertext to be encoded/decoded and the secret key as shown in Figure 1.

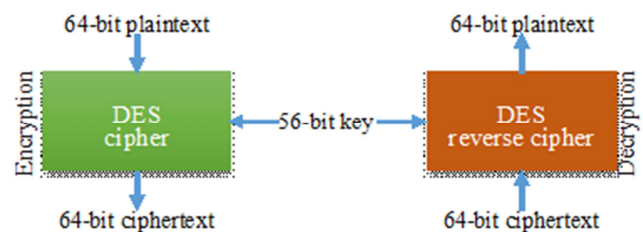


Figure 1. DES Encryption and Decryption Overview.

The encryption process consists of two permutations (P-boxes), the initial and final permutations, and these can be inverted [12]. As soon as a plain-text data is admitted for the purpose of encryption, the data is organized into 64 bit blocks needed for input. The last block is usually padded if the admitted data bit number cannot be evenly divided by 64. Several permutations and substitutions are integrated all through so as to compound the cryptanalysis rigor of the

cipher [16].

In the process of encryption/decryption, DES firstly carries out an initial permutation on the entire 64 bit block of data and then sends it into the Rounds. There are 16 rounds in the DES encryption/decryption process and each of the rounds are exactly alike and the effects of increasing their number is twofold - the algorithms security is increased and its temporal efficiency is decreased [16]. Each round is made up 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. Methodology

The methodology of this work is in two main stages namely, the development of the graphical user interfaces as the Application Platform and the development of the Token generation software.

#### 3.1. Application Platform and Server

This is the platform for the administration of the system by the utility company and for managing the database of consumers. It is made up of two parts namely the server part and the application platform part which is the Graphical User Interface (GUI) part of the utility company. The software for the server was developed in C++ language, while that for the application platform was developed using visual studio (visual basic). The server software is embedded in the Atmega328P (Arduino Uno) microcontroller and it communicates with the meter through a GSM modem (SIM900); while the Application Platform Software and User Interface reside in the PC. This platform consist of the following sub-platforms namely, the token generation platform, the registration platform and the records platform. It also consists of a Database which is developed in Microsoft Access 2013 where records are kept.

The server software includes the Serial software with the `#include <SoftwareSerial.h>` command at the beginning to set the protocol for communication on the serial port. At initialization, the server program sets both the Arduino Uno Atmega328P Serial port and GPRS Shield (SIM900) baud rate to 19200bps so as to synchronize communication with the PC and sets the counter to zero.

At start-up, the PC gets all COM ports available and set it to the last COM port detected and connects the server. It then sets Serial Port 1 to the selected COM port; sets the baud rate to 19200bps, the same for the energy meter and sets other serial port properties. This then opens the serial port and the controller writes a command to detect the server hardware which consists of the Arduino Uno Atmega328P microcontroller, the GPRS Shield and the SIM900. If the server hardware is detected, it connects it to the database of the system which is designed in Microsoft Access 2013. The controller then sends a command (AT+CMGD=1, 4) to delete all read messages. A directory to the folder “\Prepaid Files\Registrations” is then created for the storage of the database information as it is entered into the user platform if it does not exist, otherwise, it will display the message “This path exist already!”

#### 3.1.1. Token Generation Application Platform

This application platform was developed in Visual Studio 2010 and used for the generation of meter recharge token when all the necessary inputs are supplied. Figure 2 show the token generation flow chart, while the appearance screenshot of this platform is shown in Figure 3.

It is the main platform for the user application system as it serves as an entry point into the registration and records platform of the meter administration system through the file menu. The platform consists of editable textboxes for the input of data for the generation of recharge token namely the meter ID Number, Meter SIM Number, the Cost and a non-editable textbox namely “No. of Units.” The editable inputs are the inputs that must be supplied before the generation of token. The platform also contains seven buttons which functions are described in Table 1.

When the inputs are correctly entered, and the “Generate Token” button is clicked, the software subtracts any charge or debts that may have been incurred or as may have been prescribed by the regulatory authority. Thereafter, it divides the remaining amount with the tariff rate and generates the number of equivalent units. This and other inputs are then fed into the DES algorithm to generate an encrypted 8 bytes hexadecimal token with 16 digit characters which is then further converted to 20-digit numeric token. This token is displayed in the token editable textbox which can then be recharged through any of the means of recharging the meter. This token can be recharged to the meter from this platform by clicking the “recharge” button and the token will be transferred to the “Test” editable textbox. Clicking the “Test” button sends the token to the meter through the server hardware.

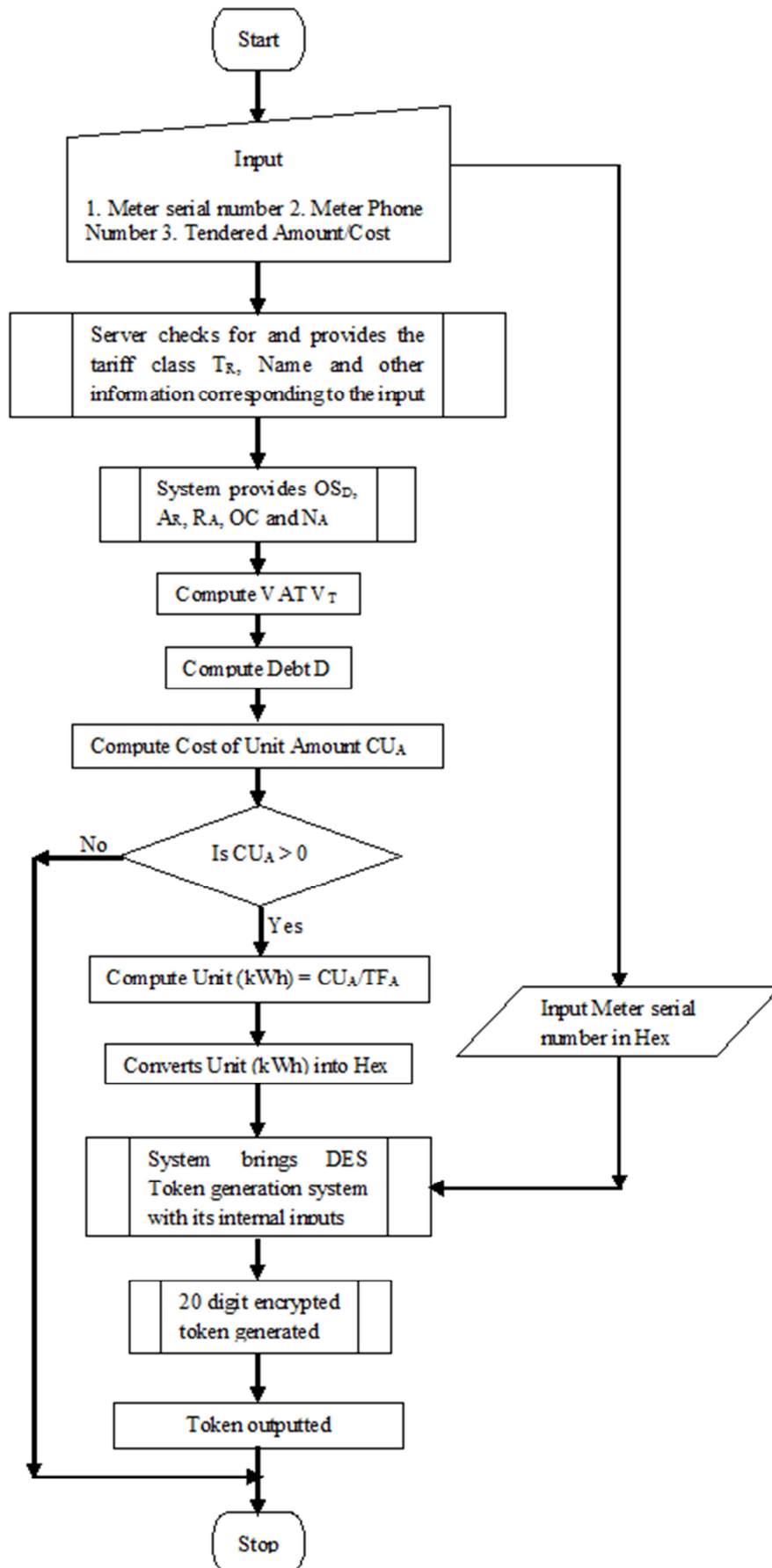


Figure 2. Token Generation Flow Chart.

Figure 3. Token Generation Platform.

### 3.1.2. Unit Purchase Mathematical Model

The inputs into this model are the Tariff Class Amount  $TF_A$ , Debts  $D$ , Value Added Tax  $V_T$ , Outstanding Debts  $OS_D$ , Other Charges  $OC$  that may be prescribe by the supply and regulatory authorities, Cost of Unit Amount  $CU_A$ , Refund Amount  $R_A$  and Tendered Amount  $T_a$ .

$$\text{Units (kWh)} = CU_A / TF_A \quad (1)$$

$$\text{Total Amount } T_A = T_a + R_A \quad (2)$$

But  $V_T = 5\%$  of  $T_A$

And,

$$CU_A = T_A - (D + V_T) = [T_a + R_A - (D + V_T)] \quad (3)$$

$$\text{And } D = (OS_D + OC) = (A_R + N_A + OC) \quad (4)$$

Where

$$OS_D = A_R + N_A \quad (5)$$

Where  $A_R$  = Arrears and  $N_A$  = Negative unit amount

Note:  $N_A$  is the equivalent cost of negative units inputted into the server as debt when “clear credit” token is given to a meter on negative unit to clear the negative units to zero.

Substituting (4) into (3) gives

$$CU_A = T_a + R_A - (A_R + N_A + OC + V_T) \quad (6)$$

Substituting (6) into (1) gives

$$\text{Units (kWh)} = [T_a + R_A - (A_R + N_A + OC + V_T)] / TF_A \quad (7)$$

If  $N_A$  is inputted into the server as debt, clear credit token is usually not given to the meter to zero the negative units before purchase. In this case, the negative units will be deducted from the purchased units when recharged into the meter. Hence, the above equation (7) will go thus:

$$\text{If } N_A = 0 \quad (8)$$

Thus, substituting (8) into (6) gives

$$\text{Units (kWh)} = [T_a + R_A - (A_R + OC + V_T)] / TF_A \quad (9)$$

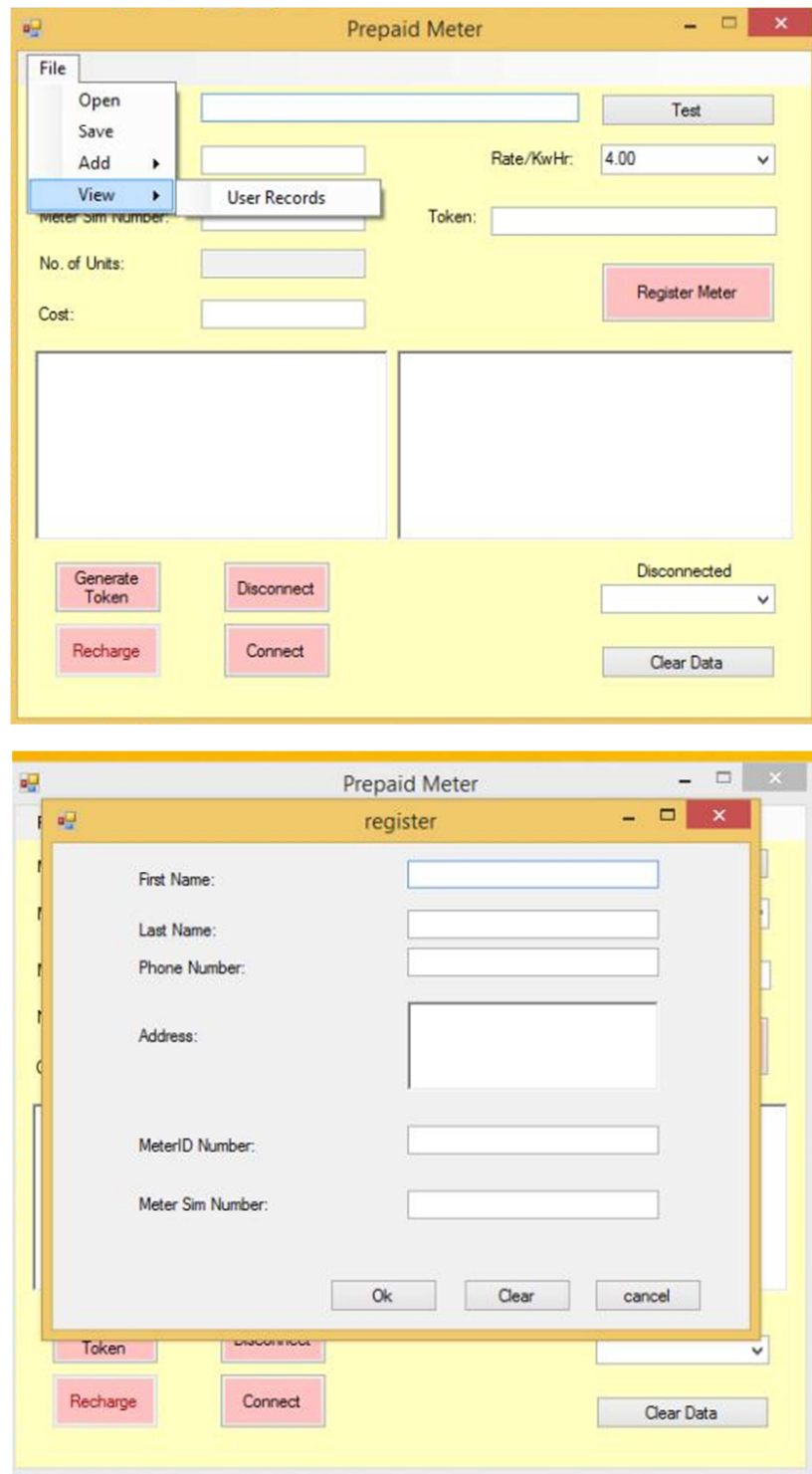
The output of either (7) or (9) usually serve as the unit input into the token generation or encryption system [17].

### 3.1.3. Registration Application Platform

This application platform developed in Visual Studio 2010 is used to register or capture the meter and the consumers' information into the database primarily for the purpose of generating token and also to keep consumers' records. The appearance screenshot of this platform is as shown in Figure 4.

**Table 1.** Functions of Application Platform Buttons [17].

SN	Button	Function
1	Generate Token	Used to generate 20 digit token when clicked after inputting the required data
2	Recharge	Used to send the token to meter when clicked after token generation
3	Disconnect	Used to disconnect meter from the server remotely
4	Connect	Used to connect meter from the server remotely
5	Test	Serves the same purpose as “Recharge Button”
6	Register	Used to register a meter in the server when clicked
7	Clear Data	Used to clear the generated token prior to another generation operation



**Figure 4.** Screenshot of Registration Application Platform.



This platform can be accessed through the file menu in the main application platform by selecting “Add” in the dropdown menu and “User Records” in the submenu. It contains six textbox field namely First Name, Last Name, Phone Number, Address, Meter ID Number and Meter SIM Number. It also contains the “Ok”, “Clear” and “Cancel” buttons.

When the fields are filled correctly and the “Ok” button is pressed, it sends the information into the database. This information is converted and stored in the excel file created through the directory “\Prepaid Files\Registrations.”

#### 3.1.4. Records Application Platform

This application platform like others was developed in Visual Studio 2010 and used to view consumers’ records. The appearance screenshot of this platform is as shown in

Figure 5.

This Platform also can be accessed through the file menu in the main application platform by selecting “View” in the dropdown menu and “User Records” in the submenu. When opened, it will display the information entered into the database earlier during registration namely the First Name, Last Name, Phone Number, Address, Meter Number and the Meter SIM Number. It displays this for one consumer at a time. There are buttons to view the next or previous consumer/meter records. To exit this platform, the “cancel” button is clicked. Records can also be viewed in the excel file format in the directory “\Prepaid Files\Registrations”. But this records opens up all the consumers’ information serially in a page to be view at once.

Figure 5. Screenshot of Records Application Platform.

#### 3.2. Token Generation Algorithm

The token generation algorithm used for this program is based on DES encryption. The DES as used in this work is designed and developed in Labview environment and compiled as DLL file and called into the main program of the Application Platform. The DES encryption uses a 64 bit or 8 bytes key for its encryption. 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. The inputs to the algorithm are the Meter ID or serial number, the Token ID,

the Unit (in KWh) which consist of the whole number part and the decimal part [17].

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 input data format. The meter ID is in hexadecimal number system and therefore is taken into the input of the DES without further conversion. The Token ID is used for the identification of all tokens generated. 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 2 digit positions in the input data format. Each numeric digit of the Token ID is mapped to two digits (characters) in hexadecimal after conversion for DES input. The Unit in Kilowatt-hour as input 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 input data format while the Decimal Part also in numeric values takes the next 2 digit positions. Some mathematical processes involving some parameters such as the tariff class, charges of any type as may be prescribed by the supply authority and the regulatory agency such as the Nigeria Electricity Regulatory Commission (NERC) are involved before arriving at the total value to be entered as unit in Kilowatt-hour. The indebtedness of the consumer and any other stipulated charges are also deducted from the tendered amount and the remaining amount is divided by the cost of the tariff class to obtain the Unit in Kilowatt-hour. The value of the Unit is in decimal number system. Each numeric digit of the Unit is mapped to two digits (characters) in hexadecimal after conversion for DES input [17].

The input frame format is as shown in Figure 6. The total number of digits (characters) input is 16 in hexadecimal or 8

bytes. The first 4 bytes takes the Meter ID which is already in hexadecimal number system and so does not undergo any conversion before input to the DES. The last 4 bytes are in decimal number system which are usually converted to hexadecimal before input into the DES algorithm by mapping a digit of the decimal value to two digits of the hexadecimal equivalent as shown in Figure 7. The above four inputs in hexadecimal are fed into the DES encryption process used for encoding the information. After the information has been encrypted, the resulting output is an 8 bytes hexadecimal token with 16 digit characters.

The 16 digit (characters) hexadecimal token (ciphertext) output from DES are converted into a 20 digit numeric value by converting each 4 digit (characters) of the hexadecimal token into 5 digit numeric character. If the resulting numeric digits are not up to 5 digits, zeros are added to the front of it to have a constant 5-digit numeric conversion. This resulting four 5-digit decimal number is concatenated to form the 20 digit recharge token. This last conversion also added another level of complexity and security into the token generation algorithm apart from that of the DES. Token generation flow chart is as shown in Figure 2.

Meter ID								Token ID		Unit					
										Whole Number			Decimal		
1	2	3	4	5	6	7	8	9	10	11	13	13	14	15	16

Figure 6. DES Token Generation input frame format [17].

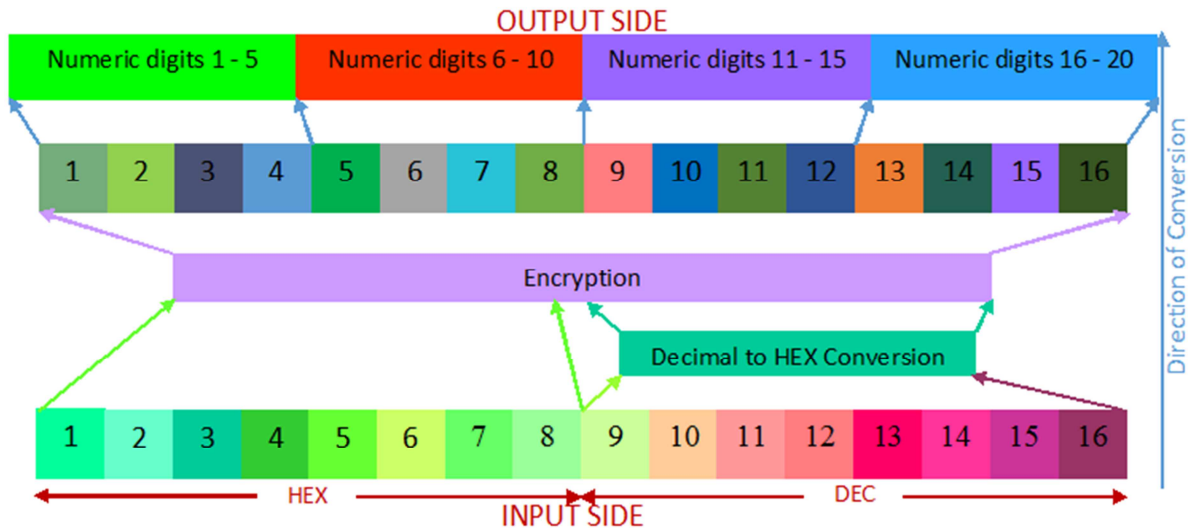


Figure 7. Digit conversion/mapping of decimal to hexadecimal.

## 4. Token Test

In this test, the token generation platform was used to generate token by providing all the required input and the generated token was recharged into the developed meter through the keypad, the mobile phone and the application platform for validation. The result is presented in Table 2.

## 5. Discussion

As evident from the token generation platform in Figure 3, this work has provided a platform for token generation using the DES encryption system in a modified form. A 20-digit token generation system has been successfully developed.



Again, from the results in Table 2, the success rate in using the developed platform is 100%.

The Application Platform is a user friendly interface developed in Visual studio environment. It is made up of sub-platforms such as the token generation, the record creation and the record viewing. This platform is use as a tool for meter operations management and as a means of creating records and database. The meter administration process start in the application platform by capturing the meter details like the meter serial number or ID, the telephone numbers of the meter and the customer; the consumer's name, address, tariff class etc. These enable token vending for the meter and proper information management in the system. The platform also has the ability to recharge the vended token into the meter wirelessly.

The token generation software enables the 20-digit recharge token to be generated for meter recharge. To vend, the Application Platform is opened and the required information namely the meter serial number or ID, the meter phone number and the amount of money tendered are provided and when the "generate" button is clicked, operations are transferred to the token generation software. Base on the model above, debts and charges are calculated and deducted and the tariff class is applied to produce the number of units required. This goes into the encryption system which output is the 20-digit recharge token. This token can be recharge into the meter through any of the stipulated means.

## 6. Conclusion

This work has provided an application platform for meter administration and a token generation software using the DES encryption system in a modified form. A 20-digit token generation system has been successfully developed and tested to be at optimum performance. This system has made token recharge more flexible, easy, and token can be recharge

wirelessly into the meter from the recharge application platform.

Application Platform and token generation are key to the administration of prepayment meters in Electricity Distribution Companies especially for keeping databases of consumer location and ownership history as well as the vending history. The token generation system is a major security point of the prepaid metering system. The selection of input into the token generation system as in this case makes it a good token system. The carefully selected inputs namely the meter number, the amount to be vended, the meter mobile phone number and an internally generated token ID made this system secure and efficient. It is a user friendly graphical interface as developed in this work requiring the operator to supply the necessary input for any of the operation to be carried out. The 20-digit token system is the standard token system that is an open system compatible with multi-vendor meter system as oppose to the 12-digit and 16-digit token systems.

The token generation system as developed in this research can easily be adapted for seamless online token purchase, recharging of meters through the internet and Automatic Teller Machine (ATM) when integrated with the banking and payment systems. This will require the agreements of the parties involved such as the power supply authority, the banks and the payment switching system vendors. With this, consumers can recharge their meters through their ATM, Credit cards, the internet and several other means. If this method is adopted, consumers will have to login into the internet webpage or the ATM and provide the needed information. The payment system will access the consumer's bank account and credit the supply authority's account from the consumer's account provided. This will then grant access to the token generation platform of the supply authority which then supply the token and forward same to the meter for recharging wirelessly.

**Table 2.** 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

## References

- [1] Deloitte, "Evolving measures for the effective implementation of Prepaid Metering in the country," Deloitte Touche Tohmatsu India Private Limited, 2011.
- [2] Roy K., "Standard Transfer Specification Guidelines," Measurement and Control Department, National PTM&C, Transmission Group, Eskom, 1995.
- [3] Don T. "Prepayment Standard Transfer Specification: An IEC standard," 2015.
- [4] R. J. Anderson and S. J. Bezuidenhout, "Cryptographic credit control in pre-payment metering systems," *Proceedings 1995 IEEE Symposium on Security and Privacy*, Oakland, CA, USA, 1995, pp. 15-23. doi: 10.1109/SECPRI.1995.398919.
- [5] Paul J. "Developments in Prepayment Standardization-The Role of The STS Association," 2001.
- [6] International Electrotechnical Commission (2003-09) "Electricity Metering – Payment Metering Systems – Part 41: Standard Transfer Specification"
- [7] 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.
- [8] Shomuyiwa D. A. and Ilevbare J. O., "Design and Implementation of Remotely- Monitored Single Phase Smart Energy Meter via Short Message Service (SMS)", *International Journal of Computer Applications*, Volume 74 No. 9, Pp 14-22, 2013.
- [9] Christopher Torr, "Using Established, Proven Standards to Build a Secure Smart Meter Infrastructure", MAOSCO Limited, 2017.

- [10] Lantronix, Inc., "Encryption and its Importance to Device Networking," pp. 1-13, 2018.
- [11] Ratnadewi, Roy Pramono Adhie, Yonatan Hutama, A. Saleh Ahmar, M. I. Setiawan, "Implementation Cryptography Data Encryption Standard (DES) and Triple Data Encryption Standard (3DES) Method in Communication System Based Near Field Communication (NFC)," *Journal of Physics: Conf. Series* 954 (2018) 012009, pp. 1-8. Doi:10.1088/1742-6596/954/1/012009.
- [12] Lihaoxu, Chapter 06 Notes, CSC 5270 "Data Encryption Standard," Wayne State University, 2015.
- [13] Hamza Megahe "DES (Data Encryption Standard)," cybrary, 2016.
- [14] Sahin Okur, Youssef Ojeil, Michael Cuervo, Md. S. Rahaman, Dr. Chung- Yong Chan "Prepaid Energy System Senior Design II," Spring 2016, University of Central Florida, May 2, 2016.
- [15] PTS Working Group, "Information Supplement: Cryptographic Key Blocks," PCI Security Standards Council, pp. 1-16, 2017.
- [16] Sourav M., "The Data Encryption Standard (DES)", MA60031, Cryptography and Network Security, 2018.
- [17] Amhenrior Henry 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.