ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Implementation of Long Range Smart Tracking System Using Internet of Things

Naganjaneyulu N
M.Tech,
St Peters Engineering College,
Hyderabad, India.

Prashanthi E
Assistant Professor,
St Peters Engineering College,
Hyderabad, India.

P.Madhuri
Assistant Professor,
St Peters Engineering College,
Hyderabad, India.

Abstract:

The current Wireless Network topologies are based on Bluetooth, WiFi, 2G, 3G, 4G Protocol all these features has advantages and their drawbacks. For this Internet of things came into wireless networks. The things which are controlled by Internet. Previous wireless modules gets range with increase in power. Without increasing the power, getting the Range these long range wireless modules are useful. We will get the tacking information for soldier and monitor their health condition their ammunition details we will send to the base Station. This project is about a new era of computing technology that is called the Internet of Things (IoT). Machine to machine, machine to infrastructure, machine to environment, the Internet of Everything, the Internet of Intelligent Things, intelligent systems—it's happening, and its potential is huge.

We see the IoT as billions of smart, connected "things" (a sort of "universal global neural network" in the cloud) that will encompass every aspect of our lives, and its foundation is the intelligence that embedded processing provides. The IoT is comprised of smart machines interacting and communicating with other machines, objects, environments and infrastructures. As a result, huge volumes of data are being generated, and that data is being processed into useful actions that can "command and control" things to make our lives much easier and safer and to reduce our impact on the environment. The creativity of this new era is boundless, with amazing potential to improve our lives. This project is an extensive reference to the possibilities, utility, applications and the evolution of the Internet of Things.

Keywords:

Smart tracking system, dspic30f4011, Lora, IOT, internet of things, future scope in India, potential possibilities, threat.

I. EXISTING METHOD:

The current Wireless Network topologies are based on Bluetooth, WiFi, 2G, 3G, 4G Protocol all these features has advantages and their drawbacks. For this Internet of things came into wireless networks. The things which are controlled by Internet. Previous wireless modules gets range with increase in power.

II. PROPOSED SYSTEM:

The project is for tracking of Wireless Soldiers Network (WSN) which are 2 Km apart from each other using the Base Station. The tracking is through Long range WAN (using spread spectrum modulation). As such, it is clear that the IoT will consist of a very large number of devices being connected to the Internet. It is an active move to accommodate new and emerging technological innovation.

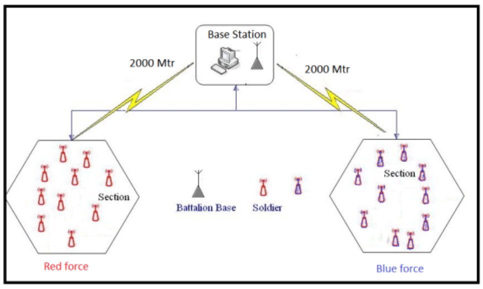


Fig 1: Wireless Solider Network

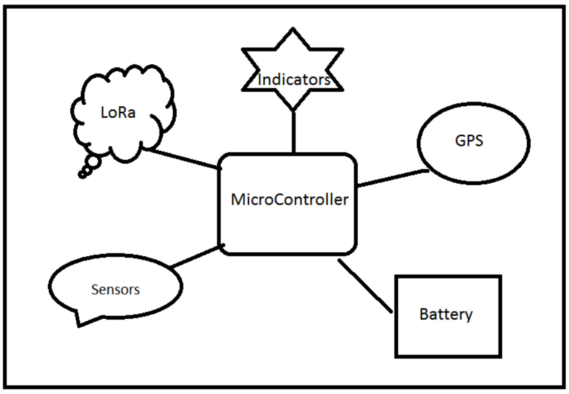


Fig 2: Transmitter Block Diagram

Wireless Soldier Network:

A wireless Soldier network (WSN) of spatially distributed autonomous tracking to monitor Location Health conditions, such as Health, Ammunition, Battery Status etc. and to cooperatively pass their data through the network to a main location. The more modern networks are bi-directional, also enabling control of Soldier activity. The development of wireless Soldier networks was motivated by military applications such as battlefield surveillance; today such networks are used in many industrial and consumer applications, such as industrial process monitoring and control, machine health monitoring.

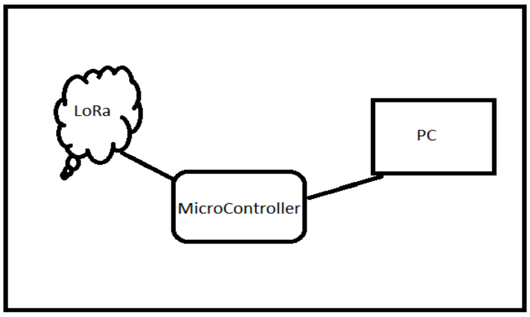


Fig 3 Receiver Block Diagram

III. HARDWARE IMPLEMENTATION:

Core Overview:

The core has a 24-bit instruction word. The Program Counter (PC) is 23 bits wide with the Least Significant (LS) bit always clear (see Section 3.1), and the Most Significant (MS) bit is ignored during normal program execution, except for certain specialized instructions. Thus, the PC can address up to 4M instruction words of user program space. An instruction pre-fetch mechanism is used to help maintain throughput. Program loop constructs, free from loop count management overhead, are supported using the DO and REPEAT instructions, both of which are interruptible at any point. The working register array consists of 16x16-bit registers, each of which can act as data, address or offset registers. One working register (W15) operates as a software stack pointer for interrupts and calls. The data space is 64 Kbytes (32K words) and is split into two blocks, referred to as X and Y data memory. Each block has its own independent Address Generation Unit (AGU). Most instructions operate solely through the X memory AGU, which provides the appearance of a single unified data space. The Multiply-Accumulate (MAC) class of dual source DSP instructions operate through both the X and Y AGUs, splitting the data address space into two parts. The X and Y data space boundary is device specific and cannot be altered by the user. Each data word consists of 2 bytes, and most instructions can address data either as words or bytes. There are two methods of accessing data stored in program memory:

• The upper 32 Kbytes of data space memory can be mapped into the lower half (user space) of program space at any 16K program word boundary, defined by the 8-bit Program Space Visibility Page (PSVPAG) register. This lets any instruction access program space as if it were data space, with a limitation that the access requires an additional cycle. Moreover, only the lower 16 bits of each instruction word can be accessed using this method.

• SWW Linear indirect access of 32K word pages within program space is also possible using any working register, via table read and write instructions. Table read and write instructions can be used to access all 24 bits of an instruction word. Overhead-free circular buffers (modulo addressing) are supported in both X and Y address spaces.

This is primarily intended to remove the loop overhead for DSP algorithms. The X AGU also supports bit-reversed addressing on destination effective addresses, to greatly simplify input or output data reordering for radix-2 FFT algorithms. Refer to Section 4.0 for details on modulo and bit-reversed addressing. The core supports Inherent (no operand), Relative, Literal, Memory Direct, Register Direct, Register Indirect, Register Offset and Literal Offset Addressing modes. Instructions are associated with predefined Addressing modes, depending upon their functional requirements. For most instructions, the core is capable of executing a data (or program data) memory read, a working register (data) read, a data memory write and a program (instruction) memory read per instruction cycle.

As a result, 3-operand instructions are supported, allowing C = A+B operations to be executed in a single cycle. A DSP engine has been included to significantly enhance the core arithmetic capability and throughput. It features a high speed 17-bit by 17-bit multiplier, a 40-bit ALU, two 40-bit saturating accumulators and a 40-bit bi-directional barrel shifter. Data in the accumulator or any working register can be shifted up to 16 bits right or 16 bits left in a single cycle. The DSP instructions operate seamlessly with all other instructions and have been designed for optimal real-time performance. The MAC class of instructions can concurrently fetch two data operands from memory, while multiplying two W registers. To enable this concurrent fetching of data operands, the data space has been split for these instructions and linear for all others.

This has been achieved in a transparent and flexible manner, by dedicating certain working registers to each address space for the MAC class of instructions. The core does not support a multi-stage instruction pipeline. However, a single stage instruction pre-fetch mechanism is used, which accesses and partially decodes instructions a cycle ahead of execution, in order to maximize available execution time.

Most instructions execute in a single cycle, with certain exceptions. The core features a vectored exception processing structure for traps and interrupts, with 62 independent vectors. The exceptions consist of up to 8 traps (of which 4 are reserved) and 54 interrupts. Each interrupt is prioritized based on a user assigned priority between 1 and 7 (1 being the lowest priority and 7 being the highest) in conjunction with a predetermined 'natural order'. Traps have fixed priorities, ranging from 8 to 15.

STATUS REGISTER:

The dsPIC core has a 16-bit Status Register (SR), the LS Byte of which is referred to as the SR Low Byte(SRL) and the MS Byte as the SR High Byte (SRH) SRL contains all the MCU ALU operation status flags (including the Z bit), as well as the CPU Interrupt Priority Level status bits, IPL<2:0>, and the REPEAT active status bit, RA. During exception processing, SRL is concatenated with the MS Byte of the PC to form a complete word value which is then stacked. The upper byte of the SR register contains the DSP Adder/Subtractor status bits, the DO Loop Active bit (DA) and the Digit Carry (DC) status bit.

LoRAWAN:

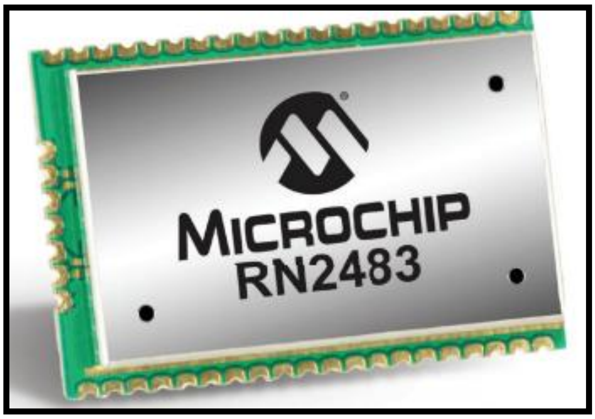


Fig 4: RN2483

OVERVIEW:

The RN2483 is a LoRa™-integrated modem with a range of more than 15 km (suburban), low power enabling a battery life greater than 10 years and the ability to connect millions of wireless sensor nodes to

LoRa gateways and IoT-connected Cloud Servers. This robust system is due to LoRa's unique spread-spectrum based modulation that is capable of demodulation 20 dB below noise level. This enables high sensitivity for ultra-long range, improved network efficiency and eliminates interference. The RN2483 modem operates over the 433 and 868 MHz license-free Industry Scientific and Medical (ISM) frequency bands and serves as the end-device in the LoRa network infrastructure (see Figure 1). The RN2483 has the complete LoRaWAN™ protocol stack on the modem and is easy to configure via simple ASCII commands through the UART, greatly reducing development time. The RN2483 is European R&TTEcertified, saving significant certification costs. Additionally, it combines a small form factor 17.8 × 26.7 × 3 mm with 14 GPIOs, providing the flexibility to connect and control a large number of sensors and actuators while taking up very little space. The RN2483 modem resolves the age-old wireless developer's dilemma to choose between longer range and lower power consumption. By employing the RN2483, you can now maximize both, while eliminating the cost of additional repeaters and increasing battery life. With its scalability, robust communication, mobility and the ability to operate in harsh outdoor environments, the RN2483 is well suited for a broad range of low-data-rate wireless monitoring and control designs.

GPS SR-92

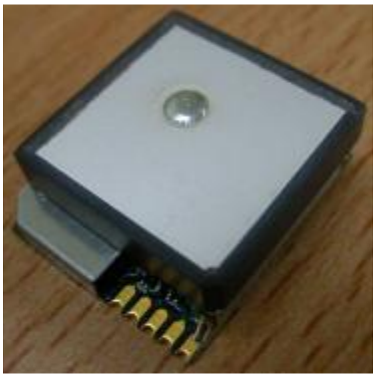


Fig 5: GPS SR-92

As shown in the above pictures, ProGin SR-92 is a low-power, ultra-high performance, easy to use GPS smart antenna module based on SiRF's third generation single chip. Its low power consumption and high performance enables the adoption of handheld applications. The slim design allows SR-92 to be placed on top side of the housing to have best GPS signal reception. The 5-pin I/O interface is then connected to the main board with either connector or wire soldering. The integrated antenna design helps reduce the RF and EMI issues to minimum. Fast adoption and high yield production becomes possible. The power control feature is very convenient to turn on/off power via GPIO control pin. It's especially useful in cases such as to turn off power as the user just wants to watch a movie and GPS function is not needed in the PMP case.

IV. SOFTWARE REQUIREMENTS:

Coding Tools

Writing, editing and navigating in source code are core tasks in application development. Therefore, the code editor is one of the key components of Qt Creator. The code editor can be used in the Edit mode to write code. The code editor offers a number of features that help developers maintain readability and coding style

- Syntax highlighting for keywords, symbols, and macros in C++ files. In addition, generic highlighting is supported for other types of files
- Code completion for elements, properties, ids and code snippets. This is also supported for developers' own classes in the current project. Checking code syntax and marking errors (with wavy underlining in red) while editing, making it unnecessary to use compilation simply as a way to find typos and syntax errors. Auto-indentation for source code layout.
- The ability to collapse and expand functions in the source code (code folding).
- The Locator navigation tool for quick access to files, symbols, hierarchy, and other information.

Support for refactoring code to improve the internal quality or your application, its performance and extendibility, and code readability and maintainability, as well as to simplify code structure.

Debugging:

In Debug mode, developers can perform common debugging tasks, including the following:

- Interrupt program execution.
- Step through the program line-by-line or instruction-by-instruction.
- Set breakpoints.
- Examine call stack contents.
- Examine and modify registers and memory contents of the debugged program.
- Examine and modify contents of local and global variables.
- Examine the list of loaded shared libraries.
- Disassemble sections of code.

Create snapshots of the current state of the debugged program and re-examine them later. as well as the call stack view and locals and expressions view.

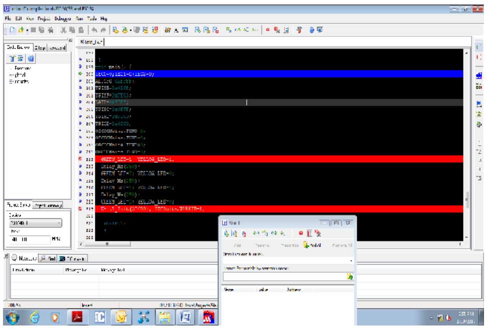


Fig 6: the Debug mode showing a breakpoint in the editor

V. RESULTS:

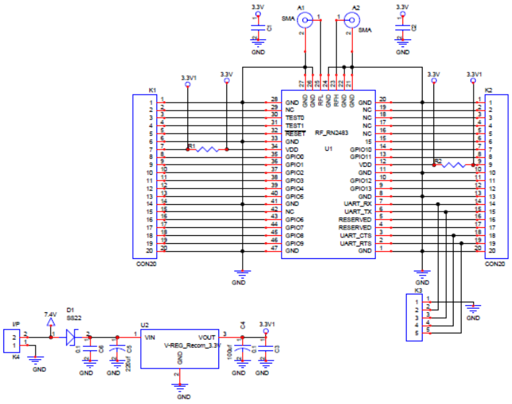
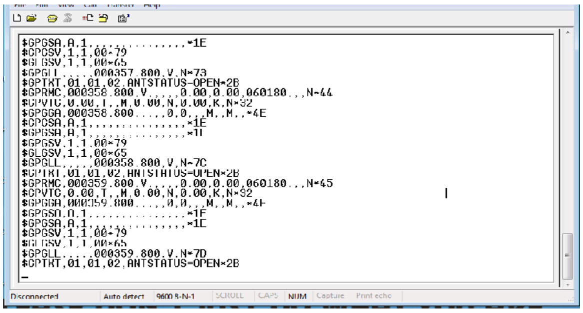


Fig 7: Schematic RN2483



```
$GPGSA,A,1,,,,,,,,,,,,,,,,*1E
$CPCSV,1,1,00*79
$GLGSV,1,1,00*65
$GPGLL,,,,,000357.800,V,N*73
$GPTXT,01,01,02,ANTSTATUS=OPEN*2B
$GPRMC,000358.800,V,,,,,0.00,0.00,060180,,,N*44
$GPVTG,0.00,T,,M,0.00,N,0.00,K,N*32
$GPGGA,000358.800,,,,,0,0,,,M,,M,,*4E
$CPCSA,A,1,,,,,,,,,,,,,,,,*1E
$GPGSA,A,1,,,,,,,,,,,,,,,,*1E
$GPGSV,1,1,00*79
$GLGSV,1,1,00*65
$GPGLL,,,,,000358.800,V,N*7C
$GPTXT,01,01,02,ANTSTATUS=OPEN*2B
$GPRMC,000359.800,V,,,,,0.00,0.00,060180,,,N*45
$CPVTC,0.00,T,,M,0.00,N,0.00,K,N*32
$GPGGA,000359.800,,,,,0,0,,,M,,M,,*4F
$GPGSA,A,1,,,,,,,,,,,,,,,,*1E
$GPGSA,A,1,,,,,,,,,,,,,,,,*1E
$GPGSV,1,1,00*79
$GLGSV,1,1,00*65
$GPGLL,,,,,000359.800,V,N*7D
$CPTXT,01,01,02,ANTSTATUS=OPEN*2B
_
```

Fig 8: Tool Terminal Window

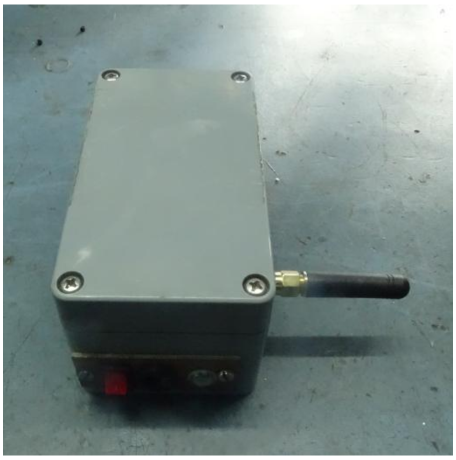


Fig 9: Transmitter Unit

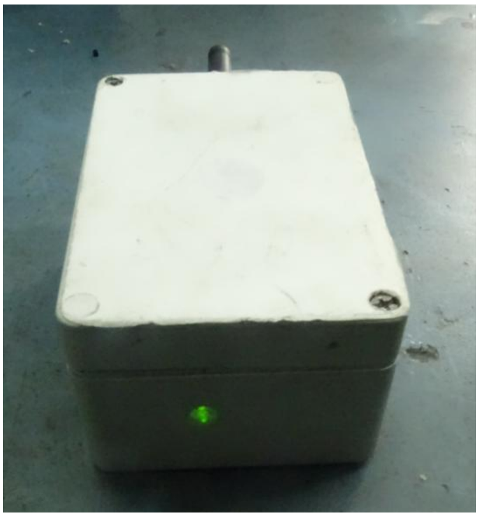


Fig 10: Receiver Unit

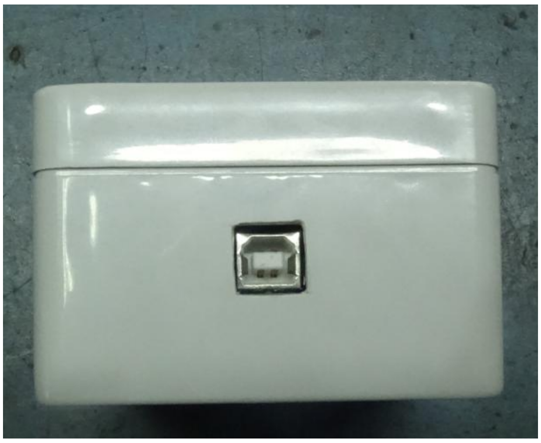


Fig 11: GPS Unit

VI. APPLICATIONS AND ADVANTAGES:

Applications

Several different domains

- Transportation and logistics
- Healthcare
- Smart environment (home, office , etc.)
- Personal and social domain

Advantages of IOT:

1. Data:

The more the information, the easier it is to make the right decision. Knowing what to get from the grocery while you are out, without having to check on your own, not only saves time but is convenient as well.

2. Tracking:

The computers keep a track both on the quality and the viability of things at home. Knowing the expiration date of products before one consumes them improves safety and quality of life. Also, you will never run out of anything when you need it at the last moment.

3. Time:

The amount of time saved in monitoring and the number of trips done otherwise would be tremendous.

4. Money:

The financial aspect is the best advantage. This technology could replace humans who are in charge of monitoring and maintaining supplies

VII. CONCLUSION:

The project "IMPLEMENTATION OF LONG RANGE SMART TRACKING SYSTEM USING INTERNET OF THINGS" is the concept which is currently being planned by major sector to eliminate the human dependency and to transmit the processed data over internet for data mobility and following global standards.In conclusion, the Internet of Things is closer to being implemented than the average person would think. Most of the necessary technological advances needed for it have already been made, and some manufacturers and agencies have already begun implementing a small-scale version of it.

FUTURE SCOPE:

This final section of the report outlines that some features potentially implementing in future releases. The current set of implemented features is a minimum to what a consumer would expect. The current system is used for accessing data at a limited use. The first and foremost thing is automation in daily life

REFERENCES:

1. dsPIC30F Family Reference Manual datasheet by Microchip Technology Inc.

2. RN2483 LoRa™ Technology Module Command Reference User's Guide by Microchip Technology Inc.

3. Product User manual GPS Engine Board, SR-92 ProGin Technology Inc.

4. Designing the Internet of Things Book by Adrian McEwen and Hakim Cassimally

5. Internet of Things: A Hands-on Approach Book by Arshdeep Bahga and Vijay Madisetti

6. Making Embedded Systems: Design Patterns for Great Software Book by John W White.

7. PIC Microcontrollers- Programming in C by Mikroelectronica.

8. D. Giusto, A. Iera, G. Morabito, L. Atzori (Eds.), The Internet of Things,Springer, 2010. ISBN: 978-1-4419-1673-0.

9. A. Chemudupati, S. Kaulen, M. Mertens, S. Murli Mohan,P. Reynaud, F. Robin, S. Zimmermann, White Paper "The convergence of IT and Operational Technology" 2012, Atos Scientific Community's IT/OT Convergence.

10. L. Atzori, A. Iera, G. Morabito, The Internet of Things: A survey, Computer Networks 54 (2010) 2787–2805.

11. H. Sundmaeker, P. Guillemin, P. Friess, S. Woelfflé, Vision and challenges for realising the Internet of Things, Cluster of European Research Projects on the Internet of Things - CERP IoT, 2010.