# Implementing an Onsite Image Reconstruction Model for Wireless Sensor Nodes Based on Incremental Reprogramming Approach

Eronu Majiyebo Emmanuel
Department of Electrical/Electronics Engineering
University of Abuja
Abuja, Nigeria

Ibrahim Tashiwa Emmanuel
Department of Electrical/Electronics Engineering
University of Abuja
Abuja, Nigeria

John Mormi Habila
Department of Electrical/Electronics Engineering
University of Abuja
Abuja, Nigeria

***Abstract*—This work examines the challenges of employing the incremental/differential approach in reprogramming procedures as implemented in selected number of wireless sensor networks technologies. In particular, we examine the energy cost of using the fundamental tool (Rsync variants) in extracting the difference between the old and new code image as well as the impact of its adoption on the reprogramming process. To address some of the deficiency attributed to the use of Rynsc variants in WSN incremental reprogramming processes we propose a model that uses the enormous resources available in the base station or server to facilitate the extraction of delta and formulation of onsite image reconstruction control directives. In contrast to existing approaches based on Rsync variants, the propose model circumvents the use of message-digest algorithm (MD5) in detecting the differences between the original and modified firmware. Hence, reducing several cycles of data block and control information exchange between the base station and the sensor nodes. In addition, saving enormous amount of energy expended. We demonstrate the efficiency of our proposed model by comparing the energy cost obtained from adopting the proposed model to those based on the use of Rsync variants. Results obtained shows that the proposed model realizes the lowest delta size and lesser energy consumption (reduces the transmission cost by about 15%) compared to other similar tools used in incremental approaches [1, 2, 3].**

***Keywords—Wireless Sensor Network; Reprogramming; Algorithm; increamental Approach***

## I. INTRODUCTION

Wireless sensor networks implementation involves the deployment of hundreds or thousands of sensor nodes over large territories to monitor and control events of interest. To physically alter the nodes functionalities dynamically in response to evolving application demands is practically not feasible. Hence, concerted efforts has been invested into developing various approaches for updating sensor nodes mainly using a number of Over the Air (OTA) dissemination techniques.

Several of these approach has been discussed in literature and we have provided an extensive review about their merits and demerits in [4]. Of interest is the increase/differential reprogramming approach. The approach depends extensively on the use of Rsync [5] and its variants [6]. This work examines the challenges of employing the incremental/differential approach in reprogramming procedures as implemented in selected number of wireless sensor network technologies.

We address some of the deficiency attributed to the use of Rynsc variants in WSN reprogramming processes by proposing a model that uses the enormous resources available in the base station or server to facilitate the entire delta extraction procedure. Unlike existing approaches, an exchange of data between the base station and the sensor nodes in order to detect differences using a message-digest algorithm based on a cryptographic hash function that yields a 128-bit hash (MD5) is completely circumvented.

The approach involves the use of a file differential tool developed for the purpose of extracting delta from two files at the object code level (the original and modified version presented in hexadecimal file format). Instead of using the expensive checksum and MD5 calculations to detect the difference between the two files, the approach employs the use of the memory addressing structure to isolate modified data by taking into consideration the memory flashing pattern applicable to most flash memories contained in wireless sensor nodes' microcontrollers. In addition, the approach allows for easy reconstruction of the image contained within a sensor node to take on the current state of the application's objectives.

We demonstrate the efficiency of our proposed model by comparing the energy cost obtained from reprogramming procedures that adopt our model to processes based on the use of Rsync variant.

## II. RELATED WORK

We examine the energy cost of using the fundamental tool (Rynsc variants) in extracting the difference between the original and modified code image as well as the impact of its adoption on the reprogramming process. Rsync usage limitations and challenges entails the following: MD5 excessive computation demands, Difference not detectable in bytes but rather in chunks and MD5 proven loopholes [7, 8].

Rsync and RDIFF algorithm [5] employs non-intersecting fixed-sized blocks for matching similar data contained in both the modified and original files. The two files are segmented into chunks, and for each chunk, a rolling-checksum and an MD5 are computed. Employing these checksums, the delta is extracted with reference to either blocks that are already present in the original version, or the complete content of the

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 5 Issue 06, June-2016

modified blocks. While the rolling checksum implementation is made to be as fast as possible, the MD5 checksum implementation in sensor nodes is not suitable. The obvious drawbacks of the algorithm is that if two blocks vary in a single byte, the whole block content is then included in the delta. In addition, the sensor nodes perform costly MD5 computation for each block of the binary image when adopted during incremental reprogramming process. Milosh, et al. altered Rsync such that all the costly operations involving delta script generation are implemented on the host computer and not on the sensor nodes [9]. Furthermore, it makes sure the expensive MD5 computation is allowed only when the low-cost checksum matches between the two blocks [9]. If no matching block is identified then the algorithm moves to the next byte in the new image and the same process is repeated until a matching block is identified. While the probability of collision is not negligible for two blocks having the same checksum, with MD5 the collision probability is negligible [9]. To improve the precision of the scheme in the rare case when two dissimilar blocks have the same MD5 hash, Zephyr performs a byte-by-byte comparison when MD5 hashes match [9, 10]. A byte-by-byte comparison done out of context is unsuitable when dealing with machine codes produced to be executed on a microcontroller. Physical addresses of data locations always vary whenever variations occur in the new image file. Taking a common reference point for the purpose of comparison becomes an issues.

## III. METHODOLOGY

We address some of the deficiency attributed to the use of Rynsc variant in WSN reprogramming processes by proposing a model that uses the enormous resources available in the base station or server to facilitate the entire delta extraction procedure. Unlike existing approaches, that exchange of data and control information between the base station and the sensor nodes in order to detect differences between the two files using MD5, in the case of

The design and implementation of this work is concentrated on two software components: The first one relates to the base station or the monitoring and control center while the second one involves the sensor node. We implement a file database management system that is link to an interactive web based platform. The interactive web based platform is composed of two components. The first component runs a delta extraction routine and the second generates an appropriate image reconstruction control directive. The control directive facilitate the reconstruction of the image contained within targeted sensor nodes to reflect the new image intended. Highlights of the design and implementation of each of these components' operational objectives are illustrated in Figure 1.0. The terms *OrigA*, *OrigD*, *ModA* and *ModD* signify addresses and Data values of the original and modified files respectively. The *n* represents the index employed in retrieving the *OrigA*, *OrigD*, *ModA* and *ModD* contents in words for a 32bit microcontroller.

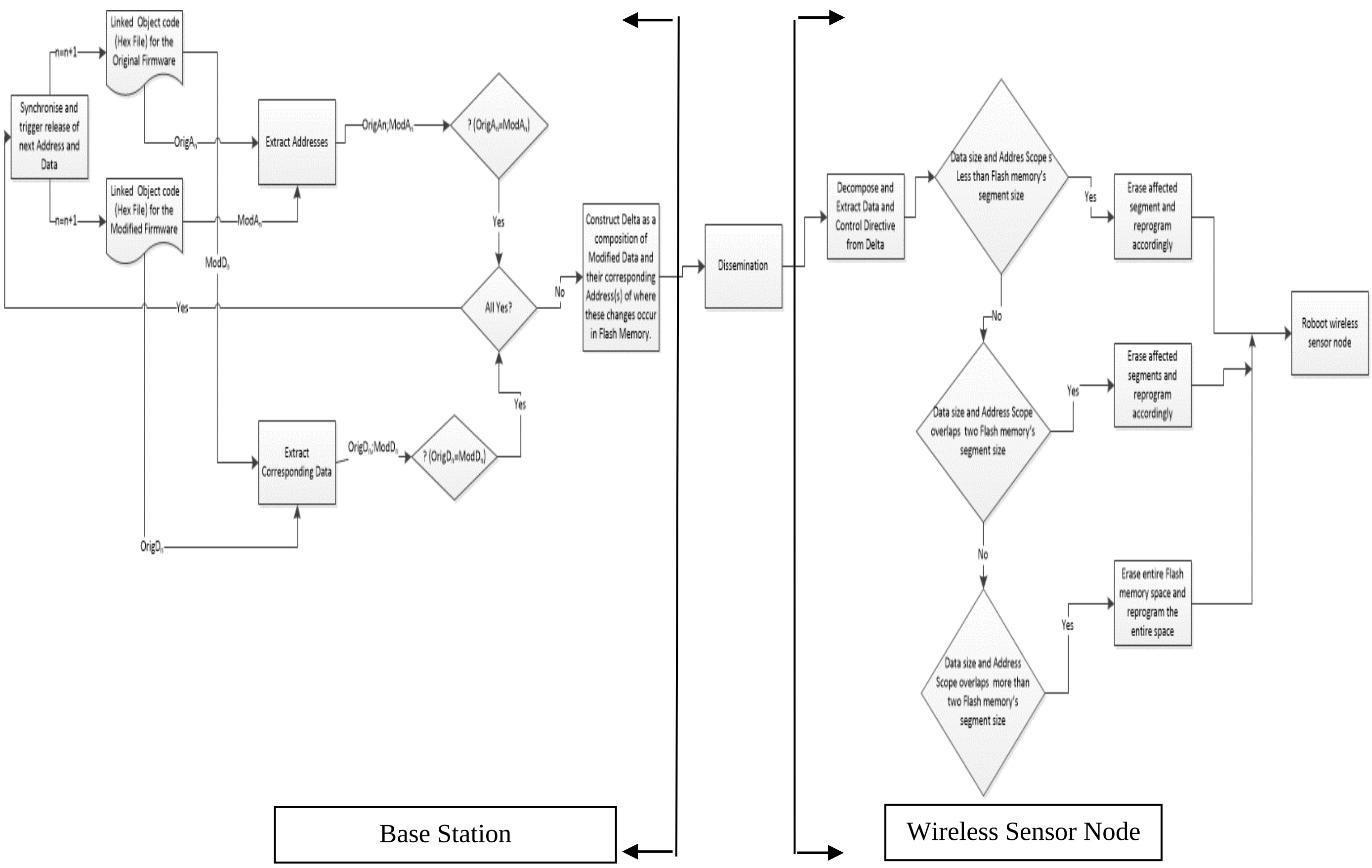


Fig. 1. Highlights of the design and implementation of each of these components' operational objectives.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 5 Issue 06, June-2016

Algorithm 1 listing highlights the re-flashing algorithm implemented in the wireless sensor nodes. It is a modified version of what was presented in our previous work [11]. The terms $a_k$ and $d_k$ denote the address and data of delta extracted at the server or base station. where k signify the index or position of each member in the set with cardinal value of m. subsequently, the symbols $SO_i$ and $SN_j$ represent the segments containing the original and modified firmware in flash memory, while $i$ and $j$ are their respective locations within a set of n segments. *T(r)* serves as an array for storing the index of modified segment(s).

ALGORITHM 1: FLASH PROGRAM MEMORY RE-FLASHING

```
r = 0; j = 1; k = 1
While ( j<=n) do
While (k<=m) do
If (a_k => start address of SN_j & a_k <= start address of SN_j)
T( r ) =  j
end if
k++ ; r++ ;j++
 end while
end while
Select  |T( r )|
Case 1:
Erase and reprogram within  SOT(0)
Case  2:
Erase and reprogram SOT(0) and  SOT(1)
Case  >2:
Erase and reprogram entire memory space
end select
```

## IV. RESULT AND DISCUSSIONS

We perform testbed experiments to evaluate the reprogramming energy expended by the two approaches in a testbed similar to that adopted in a previous work [12]. The testbed consist of nodes which are compose of Microchip PIC32MX320F128H microcontroller and a Microchip MRF24J40MB transceiver for implementing low-cost WSN.

We examine the energy consumed during dissemination as a function of delta size generated by the proposed approach compared to Rsync variant used in Zephyr [10], via change cases involving modifications effected at various source code program structures. The ensuing modified files were paired with their respective originals and utilized in the two approaches.

Figure 2 shows the energy consumed during dissemination (taking the number of transmissions as a metric for energy consumption) by the two approaches under examination. The approach proposed reduces the transmission cost by about 15% when compared to Rsync variant used in Zephyr [10].

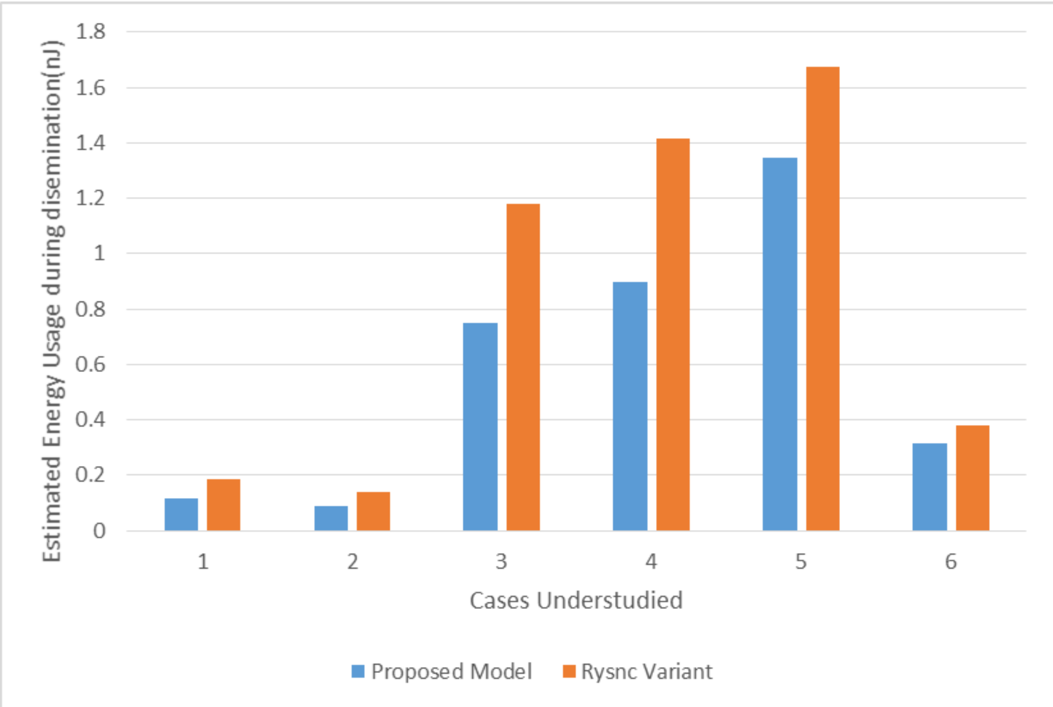


Fig 2. Energy consumed during dissemination by the two approaches under examination

## V.CONCLUSION

A design and implementation of a wireless sensor node's onsite image reconstruction model based on the incremental reprogramming approach is presented. The work addresses some of the challenges attributed to the use of Rsync variants as delta extraction tool. Reducing several cycles of data block and control information exchange between the base station and the sensor nodes which subsequently saves enormous amount of energy expended constitute the core contributions of this work. The proposed model realizes the lowest delta size and lesser energy consumption (reduces the transmission cost by about 15%) when compared to other similar tools used in incremental approaches [1, 2, 3].

## REFERENCES

[1] W. Dong, M. Biyuan , C. Huang, Y. Liu and C. Chen, "R3: Optimizing Relocatable Code for Efficient Reprogramming in Networked Embedded Systems," in *IEEE INFOCOM Mini-conference*, Turin, Italy, 2013.

[2] R. K. Panta, S. Bagchi and S. P. Midkiff , "Zephyr: efficient incremental reprogramming of sensor nodes using function call indirections and difference computation," in *USENIX Annual technical conference*, CA, USA, 2009.

[3] J. Jeong and D. Culler, "Incremental network programming for wireless sensors," in *the IEEE International Conference on Sensor and Ad Hoc Communications and Networks (SECON'04)*, 2004.

[4] E. M. Eronu, S. Misra and M. Aibinu, "Reconfiguration Approaches in Wirless Sensor Network: Issues and Chanllenges," in *2nd IEE Conference on Emerging & Sustanable Technologies for Power & ICT in a Developing Society (NIGERCON)*, Owerri, Nigeria, 2013.

[5] A. Tridgeell and P. Mackerras, "The Rsync Algorithm," The Australian National University., Canberra, Australia, 1996.

[6] D. Korn, J. MacDonald, J. Mogul and K. Vo, "The VCDIFF Generic Differencing and Compression Data Format. RFC 3284 (Proposed Standard)," 2002.

[7] J. Black , M. Cochran and T. Highland, "A Study of the MD5 Attacks: Insights and Improvements," 2006.

[8] W. Xiaoyon and Y. Hongbo, "How to Break MD5 and other Hash Functions, Advances in Cryptology," *Lecture Notes in Computer Science*, vol. 3494, pp. 19-35, 2005.

[9] S. Milosh, P. J. Cuijipers and J. J. Lukkien, "Efficient reprogramming of wireless sensor networks using incremental updates and data compression," in *International conference of Pervasive Computing and Communications Workshops(PERCOMWorkshops),*, 2013.

[10] P. R. Krishna, S. Bagchi and P. S. Midkiff, "Zephyr: Efficient Increamental Reprogramming of Sensor Nodes using Function call Indirections and Difference Computation," in *Proceedings of the Annual Technical Conference (USENIX)*, San Diego, CA,USA, , June, 2009.

[11] E. Emmanuel and E. T. Ibrahim, "Contextual information to optimize Wireless Sensor Node's Energy Consumption Rate during Reprogramming Procedures," *International Journal of Computer Application,* volume 143, no 11, 2016.

[12] E. Emmanuel and S. Misra, "Precise Delta Extration Scheme for Reprogramming of Wireless Sensor Nodes," *Nigerian Journal of Technology (NIJOTECH),* vol. 35, no. 1, pp. 144 - 154, 2016.