International Journal of Innovative Research in Advanced Engineering (IJIRAE) ISSN: 2349-2163
Volume 1 Issue 10 (November 2014) www.ijirae.com

Secure Data Aggregation in Wireless Sensor Networks

Abdul Ahad Md[1], Y.Mahesh[2] , A.Uma Sri[3]
[1]*Asst. Professor, Department of ECM, KL University, Vaddeswaram, INDIA,*
[2]*Student, Department of ECM, KL University, Vaddeswaram, INDIA*
[3]*Student, Department of ECM, KL University, Vaddeswaram, INDIA,*

Abstract—the remote sensor system is gathering of sensors imparting to one another in remote medium. The configuration of the remote sensor system puts an imperative part of the information dependability and the outline of a sensor put a critical part in the productive use of restricted assets on it. The sensors have less estimations of assets like memory, battery, sensing. One of the best approach is to utilize the assets as a part of the productive route is from information collection as the estimation of the assets wards on the measure of information changed so to lessen the measure of information and to send the obliged information to the end of the line is called as information total. Security in remote sensor systems put a vital part in securing from remote or dangerous environment region that is inclined to assaults easily. So information accumulation and security are crucial for WSN. Numerous secure collections are proposed. In remote sensor system anyhow, because of asset compelled nature, secure information total likewise requires new methodologies. In this review we are going to think about existing secure information conglomeration convention and their constraints and preferences

Index Terms- base station, sensors, data aggregation, and security in sensor networks, error correction and detection

INTRODUCTION

Wireless sensor network, usually consists of hundreds of thousand numbers of sensors with limited amount of memory, battery life and low powered sensing devices and this type of network is mainly used in the areas of military and civilian and many applications

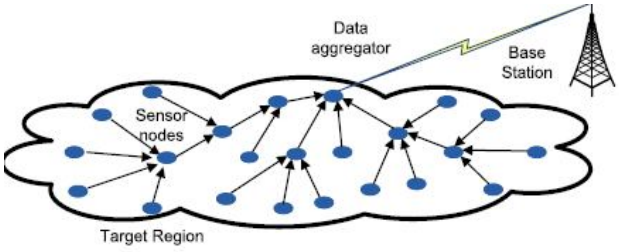
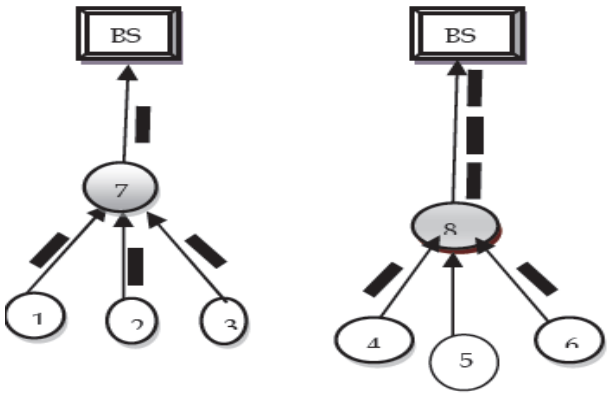


Fig. 1. Data aggregation in a wireless sensor network.

DATA AGGREGATION

Data aggregation is defined as the collection of large data from multiple sensors and send it to the base station for the processing, but the sensors have limited resources so the direct transformation of data to a base station for processing cases

We need new technologies for converting data into more quality information in the intermediate or sensor nodes, which can reduce the number of packets transmitted to the sink so that we can conservation of energy and bandwidth. Thus, data aggregation is the process of reducing the transmitting data. This can be explained with a block diagram



In the above diagram 1,2,3,4,5,6 are the sensors and 7,8 are the aggregators and its task is to collect the data from the sensors and do the summarization on the data and sending the summarized data to the base station for the future processing.
Another unique future of the sensor networks is that each sensor consists processor, instead of sending the raw data to the base station it can do the processing.

Data aggregation based networks:

1. Flat networks
2. Hierarchical networks

Flat networks: In this all the sensors will have the equal power, battery life, time and possess equal task and at first the sink will send the data to the sensors and if the sanded data match's with the sensor data then the data is retransmitted.

Hierarchical network: in the above network there is the problem of high power loss. The data in this aggregation is done by the special nodes so that by those special nodes we can reduce the number of bits to be transmitted.

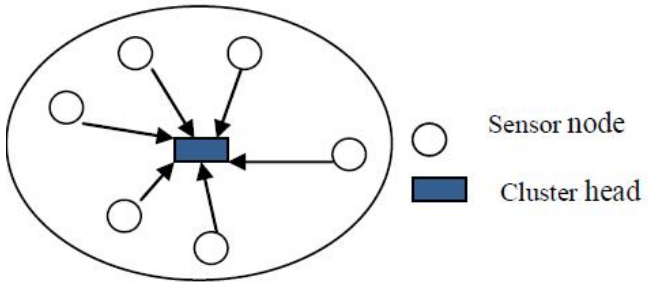
ARCHITECTURES OF DATA AGGREGATION

Based on various applications and requirements there are several existing architectures for data aggregation.

- Centralized
- Decentralized
- Cluster based
- Tree based
- Grid

Centralized architecture

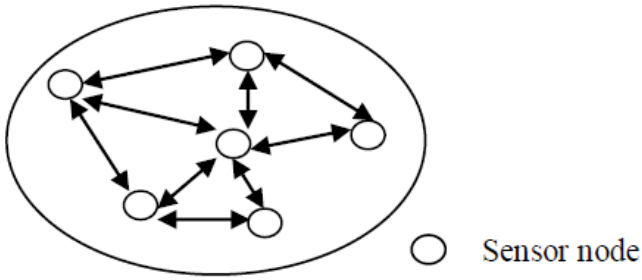
The united structural building is extraordinarily most clear building configuration of the remote sensor framework. In which we can apply data consolidation process. Suggests each sensor center points since a data and transmit to the one central center point, called central processor blend center point. This central processor merges the reports assembled by all sensor centers. In this structural arranging central center point have a commitment of whole framework. The vital point of convergence of this building configuration is it can be smoothly gotten off base report of information which is taken by the entire remote sensor framework. The trouble is that unbending to sensor changes and the workload is concerned at a lone point.



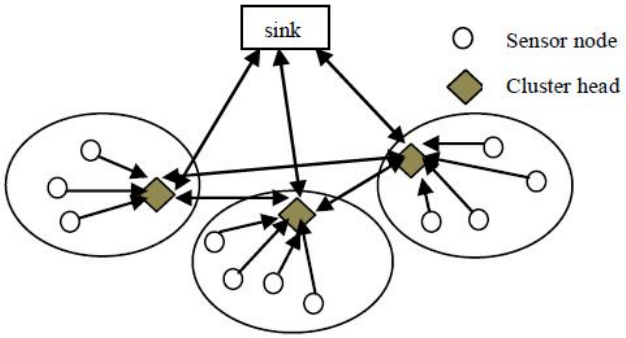
Centralized architecture

Decentralized architecture

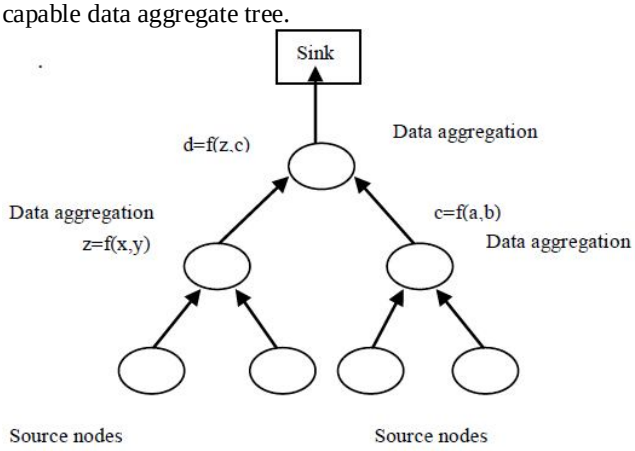
The decentralized building configuration of the remote sensor framework, there is no single focused center point that settles on decisions for the profit of all the sensor centers. Data mixture happens for the most part at each center on the reason of the close-by discernments and the information gained from neighboring centers. In which all sensor center points are connected with each other on the observation .The playing purpose of this structural arranging are flexible and tolerant to the extension or loss of sensing centers or component changes in the framework.



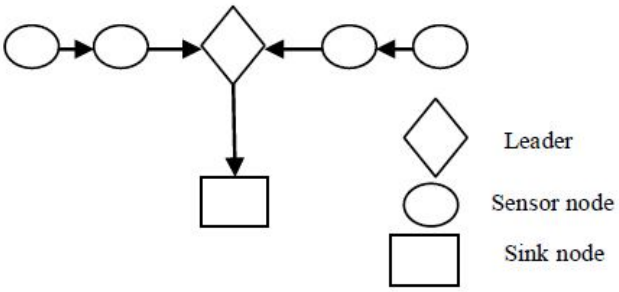
Cluster based architecture:
The wireless sensor framework is resource commitment that is the reason sensor can't direct transmit data to the base station. In which all ordinary sensors can send data group to a gathering head (adjacent aggregator) which adds up to data bundle from all the standard sensors in its cluster and sends the conservative buildup to the base station. With the support of the arrangement we save the imperativeness of the sensors. In essentials propelled sensor frameworks of extensive size, it is inefficient For sensors to transmit the data clearly to the sink. In such circumstances, sensors can transmit data to a close-by aggregator or gathering head which adds up to data from all the sensors in its pack and transmits the concise diagram to the sink. There are a couple of issues included with the strategy of collection in a remote sensor framework. At first issue is, the thing that number of groups should be encircled that could update some execution parameter. Second could be what number of center points should be taken into a lone gathering. Third principal issue is the decision system of gathering head in a cluster.



Tree based architecture:
In the tree-based approach performs add up to by creating an accumulation tree, which could be a base intersection tree, secured at sink and source centers are considered as gets out. Each center point has a watchman center to forward its data. A stream of data starts from surrenders center points over to the sink and in that the combination done by gatekeeper centers. In which all centers are dealt with in appearance of the tree means dynamic, with the aid of moderate center point we can perform data mixture strategy and data transmit leaf center point root center point. One of the essential parts of tree-based frameworks are the improvement of an essentialness capable data aggregate tree.

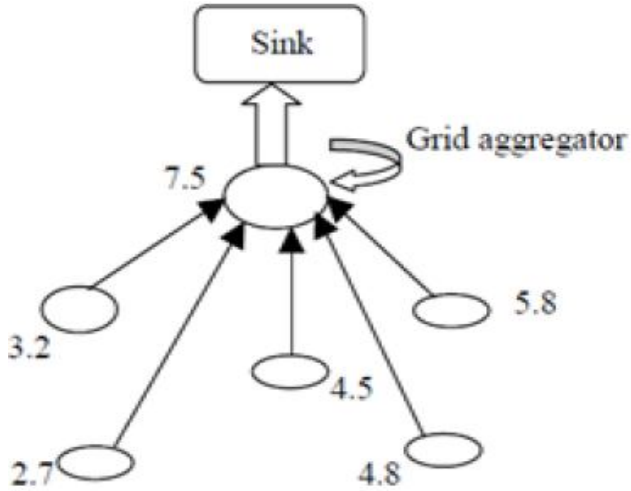


Chain based architecture:
In which each sensor sends data to the closer neighbor. All sensors are sorted out into a straight chain for data accumulation. The center points can structure a chain by using a rapacious count or the sink can pick the chain in a concentrated manner. In the Greedy chain foundation expect that all sensors have far reaching data of the framework. The most remote center point from the sink begins chain game plan and, at every one stage, the closest neighbor of a center is picked as its successor in the chain. In every data collecting round, a center gets data package from one of its neighbors, add up to the data with its own, and sends the sums data group to its other neighbor along the chain. At last, the pioneer center point in the are similar to bundle head sends the amassed data to the base station.



Grid base architecture:

In which a set of sensors is consigned as data aggregators in the settled regions of the sensor framework. The sensors in a skeleton send the data package directly to the aggregator of that network. Hence, the sensors in a network don't relate with each other. In-framework, gathering is similar to grid based data amassing with two noteworthy differentiations; each sensor inside a system talks with its neighboring center point. Any center inside a system can acknowledge the piece of aggregator center in regards to adjusts until the last center kicks the container. This is similar to gathering based data combination in which the group heads are settled. In-framework accumulation, the sensor with the most separating information adds up to the data packages and sends the entwined data to the sink. Each sensor transmits its banner quality to its neighbors. In case the neighbor has higher sign quality, the sender stops transmitting packs. In the wake of getting data packages from all the neighbors, the center point that has the most amazing sign quality transforms into the data aggregator. The in-framework aggregate arrangement is best suitable for circumstances where events are exceedingly bound.



SECURITY IN WIRELESS SENSOR NETWORK

Requirements for Data Aggregation Security

- ***Information Confidentiality****: Guarantees that data substance is never uncovered to any individual who is not approved to get it. It can be separated (in secure information conglomeration plans) into a jump by-bounce premise and an end-to-end premise. In the jump by-bounce premise, any aggregator indicate needs unscramble the got encoded information, apply a total capacity, encode the accumulated information, and send it to the upper aggregator point. This sort of secrecy execution is not viable for the WSN since it obliges additional calculation*
- ***Information Integrity****: Guarantees that the substance of a message has not been adjusted, either anxiously or unintentionally, amid the transmission process. Privacy itself is insufficient since a foe is still ready to change the information in spite of the fact that it knows nothing about it. Assume a safe information collection plan concentrates just on information secrecy. An enemy close to the aggregator point will have the capacity to change the totaled result sent to the base station by including a few parts or controlling the parcel's substance without discovery.*
- ***Information Freshness****: Guarantees that the information are late and that no old messages have been replayed to ensure information conglomeration plans against replay assaults. In this sort of assault, it is insufficient that these plans just concentrate on information privacy and trustworthiness on the grounds that an uninvolved foe has the capacity listen, to try and scrambled messages transmitted between sensor hubs can replay them later on and disturb the information accumulation results.*

- ***Information Availability****: Guarantees that the system is alive and that information is available. It is very proposed in the vicinity of the tradeoff hubs to accomplish system corruption by taking out these terrible hubs. Once an assailant gets into the WSN by bargaining a hub, the assault will influence the system administrations and information accessibility, particularly in those parts of the system where the assault has been propelled.*
- ***Confirmation****: There are two sorts of verification; substance validation, and information confirmation. Substance confirmation permits the recipient to check if the message is sent by the asserted sender or not. Hence, by applying verification in the Wsns, a foe won't have the capacity to partake and infuse information into the system unless it has substantial validation keys.*

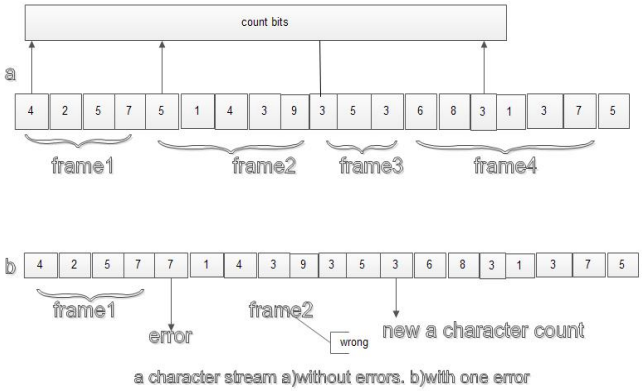
In order to provide more protection to the encrypted data we are adding the concept called framing in the computer network's where the physical layer takes the raw bits from the network layer and transmission of error free data to the destination is very important in the physical layer the encrypted Data is sent to the destination and the number of bit's transmitted may be not equal to the number of bit's received and they may be different values. And the data link layer will detect and correct the errors.

The data link layer will add some check bits for the frame and the frame will be converted into bits in the physical layer and the bits are encrypted and send it to the destination and at the receiver end the recomputed. Breaking the bits stream up into frames is more risky, so to reduce it time gap is inserted between each frame. Networks rarely make any guarantees about timing, so there is a possibility these gaps may be squeezed out other gaps might be inserted during the transmission.

The new methods added for the framing for avoiding the timing gap problem area
1. Adding count bit.
2. Starting and ending characters, with character stuffing.
3. Starting and ending flags, with bit stuffing.
4. Physical layer coding violations.

Adding count bit:

In this method during the transmission of the encrypted data we place the count byte to specify the number of bits that a particular frame is taken. The destination sensor's data link layer sees the number of characters, and takes the how many character a particular frame has taken, the method is explained For four frames of size 4,5,3and 6 characters respectively.

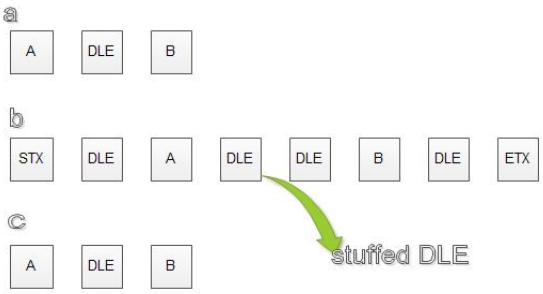


a character stream a)without errors. b)with one error

The problem with this method is the header bit may loss during the transmission. For example, if the character count of 5 in the 2nd frame is changed to 7 during the transmission, then at receiving sensor will think wrong data. so the total data will be Goes wrong, so this method was rarely used anymore.

Starting and ending characters, with character stuffing

In this method will avoids the problem of resynchronization after the an error by having frame start with the ASCII character sequence DLE STX and end with the sequence DEL ETX. So in this way, even though the character The count was loss the sensor will look for the DLE STX or DLE ETX. The problem with this method is when a binary data, such as the object programs or floating-point numbers are being transmitted then it may match with the DLE STX or DLE ETX. So one way to avoid this problem is to insert an ASCII DLE character just before each "accidental" DLE before the data are given to the network layer and this method is called "character stuffing" thus framing DLE STX or DLE ETX can be differentiated by absence or presence of single DLE, the DLE in the data are all ways double. And in the below diagram will represent the data stream before stuffing, after stuffing and after de stuffing.

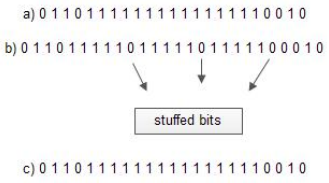


a)data sent by the network layer
b)data after being character stuffing by the data link layer
c)data passed to the network layer on the receiving side

A major problem with this framing method is that it is closely tied to 8-charactres.so another new method called as bit stuffing.

Bit stuffing:

In this method during the transmission of data to the destination sensor whenever the data is encounter's with 5 consecutive 1 bit's then it will place the bit'0' to the next of it.so whenever the receiver encounters with 5 consecutive 1 bits then it will automatically removes the 0 bit.so the pattern is in this form if the user data is 0111111011 then this flag is transmitted as 01111101011 but stored in the receiver sensor as 0111111011 the following fig gives the example of bit stuffing.



Bit stuffing a)original data.
b)the data after bit stuffing
c)the data at the receiver's memory after destuffing

By the bit stuffing method the limits between the two frames can be easily identified. Thus if the losses the finding the flag bit it never traces the stuffed bits.

Physical layer coding violations:

This method is only used in the networks of sensors when there is a redundant data in the physical medium

Error control:

After completing the problem of framing we have another problem called controlling the errors during the transmission of the data from one sensor to the other sensor, the sending sensor must know whether the data is received properly or not without any errors and loss of data and this can be done by sending the response from the receiver so it must contain a reliable delivery of data. One way of providing the reliable delivery is there should be a feedback from the receiver to the sender about the incoming data so that the receiver knows the negative or positive feedback from the receiver and to the corresponding work. If the sender gets the positive feedback from the receiver sensor then it will understand that the data had sent successfully. If it gets the negative feedback from the receiver then it will understand that there is missing of data, and the data must be retransmitted. There is another problem with this is that if the data is completely lost, then the receiving sensor will not give the acknowledge to the receiver because receiving sensor will don't know whether the sender had sent the data or not. so there will be no response from the receiver. One way to solve this problem is placing the timer into the sensor so that when the data is transmitted to the receiver then the timer will start's counting if there is no acknowledge from the receive until the actual time is completed. Then sender get to know that the data had lost and it will resend the data to the receiving sensor. In this there is one more problem is that there is a chance of losing the acknowledgment also if it happened then the sender will resend the data to the receiver so that there will be chances of redundant data at the receiver. One way to avoid this problem is that the sending sensor had to add the sequence number to sending data so that the receiver will distinguish between the retransmitted data and the original data. The total methods are ultimately to send the data without any loss and without any redundant data.

Flow control:

Another designing issue is that if the sender sends the data faster than the receiver then there will cause for the data loss.it will happen many if the sending sensor is faster than the receiving sensor.so this can be prevented by sending the acknowledgment signal to the sender that it had completely received the data or if the receivers buffer is full so that it will send the next data or stop until the positive acknowledgment is received from the receiver.

ERROR DETECTION AND CORRECTION

Network designers had first designed the sensor networks in such a way that the receiver will first have the capacity to identify the error, but it doesn't know there the error had occurred and it will simply send the acknowledgment single for retransmission later day they had designed the network in such a way that the receiver had the capable of detecting and correcting of the errors. So that lot of time taken will be reduced.

One of the method for detecting the errors in the received data is based on parity checking in this the sender will add a parity bit to the message bits and there 2 types of parity adding methods they are

1 even parity

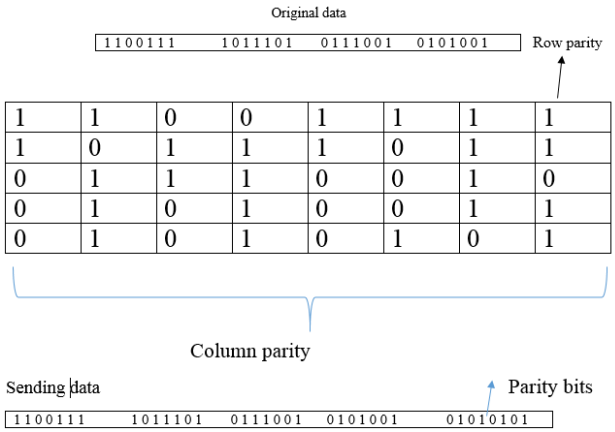
2 odd parity

Example: for even parity if the data is 10101 in this the number of 1's are 3 so we add one more 1 to get the even parity.

Similarly for the odd parity if the data is 1010 in this the number of 1's are 2 so we add one more 1 to get odd parity.

2 D parity check:

In this method the receiving sensor can detect the block of errors, In the at the sending side the data to be send is arranged in the form of table and it can be explained with the help of example



Transmission of double data bits is also sometimes useful for error detecting because if *xx* is sending and receiving sensor get the Different bits, then it can know that error had happened, but it cannot correct the error it has to send the acknowledgment to the sender for retransmission of data.

To understand what error handling is first of all we have to understand what the error is. Normally the sending data *and* consists

M data bits and *r* check bits so total length of *n* be (*m*+*r*).and then bits often referred to be codeword.it can be explained, taking an example of 2 code words 1001100110 and 1011101101 in this there 3 bits are different and it can be known by ding the EX-OR operation to the two code words and finding the number of 1 is the output will gives number of error bits and their corresponding positions. And the number of 1's in the output is called as the HAMMING DISTANCE d.

The error detection and error correction depends of the code word depends hamming distance.to detect d errors, we need a d+1 distance code.it can be explained with an example consider a code in which a single parity bit is added to the data. The parity bits are depending on the number of 1's in the data either it is even or odd number of 1's.for example, when 10110101 is sent in even parity by adding a parity bit at the end. Then it will become 101101011, whereas 10110001 becomes 101100010 with even parity. A code with a single parity bit will have 2 distance because a single bit error will produce the wrong parity. And it can be used to detect the errors.

Example

0000000000 0000011111 1111100000 1111111111

The code has a distance of 5 that means the receiving sensor has the capable of correcting 2 errors. If it gets the data as 0000000111 then the receiving sensor will think that the data must be 0000011111 if the data gets has triple errors changes

0000000000 into 0000000111 then the receiver cannot correct properly.

Hamming code:

Hamming code is to detect one bit error in the data and it uses the extra parity bits for identifying the error.

Steps for converting the data into the hamming code

Step 1: place the data in the 2^m positions.

Step2: the remaining positions like 1,3,5,7,9…..for the parity bits.

Step3: each parity is calculated based on the even or odd parity.

Step4: for the first position of parity bit we have to check the parity for the bits in the (1,3,5,7,9,11,13,15,.. etc) positions.

Step5: for the second position of parity bit we have to check the parity for the bits in the (2,3,6,7,10,11,14,15,.. etc) positions.

Step6: for the third position of parity bit we have to check the parity for the bits in the (4,5,6,7,12,13,14,15,20etc) positions.

Step7: place the parity bit as 1 if it get odd number of 1's and place '0' is it get even number of 1's.

Here is an example:

A byte of data: 10011010

Create the data word, leaving spaces for the parity bits: _ _ 1 _ 0 0 1 _ 1 0 1 0

Calculate the parity for each parity bit (a ? represents the bit position being set):

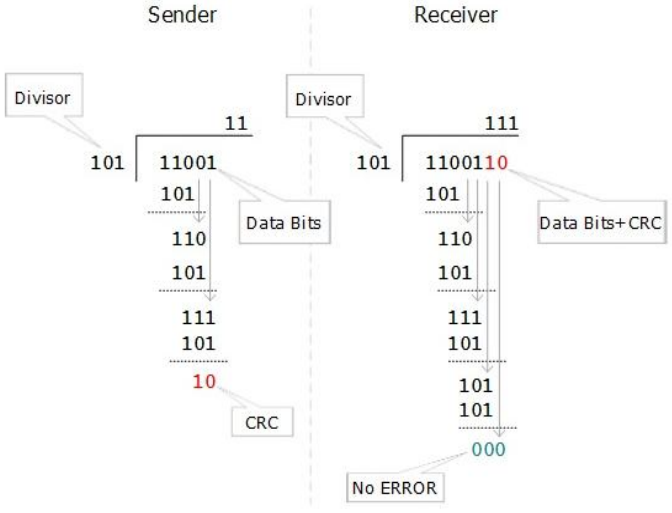
- Position 1 checks bits 1,3,5,7,9,11:
  ? _ 1 _ 0 0 1 _ 1 0 1 0. Even parity so set position 1 to a 0: 0 _ 1 _ 0 0 1 _ 1 0 1 0
- Position 2 checks bits 2,3,6,7,10,11:
  0 ? 1 _ 0 0 1 _ 1 0 1 0. Odd parity so set position 2 to a 1: 0 1 1 _ 0 0 1 _ 1 0 1 0
- Position 4 checks bits 4,5,6,7,12:
  0 1 1 ? 0 0 1 _ 1 0 1 0. Odd parity so set position 4 to a 1: 0 1 1 1 0 0 1 _ 1 0 1 0
- Position 8 checks bits 8,9,10,11,12:
  0 1 1 1 0 0 1 ? 1 0 1 0. Even parity so set position 8 to a 0: 0 1 1 1 0 0 1 0 1 0 1 0
- Code word: 011100101010.

Detecting and correcting the bad bits:

In the above example instead of 0 1 1 1 0 0 1 0 1 0 1 0 we the receiving sensor gets the data as 0 1 1 1 0 0 1 0 1 1 1 0 then receiver has the capability of finding which bit is wrong and fixing the error bit. The method followed by it is, it will verity the all the parity bits by checking the corresponding positions so that by doing this it gets parity bits 2 and 8 are incorrect and it will add the 2 and 8, 2+8=10 so it identifies that 10th bits was the wrong bit.

Cyclic Redundancy Check (CRC):

In this method the encrypted data to be send is performed with binary division. And the divisor is generated by the polynomials and the sensor will performs the division and the remainder is taken from this division. Before sending the actual data to the receiving sensor the remainder bits are added at the end of the actual bits to be transmitted. In this the remainder and the actual data is called as the code word. At the receiving end the sensor will performs the division operation on the received bits with the devisor if the remainder is zero then only the receiver will accept the data otherwise the acknowledgement is sent to the sender for the retransmission of the data.



Error correction:

In the digital world error correction can be done in two ways

1. *Back ground error correction*: when the receiver detect the errors then it will request the sender to retransmit the data.
2. *Froward error correction:* when the receiver get the error bits the, it will correct the error bits with help of the sender for the retransmission of the data. And it can be done with the help of redundant bits as the parity bits *r*. and the number of parity bits to be add is depends on the following condition

$$2^r >= m+r+1$$

Where r is number of parity bits

M is number of actual bits

CONCLUSION

We have displayed an exact study on different parts of remote sensor systems and diverse architectures of information conglomeration every one of them concentrate on improving paramount execution measures, for example, system lifetime, information idleness, information precision and vitality utilization. Furthermore, we likewise portrayed about the portion of the security issues recording to the scrambled information furthermore for mistake location and blunder remedy strategies in the remote sensing systems.

FUTURESCOPE

In future we are going to principally think diverse secure techniques for information sending starting with one sensor then onto the next sensor and we have a few thoughts like "partition and sending with distinctive game plans" and we are going execute them in future.

REFERENCES

[1] Sushruta Mishra, Hiren Thakkar: Features of WSN and Data Aggregation techniques in WSN: A Survey: International Journal of Engineering and Innovative Technology (IJEIT) Volume 1, Issue 4, April 2012.

[2] Suat Ozdemir a,*, Yang Xiao b:Secure data aggregation in wireless sensor networks: A comprehensive overview: journal homepage: www.elsevier.com/locate/comnet.

[3] Nandini. S. Patil, Prof. P. R. Patil: Data Aggregation in Wireless Sensor Network: 2010 IEEE International Conference on Computational Intelligence and Computing Research

[4] Hani Alzaid Ernest Foo Juan Gonzalez Nieto: Secure Data Aggregation in Wireless Sensor Network: a survey.

[5] Kiran Maraiya, Kamal Kant, Nitin Gupta: Wireless Sensor Network: A Review on Data Aggregation: International Journal of Scientific & Engineering Research Volume 2, Issue 4, April -2011 1 ISSN 2229-5518.

[6] Priyanka K. Shah and Kajal V. Shukla: Secure Data aggregation Issues in Wireless Sensor Network: A Survey: journal of information and communication technologies, volume 2, issue 1, january 2012.

[7] Li Qun Zhuang, Jing Bing Zhang, Dan Hong Zhang and Yi Zhi Zhao: Data Management for Wireless Sensor Networks: Research Issues and Challenges: 2005 Intemational Conference on Control and Automation (ICCA2005) June 27-29, 2005, Budapest, Hungary.

[8] Anindita Ray, Debashis De:Data Aggregation Techniques in Wireless Sensor Network: A Survey: International Journal of Engineering Innovation & Research Volume 1, Issue 2, ISSN: 2277 – 5668

