Embedded System for Mine Process Monitoring in a Network Constrained Environment Using Wireless Communication Bridge

Himansu Sekhar Pradhan[a],Sampad Bhusan Mohanty[b],Santosh Madhukar Yerme[a],Paresh Govind Kale[b], Debiprasad Priyabrata Acharya[a]

[a] Dept. of Electronics and Communication Engineering
[b] Dept. of Electrical Engineering
National Institute of Technology, Rourkela, India

***Abstract*— An efficient and robust implementation of a vehicle monitoring system in places where a Global Navigation Satellite System or GNSS (e.g. GPS, GLONASS, GALILEO) signal is available but Cellular Network Signal is unavailable/unreliable for making a GPRS connection like in an open cast mine is presented. A ZigBee Wireless Network is deployed at the location of interest, i.e. the open cast mine, with the ZigBee Network Coordinator attached to a Single Board Computer(SBC) which is placed at a remote place where it has uninterrupted Internet access either via LAN, Wi-Fi or Cellular Network (GPRS). The location of SBC is chosen such that the region devoid of cellular network falls within the ZigBee Coordinator's range. Other ZigBee nodes are attached to the vehicle's Tracking Equipment as ZigBee Routers. Whenever a vehicle's tracking equipment fails to make a GPRS connection, it sends its location via the ZigBee network to the coordinator which acts as a gateway and uploads the received location data to the tracking server. ZigBee Network Watchdog Timer are configured such as to allow automatic rejoining of ZigBee nodes when vehicles leave and enter the ZigBee Network Range. Vehicle tracking equipment and the Gateway Coordinating Equipment are implemented using hardware and open source software platforms and libraries such as Arduino, Raspberry-Pi and Linux.**

***Keywords*— Embedded Systems, Wireless Communication, Wireless Sensor Networks, Vehicle Tracking, Internet Of Things, ZigBee**

I. Introduction

All Vehicle Tracking Systems [1] use Global Satellite Navigation System (GNSS) to fetch the current location of the vehicle. Global Positioning System (GPS) is the most widely used GNSS today. Vehicle tracking relies on cellular network and hence tracking is possible in all the locations that fall under the coverage of cellular network. But there exist places where either there is no cellular network coverage or the signal strength is too weak for GPRS connection but still there is a requirement for tracking in the area. The tracking/monitoring in such no network area might be very crucial in terms of safety and productivity. One such example is an Open Cast Mine. Open cast mines have deep excavations consisting of different benches. Even in a case where a cell phone tower is located very near to the open cast mine, the network signal degrades very quickly while moving down the benches because of shading of lower benches by the upper ones. A representative solution to this problem is presented in this paper.

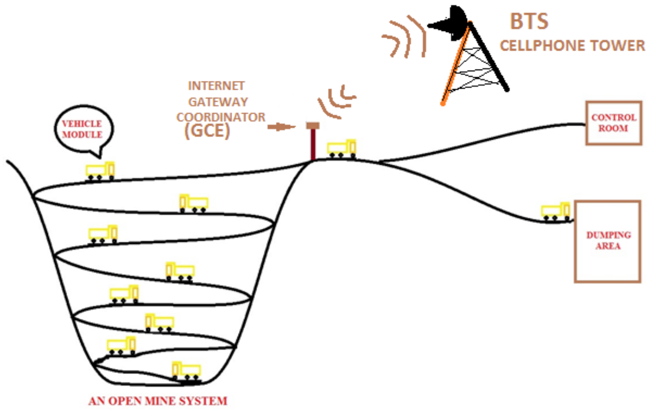

Figure 1: Vehicles operating in an Open Cast Mine showing the proposed solution.

II. The Solution

A. Need for a Local Wireless Network

A tracking equipment contains a modem which connects to cellphone network to establish GPRS connection for sending position data acquired from the GNSS (GPS in our context and henceforth) receiver. In locations where the cellphone network signal is unavailable or weak, the modem is unable to establish a GPRS connection and hence fails to send the most recent position data acquired. The controller present in the tracking equipment might store the position data and send it latter when the cellular network is available the next time. Although this method allows us to track the vehicle very closely as not even a single position data point received by the GPS receiver is missed, it doesn't allow a near real-time operation as it cannot be said for how long a cellular network might not be available to the tracking equipment.

To solve this problem a local wireless network is created which is low on both cost and energy. ZigBee [2] Wireless Communication Standard is selected for implementing this local network as it meets our following requirements – a) reliable and robust b) energy efficient c) cost effective d) advanced routing and self-healing networking capabilities e) a global standard. There are many vendors who provide ZigBee wireless modules but the DIGI International's[3] XBee Series 2 modules are selected because of wide availability of the product, larger user base and ease of implementation as it offers an UART gateway for the ZigBee which makes it compatible with almost every processor/controller available.

B. The ZigBee Local Wireless Network

The ZigBee protocol builds on the top of IEEE 802.15.4 wireless protocol. IEEE 802.15.4 defines the physical (PHY) and the media access layers (MAC) of the personal area low power wireless network [4]. While 802.15.4 itself doesn't have mesh networking and multi-hop capabilities, the ZigBee's Network (NWK) layer implements them. Additionally the ZigBee's upper layers also improve the reliability of packet delivery making it a very reliable network. A ZigBee network consists of exactly one Network Coordinator, Routers and End Devices where each is called a node. Coordinator is responsible for starting, building up and maintenance of the network. A Router or End Device may join the Coordinator or another Router as a child. The Routers are devices which can relay information from a source node to a destination node which are not in each other's direct radio range. The End Devices are used either as interfaces to sensors and/or actuators. Routers and Coordinator can also act as interfaces to sensors and actuators but in addition to that they can be parents to other nodes whereas End Devices cannot be a parent to other nodes. Building Wireless Sensor Networks [5] by Robert Faludi and ZigBee wireless networking [6] by D. Gislason were great resources for learning about Digi's XBee Modules and ZigBee Protocol during our research.

A ZigBee Network is designed through which the Tracking Equipment present on the vehicle can send its location data to a nearby Internet Access Point whenever a GPRS connection is not possible. This ZigBee network consists of a Coordinator and several Routers. A Router is installed into each Tracking Equipment. The Coordinator is installed at the nearby Internet Access Point from where it can listen to any Router (sitting on a Vehicle) sending its location data. This configuration of the ZigBee network helps to increase the coverage as each vehicle contains a Router which can relay for another vehicle trying to send its location data to the Coordinator if the same is not in direct radio range of the vehicle. When Coordinator receives a location data, it is passed to a Single Board Computer (SBC) that is connected to the Internet through the Access Point. This SBC contained in the Gateway Coordinating Equipment (or GCE), runs a program which immediately stores the received coordinates in a local database along with the details of the vehicle from which it was sent. The coordinator may receive location data from many vehicles simultaneously as the ZigBee takes care of all collisions and makes the message delivery process more reliable. Another program in the SBC running parallel as a thread checks the database for any stored location data from vehicles, if found, immediately uploads the data to the Tracking Server via the Internet. Hence the tracking system runs in near real-time.

III. Implementation

The ZigBee network is implemented using the Digi's XBee Series 2 modules each of which contain and run a full ZigBee protocol stack. XBee modules can be flashed as ZigBee Coordinator or ZigBee Router or ZigBee End Device. Again each of them can interact with a host (a microprocessor which controls it) either in a transparent mode or in API mode. The transparent mode is the simplest mode of operation. While the API mode is complex, it offers many options and controls for taking full advantage of the ZigBee protocol as it exposes ZigBee's powerful features to the host. So API mode is the chosen mode of operation for all our XBee nodes. In API mode all the interaction and transactions with XBee are done over the UART in a packet frame data structure. All API frames have a generic structure as shown below in figure 2.The start byte is always 0x7E followed by 2 bytes representing the length of the payload starting from the 4^{th} byte to the last byte i.e. checksum but excluding it. The 4^{th} byte specifies the Frame Type. Some of the frame types are - AT Command immediate(0x08),AT Command Queued (0x09), AT Response (0x88), Remote AT Command (0x17), Remote AT Response (0x97), Transmit Request (0x10), Transmit Status (0x8B), RX Receive (0x90), RX I/O Data Received (0x92), Node Identification Indicator (0x95), Modem Status (0x8A).

Byte Number :	0	1	2	3		n	n+1
Byte Contents :	0x7E	MSB	LSB	Frame Data			Chksm
				API cmd	cmdData		

Figure 2 : XBee API generic frame packet format

By using API mode in our implementation, it is possible to send message to different nodes by setting the address of the destination node on the fly which is not possible in transparent mode. Also in API mode a Transmit Status is received as an acknowledgment which tells us whether the message reached the destination successfully. There are two hardware modules for complete system implementation viz. the Tracking Equipment (TE) and the Gateway Coordinating Equipment (GCE).

A. Tracking Equipment

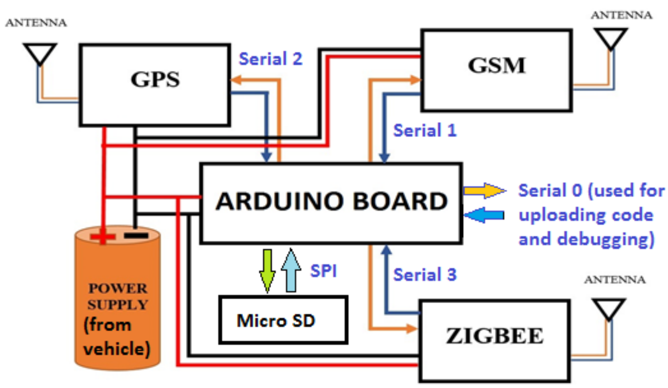

Figure 3 : Tracking Equipment (TE)

Normally a tracking equipment contains a GPS receiver, a Modem and a Processor/Controller. In our implementation, one more component is added – the XBee module which will be a part of our ZigBee network. Whenever it is not possible to send the location data through the Modem due to unavailable cellular network or low signal strength, the Controller will try to send the data through the ZigBee Network to the Gateway coordinating Equipment. The program flow implemented in the controller i.e. Arduino Mega is as follows -

1) *Initialize GPS receiver, XBee and GSM Modem*
2) *Get current position of vehicle from the GPS receiver*
3) *Attempt GPRS connection to send current position*
4) *If successfully sent, go to step 8*
5) *Else attempt to send position data via XBee*
6) *If Transmit Status successful delivery, go to step 2*
7) *Else store location data in SD card and go to step 2*
8) *If location data present in SD card, send via GPRS*
9) *Go to step 2*

B. Gateway Coordinating Equipment

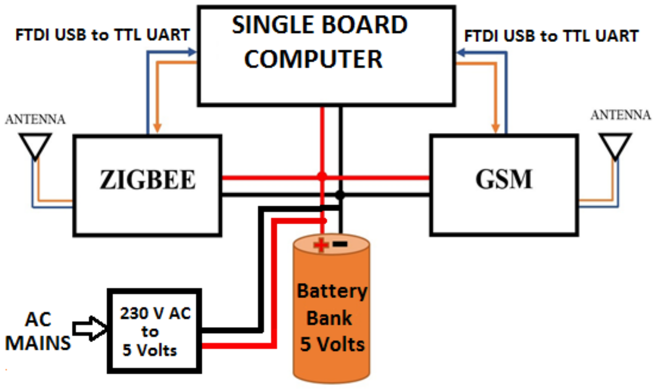

Figure 4 : Gateway Coordinating Equipment (GCE)

The Gateway Coordinating Equipment (GCE) is the device that sits at the nearby Internet Access point and listens to TE location data over the ZigBee network. The GCE must have a reliable Internet connection all the time. The internet connection might be over anything – Ethernet, Wi-Fi, GPRS, Satellite, etc. In our implementation in the mines, there were no nearby Ethernet or Wi-Fi connection where we can place the GCE such that all the mine benches are covered within the range of the ZigBee coordinator present in GCE. But there was a good Cellular signal strength, sufficient for GPRS connections, at the uppermost mine bench. The GCE consists of a Single Board Computer (SBC) – the Raspberry Pi 2 which runs Raspbian – a Debian Operating System optimized for Raspberry Pi hardware from an SD Card. Both Raspberry Pi and the Raspbian are Open Source. The GCE also contains an XBee module programmed as Coordinator for our ZigBee Network. For connecting to Internet via GPRS, GCE contains a GSM modem which is the same as the one installed in the TE. The power supply to the GCE contains a battery bank which can run the GCE through 4 hours of power loss.

The program that handles all the tasks of the GCE is written in Python. The main program on startup creates a thread that listens to the Serial port connected to XBee module indefinitely and if any location data is received form a vehicle, it immediately stores the data in a local database. The database is implemented using SQLite because of its simplicity and portability. The main thread of the program monitors the local database for new entries and if found sends the data over using SIM900A GSM Modem via GPRS. This mechanism allows for a near real-time operation of the vehicle monitoring system. After sending data from new entries, the entries are marked as old by writing a flag into the database. The main thread is -

1) *Initialize XBee ,GSM Modem & database*
2) *Start thread to listen to XBee Serial Port indefinitely*
3) *Check if new entry/entries in database*
4) *If new entry found ,send data over GPRS via SIM900*
5) *Go to step 3*

The daemon Thread listens to XBee Serial Port

1) *Listen to XBee Serial Port for API frames*
2) *If RX Receive API Frame arrives, store in database*
3) *Go to Step 1*

IV. Working

The tracking equipment on a vehicle sends the location data to the GCE only when it isn't able to send the same through its own SIM900A modem over GPRS because of poor Cellular Signal. The location data is sent in an XBee Transmit Request API frame with the 64 bit Destination MAC address set to zero which is a special address for the Coordinator of the ZigBee Network, i.e. the XBee module present in the GCE. The payload of the API frame, i.e. the Location Data also contains the time from the GPS at which the position was read. The XBee of the GCE receives the data and forwards it through the serial port to Raspberry Pi as a RX receive API frame.

Receive (RX) API Frame

Byte Sequence	0	1	2	3	4 5 6 7 8 9 10 11
Byte Content	0x7E	MSB	LSB	0x90	Source 64 bit MAC Address

12 13	14	15 16 17 ... n	n+1
Source 16 bit Network Address	Receive Options	RF data (payload)	Checksum

Figure 5 : XBee Receive API Frame

The RX Receive API frame contains the data sent along with the MAC address of the source XBee i.e. the XBee present on the tracking equipment that sent the data. A mapping between the MAC address of XBee and the vehicle Identifier (e.g. vehicle name or the vehicle registration number) is already added to the database either manually or automatically using a network discovery feature of ZigBee. This allows us to identify which vehicle sent the data to GCE. The data is immediately inserted to database by the thread listening to XBee serial port.

Transmit (TX) Request API Frame

Byte Sequence	0	1	2	3	4	5 6 7 8 9 10 11 12
Byte Content	0x7E	MSB	LSB	0x10	Frame ID	Destination 64 bit MAC Address

13 14	15	16	17 18 ... n	n+1
Destination 16 bit Network Address	Broadcast Radius	Options	RF data (payload)	Checksum

Figure 6 : XBee Transmit Request API Frame

The main thread, on finding a new entry in the database immediately uploads the data to the server via GPRS using GSM Modem and marks the status of the entry in the database as 'uploaded'. The tracking equipment sends location data over the ZigBee Network by issuing a Transmit Request Frame to its XBee and then waits for the Transmit Status Frame from it. In a case the data is unable to reach the Coordinator XBee at GCE, the tracking equipment receives a Transmit Status API Frame containing a field Delivery Status set to a non-zero value (zero means successful delivery) and hence stores the location data in a SD card in a text log file. If the ZigBee Network is carefully designed to cover the area where the cellular signal is weak or unavailable then there will be very rare occasions when the tracking equipment is unable to send location data either through GPRS and the ZigBee.

Transmit Status API Frame

Byte Sequence	0	1	2	3	4	5	6
Byte Content	0x7E	MSB	LSB	0x8B	Frame ID	Destination 16 bit Network Address	

7	8	9	10
Transmit Retry Count	Delivery Status	Discovery Status	Checksum

Figure 7: XBee Transmit Status API Frame

The location data is stored in a SD card in case of the above rare occasion and this data is uploaded to the server once the tracking equipment is able to make a GPRS connection when cellular network is available again later. This stored data in SD card is not sent over the ZigBee Network deliberately to prevent overwhelming the ZigBee Network. Being a many (TE) to one (GCE) communication most of the time, the Aggregate Routing Notification (AR) parameter on XBee coordinator is set to 3 causing it to broadcast Route Record Indicator packets every 30 seconds enabling others to trace the most efficient path back to the coordinator. The Network Watchdog Timer of the XBee nodes on vehicles are set to 1 minute using the AT command NW. The XBee nodes on the vehicles search for the network coordinator every 1 minute and if cannot communicate with it within 3 NW periods, they leave the current network and search for a new ZigBee network to join. This allows for automatic rejoining to ZigBee coordinator when trucks move in and out of the coordinator's range.

V. Performance and Assessment

The system was first deployed at National Institute of Technology, Rourkela for testing purposes. The GCE was placed at a location having good cellular network strength on the top of a tall academic building. Tracking equipment were attached to two vehicles and were made to travel by different paths. During the testing the vehicles were made to enter and leave ZigBee coordinator's radio range many times. Also the cellular signal strength available to the GSM modem (SIM900A) present in the tracking equipment was purposefully attenuated by removing its antenna. The following cases were tested when the vehicle was present in the range of –

1) *Both cellular and ZigBee network*
2) *Only cellular network*
3) *Only ZigBee network*
    a) *Long after entering ZigBee Network (>3 min)*
    b) *Shortly after entering ZigBee Network(<3 min)*
4) *No network*
5) *Entering Cellular Network from no network case*

In the 1st case the tracking module sent all of the location data via GPRS with a few occasional failures due to cellular network congestion which were sent over ZigBee network to the GCE. In the 2nd case, when these occasional failures occurred, the location data was transmitted by the XBee router of the vehicle equipment but as no acknowledgement was received from the XBee coordinator, the TX Status API frame informed the failure and the GPS data was saved to the micro SD card. In the case 3a, all the GPS data were sent to the GCE after failing to send the same over GPRS. We expected a few failures here but to our surprise there weren't any. We concluded that this was because we were well within ZigBee coordinators range and because there were only two tracking equipment, there wasn't any congestion in the ZigBee network. In the case 3b, the vehicle which had just entered the ZigBee network range tried to send the GPS data over ZigBee but failed because the router in the tracking equipment hadn't yet joined the ZigBee network. Hence the GPS data were instead stored in the micro SD card. The routers joined the ZigBee network within 3 minutes of entering coordinator range and thereafter could send data successfully to the GCE. In case 4, all the GPS data were stored to the micro SD as expected. In case 5 when the vehicle recovered from a no network into a cellular network area, it was able to send GPS data over GPRS again and it also tried to send the stored data in the micro SD in bulks of 5 GPS data points per each upload (HTTP) request.

VI. Conclusion

A field deployable embedded system prototype is developed for monitoring the mine process using the wireless communication networks. ZigBee standard is successfully deployed to bridge the wireless protocol in the absence of standard GPRS network. The system is tested rigorously and found to be performing up to the expectations during the testing. The application of such kind of systems can be extended to more geographically constrained regions. Out of many improvements that can be made to the present system, reducing the size of the hardware is a priority. Another priority is extending the range of the local short range radio network by including support for other ZigBee like networks. The applications are expected to be easily extended to similar engineering situations and problems other than the one presented in this paper.

References

[1] S. Lee and G. Tewolde, 'Design and implementation of vehicle tracking system using GPS/GSM/GPRS technology and smartphone application', IEEE Computer Society, 2014. Zigbee.org, ' The ZigBee Alliance | Control your World', 2015. [Online]. Available: http://www.zigbee.org/. [Accessed: 15- Aug- 2015].

[2] Digi.com, 'Your M2M Expert - Digi International', 2015. [Online]. Available: http://www.digi.com/. [Accessed: 15- Aug- 2015].

[3] P. Baronti, P. Pillai, V. Chook, S. Chessa, A. Gotta and Y. Hu, 'Wireless sensor networks: A survey on the state of the art and the 802.15.4 and ZigBee standards', Computer Communications, vol. 30, no. 7, pp. 1655-1695, 2007.

[4] R. Faludi, Building wireless sensor networks. Beijing: O'Reilly, 2011.

[5] D. Gislason, Zigbee wireless networking. Oxford: Newnes, 2008.