**Design and Implementation of a Secure, Reliable and Software Reconfigurable CubeSat Communications Subsystem**

**1 Ayman N. Mohi, 2 Jabir S. Aziz, 3 Lubab A. Salman**

***Department of Electronic and Communications Engineering, College of Engineering, Al-Nahrain University***

***PO Box 64040, Baghdad, Iraq***

**ABSTRACT**

A CubeSat is one of the most popular subjects in radio amateur or university projects nowadays. CubeSats provide students with a true hands-on experience in designing, developing, testing, and operating a real spacecraft and its ground station. This paper introduces the design of a secure and reliable communications subsystem to support the KufaSat CubeSat mission. The system is designed with commercial off-the-shelf components which allows an inexpensive construction of the subsystem and reduce overall cost and complexity. The system can be reprogrammed over the air to adjust the transmission power, data rate, modulation scheme, and operating frequency. The implemented communications protocol builds on packetizing the payload data prior to transmission to the ground station allowing thereby for the maintenance of a reliable link over the whole transmission session. This packetizing scheme also serves as a premier security measure against unauthorized data access. Finally, a prototype of the system is implemented and tested for proper operation as a verification of the design methodology and concept.

**Keywords:** CubeSat, Communications Subsystem, Terminal Node Controller (TNC), Nanosatellite, RFM22, KufaSat

**Introduction**

More and more people are now realizing the capabilities and the potentials of CubeSats. CubeSats are small cubic satellites people can use to explore space and take advantage of space technology at a very low cost, many universities were involved in CubeSat projects so that students can learn about space and spacecraft. The CubeSat has more extreme limitations than other satellites in terms of mass, power and volume standards. The volume of a single CubeSat unit is restricted to cm3 with a mass of no more than 1.33 kg, and commonly uses commercial of-the-shelf (COTS) components for its electronics [1]. Many CubeSats have been built and launched into space. To reduce space junk they are usually placed in low earth orbits and fall back to earth within a few weeks or months [2]. The basic idea of limiting the CubeSat specification was to overcome several challenges, by the simplification of the design infrastructure which makes it possible to design and implement a CubeSat at low cost within a relatively short period of time [1].

One of the primary objectives of any satellite is the ability to communicate with the earth. Our first requirement, therefore, is to create a communications subsystem that can communicate with our earth-based command station, reliably, while in orbit. Communication with the earth can be established using a wide range of radio frequencies, depending on the data rate requirements, earth station equipment costs, and FCC licensing restrictions [3]. There are limitations, standards and regulations which must be followed during the design process which introduces major difficulties and challenges. The CubeSat will operate in regulated frequency bands and the most common one is the 70 cm amateur band so there is a ready supply of RF power amplifiers and high gain antennas for use on the ground also the small size of the CubeSat makes it difficult to find a ready built-in radio transceiver for communications and the limited surface area restricts the amount of the power generated from the solar panels, restricting the power available for the communications subsystem. According to the regulations the system should not consume more than 1W of peak power when transmitting which introduces another difficulty.

CubeSats have evolved to the point where many of the encoders, modulators, filters, and decoders used by these communications systems can be implemented in software to overcome the persistent needs for increasing data rates and to make it easy to alter the system specifications without modifying the system hardware. To increase our design flexibility to match any future requirements, the suggested communications system is software reconfigurable in terms of the following system parameters: the transmitter’s power, data rate, modulation scheme, and frequency of operation. The system presented here is intended to support the KufaSat satellite earth imaging and remote sensing mission sponsored by the University of Kufa in Iraq as part of its academic research program [4].

In addition to this introductory section, this paper is organized as follows. The system architecture and the design procedure is outlined in Section 2. Section 3 discusses the suggested communications protocol and highlights its main advantages in terms of reliability and security. Practical implementation aspects and challenges of the proposed design are discussed in Section 4. Finally, conclusions and recommendations for further development of the subsystem are given in Section 5.

**System Architecture**

Based on the available regulations and standards, a set of requirements must be identified before proceeding to the design process and it’s illustrated as follows:

The frequency chosen for the uplink and downlink is the UHF (435 MHz).

The transmission power is 100 mW (or 20 dBm).

The orbit altitude is 600 km low earth orbit (LEO).

The dimensions are according to the regulations and the choice of KufaSat to be 1U CubeSat (10 x 10 x 10 cm3), so the communications unit inside the CubeSat must not exceed (10 x 10 cm2) to fit in the Cube.

The uplink and downlink data rate should be at least 1200 bps and the preferred data rate is 9600 bps for payload data.

The modulation scheme is FSK.

Designing and implementing a data packet encoding and decoding algorithms.

A new algorithm to generate an audible CW Morse coded beacon signal.

The suggested communications system architecture was implemented using COTS components. The RFM22 transceiver was chosen as the main data transceiver controlled by an Arduino Microcontroller with an integrated SD card system and a group of sensors to take the measurements. The block diagram in Fig. 2-1 illustrates the main parts of the communications system.

The adopted solution as we can observe that the uplink, downlink and the beacon signals share a common data link in a half-duplex regime at the frequency of (435 MHz). The advantages gained from this design is to reduce the size, power consumption, cost and complexity of the total system, by adopting a time sharing principle to keep the system efficient. Other CubeSat communications systems incorporate a separated beacon board for transmitting the beacon signal and also some of them chose 2 frequency bands (VHF, UHF or S-Band) for uplink and downlink. Those designs require more power, larger in size and more costly compared to our design.



*‎0‑1: System architecture.*

**The Microcontroller Unit**

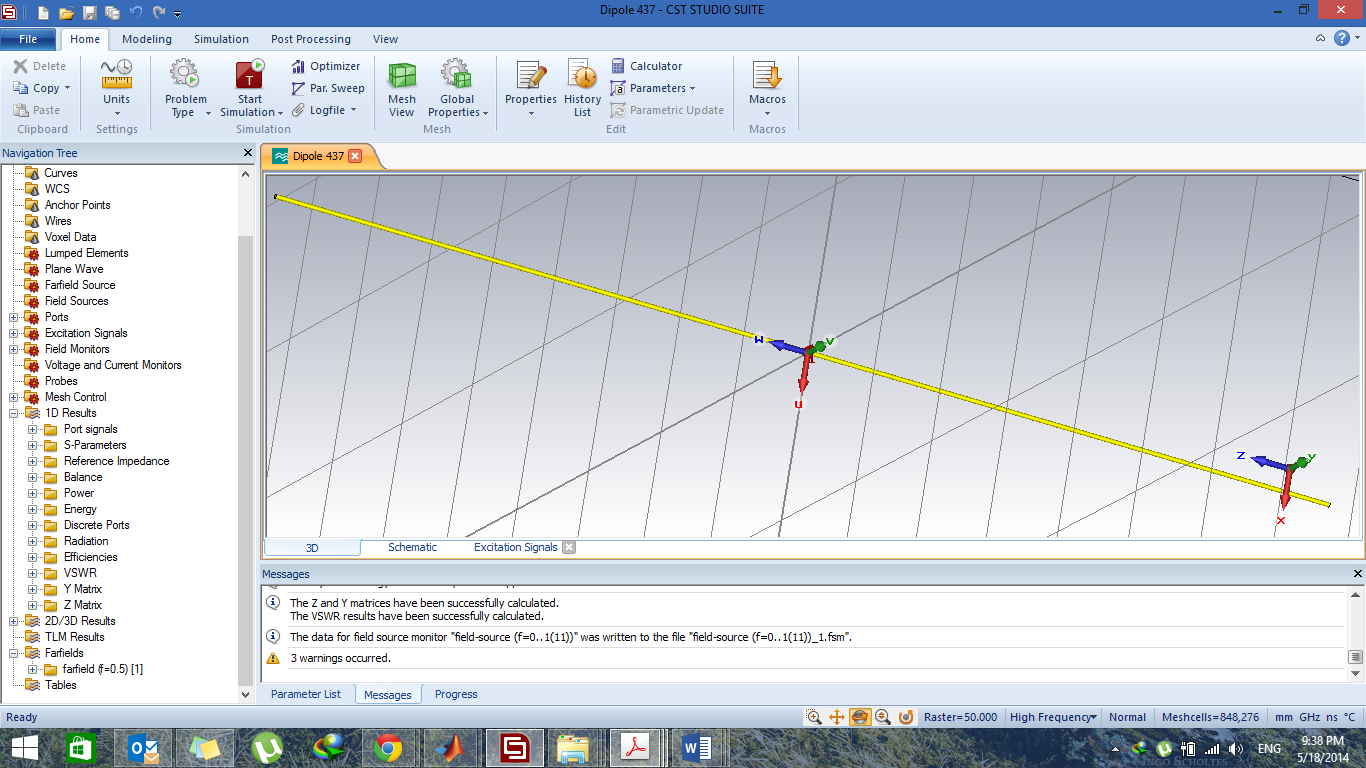
The microcontroller board chosen is Arduino board as the main TNC and it has been programmed using the Arduino programming language based on C language. The Arduino board communicates with the computer through a USB interface [5]. Due to size limitations the Arduino Nano has been chosen for our CubeSat communications system and the Arduino Mega for the ground station since there is no size limitations in the ground station. The choice of Arduino over other microcontrollers like PIC was due to some advantages like: it's relatively cheap, well documented and easy to program, also it is a simple open source development environment.

**Transceiver Unit**

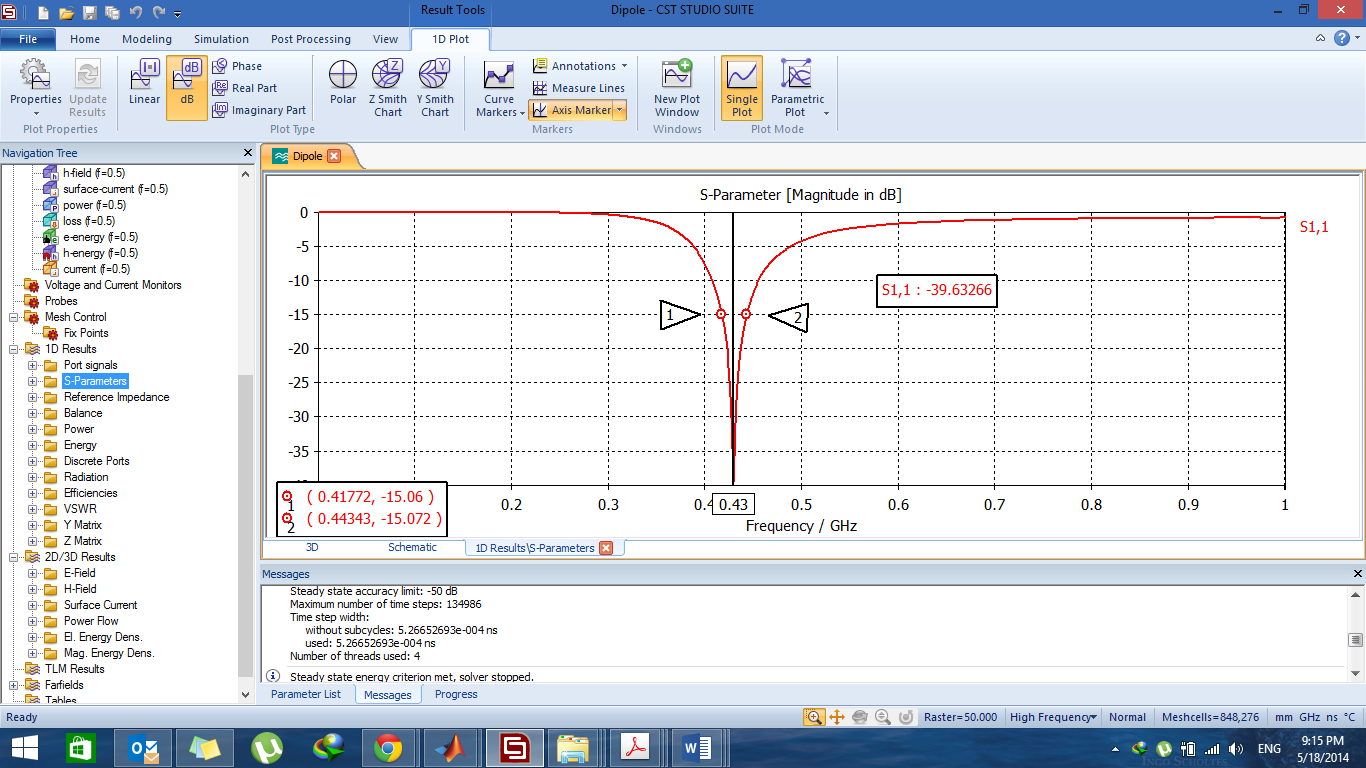
The RFM22 is low cost ISM transceiver module, and offers advanced radio features including continuous frequency coverage from 240–930 MHz and adjustable output power of up to +20 dBm, the communication between the Arduino and RFM22 is done by 4-wire SPI (Serial peripheral interface) connection. The wide operating voltage range of 1.8–3.6 V and low current consumption (85 mA max. TX current and 0.01µA for standby) make the RFM22 an ideal solution for battery powered applications [6][7].

**Antenna Design and Simulation**

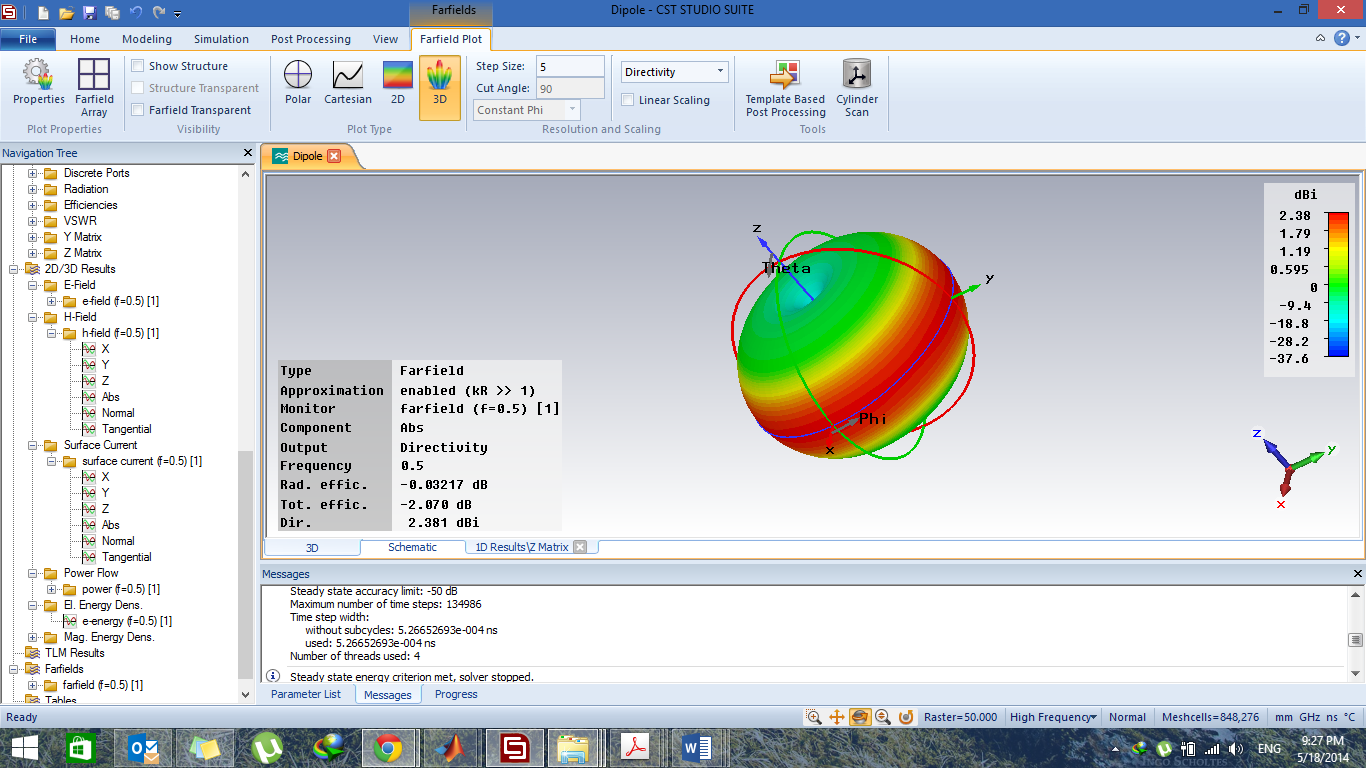
A set of calculations has been used to extract the wire length since the monopole antenna length is ¼ wavelength and the operating frequency is 435 MHz, using the equation , the parameters have been simulated after calculations using CST Software. Fig. 2-2, Fig. 2-3 illustrates the simulation of the Antenna with CST and S-Parameter of the Antenna and Fig.2-4 shows the radiation pattern of the antenna.



*Figure ‎0‑2: Dipole antenna designed with CST.*



*Figure ‎0‑3: S-parameter (return loss).*



*‎0‑4: Radiation pattern.*

**Communications Protocol**

The communications, data coding, beacon generation algorithms have been devolved using Arduino programing language based on C language and they are developed in a way to make the system and its parameters flexible and configurable.

The new ideas adopted in the designed system are obvious in the suggested software, the major advantage of that is the flexibility and compatibility with the different future requirements and they are illustrated as follows:

A common shared communications data link for the beacon, downlink and uplink is to reduce the size, cost and complexity of the system.

A timing protocol is introduced to control the information flow through the communications data link.

A new representation for the Morse coded beacon signals

A new packet coding technique for the payload information transmission.

Increasing communications system flexibility by introducing a software defined system parameters such as Tx power, modulation, frequency and data rate. This makes the system capable of matching any future requirement by KufaSat such as increasing data rate, changing the frequency of operation, flexible transmission power handling and so on while maintaining the same designed hardware.

**Communications Algorithm**

The communications protocol is described below:

The first process is the initialization process for the microcontroller to identify the variable and constant parameters and identify the input and output pins.

The following process is the initialization of the RFM22 transceiver applying the modulation, frequency of operation, Tx power and transmission data rate.

After the initialization is finished the memory system will start the continuous process of recording the payload information and saving it to the SD card system, while the RFM22 transceiver is transmitting the beacon signal in parallel.

The communications system periodically checks for an uplink signal reception.

If there is no uplink signal received the RFM22 continues transmitting the beacon signal.

When the first time uplink confirmation signal is received, the communications system will change the transmission mode from the beacon signal to the payload information and keep transmitting the payload data for the complete window time (approximately 5 minutes) and initialize the timer value to the exact time of the orbital period (96 minutes).

At the next CubeSat rotations, the payload information is transmitted to the ground station at either one of the situations: the timer value of the orbital period has reached zero or the uplink confirmation signal is received.

The reliability of this design can be understood by exploiting the situations below:

After launching the CubeSat to the orbit, at the moment the CubeSat is released from its P-Pod container, all the CubeSat subsystems are activated at this point including the communications subsystem.

If the CubeSat happens to be on eclipse time out of its 5 minutes communication window according to the suggested protocol it will maintain transmitting the beacon signal, so that the ground station shall be able to hear from it and respond by sending the uplink confirmation signal.

The other possibility is when the CubeSat is launched at any chance within its window time, at this point the ground station should be able to hear the beacon signal instantaneously.

For each one of the situations the uplink signal needs to be sent from the ground station and received by the CubeSat one time only to maintain the timing procedure is in order, since at the next passes the CubeSat shall depend not only on the uplink signal but also on the period timer which is updated at the end of each pass.

Fig. 3-1 illustrates the flow chart of the suggested and implemented algorithm for the proposed communications system.



*Figure ‎0‑5: Communications algorithm.*

**Morse Code Beacon Algorithm**

The Morse beacon signal in the suggested design is generated using the Arduino microcontroller and transmitted with the RFM22 transceiver unit by representing the dots and dashes by zeros and ones.

Morse code characters are all of length six or less, and each element is either a dot or a dash, so it would seem that it can be stored the pattern in six bits or less. Assuming that dots are zero and dashes are one, and store them so the first element gets stored in the least significant bit, and the next in the second most, and so on.



*Figure ‎0‑6: Beacon generation algorithm.*

The only trick is knowing when there are no elements left, because otherwise we can’t tell (for example K (-.-) from C (-.-.) to do that, we store a single extra one after all the other elements are taken care of. Then, when looping, we do the following, if the pattern is equal to one, it is done (that’s our guard bit). If not, we look at the least significant digit. If it is a zero, we have a dot, if we have a one, it’s a dash. We then get rid of that element (by dividing by two, or shifting right) and repeat.

Each character takes only a single byte (8 bits) to store its pattern while the most significant bit are always zero till the guard bit which is one and the reaming bits are the Morse sequence، and decoding is just done in the same way. Fig. 3-2 shows the Beacon generation algorithm.

**Payload Information Coding Protocol**

The payload important information is coded in a packet coding format and transmitted and received as packetized messages, the suggested software provides functions for sending and receiving messages of up to 255 octets on any frequency supported by the RFM22, in a range of predefined data rates and frequency deviations. Frequency can be set with 312Hz precision to any frequency from 240.0MHz to 960.0MHz.

All messages sent and received conform to following the packet format:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **PREAMBLE** | **SYNC** | **HEADER** | **LENGTH** | **DATA** | **CRC** |
| 4 Octets | 2 Octets | 4 Octets | 1 Octet | 0-255 Octets | 2 Octet |

The coded messages are unaddressed, reliable, retransmitted, acknowledged messages, but the acknowledgement process has its disadvantage of being time consuming but it’s necessary to avoid any information loss, and there is no need for addressing process since we have only one transmitter sending to another.

The Preamble and the SYNC Octets are responsible for the synchronization process in the data layer by identifying the start of each frame. The coding algorithm is represented in the flowchart shown in Fig.3-3.

The suggested packet format insure the security of the communication process since each frame start and end are identified by the PREAMABLE octets and the important information are encapsulated in the DATA section in the frame, that means any unauthorized attempt to extract the information will not succeed unless the attacker has a prior knowledge on the content of the PREAMBLE octets and the location of the DATA octets. Adding the acknowledgement and retransmission process to the coded messages insures the reliability of the communications process.

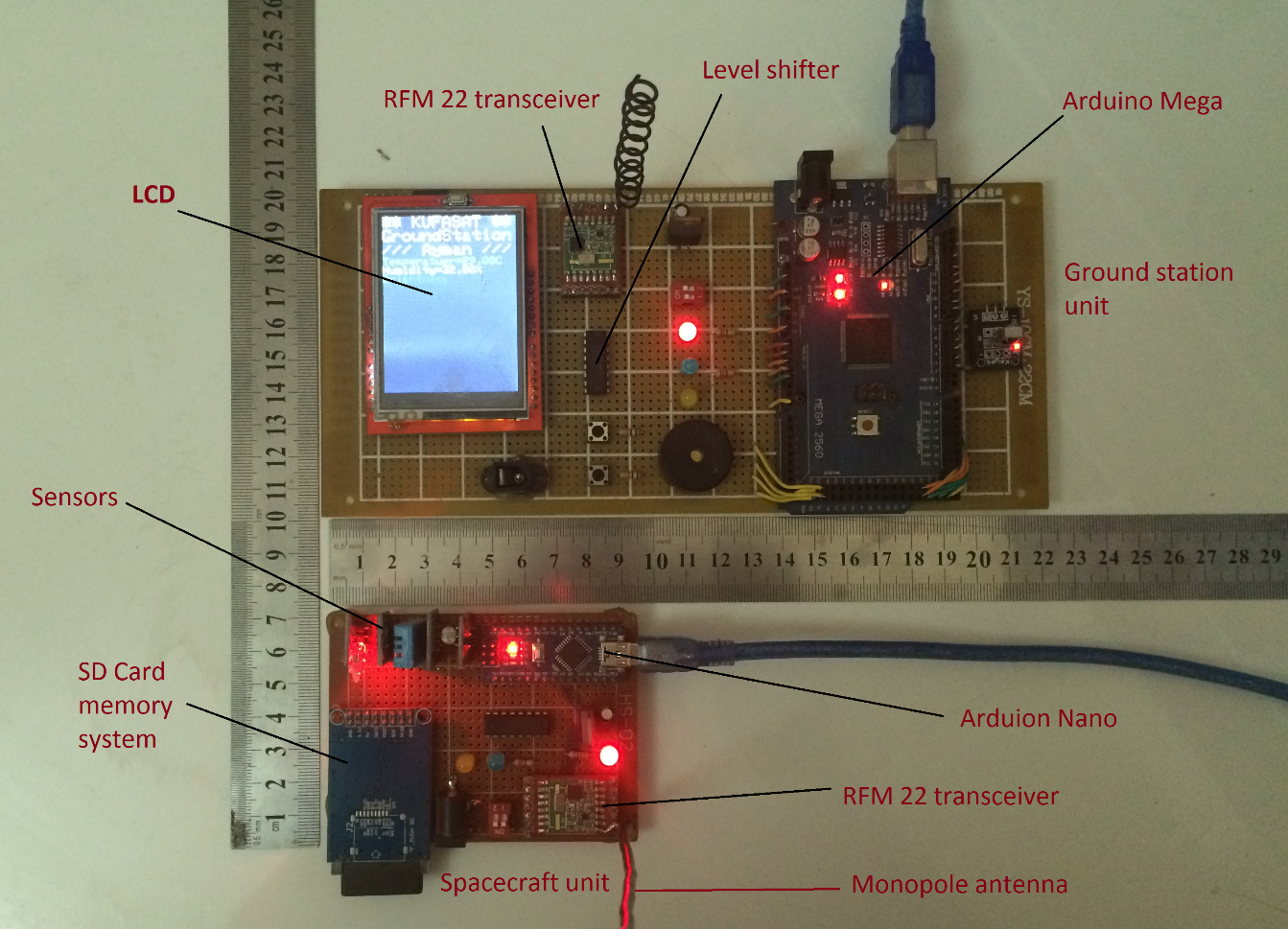


*Figure ‎0‑7: Information coding algorithm.*

**Practical Implementation**

The complete prototype system is shown in Fig.4-1 where it can be seen that the whole apace unit board size is no more than 10 x 10 cm2 in area.

The designed system was subjected to many tests to ensure its compatibility with the different circumstances and issues that may happen during its life time in space, the input and output signals are measured and recorded while different configuration parameters have been modified by software.

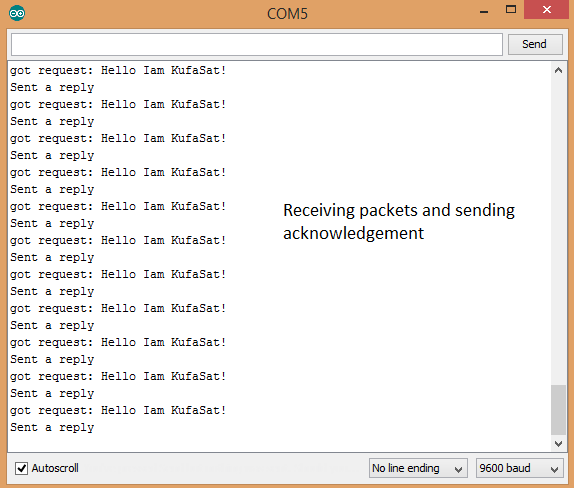
**

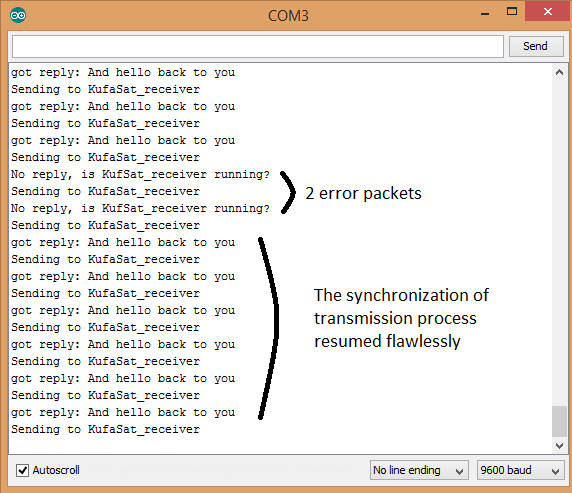
*Figure ‎0‑8: The complete prototype system.*

The first test was the beacon signal transmission and reception at 1200 b/s with 100 mW TX power and 434 MHz and 435 MHz frequency, the second test was payload signal transmission and reception at 1200 b/s and 100 mW TX power with the same frequency fixed, the third test was at 9600 b/s with 100mW and fixed frequency, the fourth test was on the reception of the uplink signal on the CubeSat side at 100 mW Tx power and 1200 b/S baud rate, the rest of the tests was done by changing frequency and fixing other parameters to ensure the possibility of software modification for all the parameters.

The results were viewed using the serial window on a laptop to view transmitted and received information and check if any the errors received. The serial port number 3 was identified for space segment and serial port 5 for the ground station segment.

The acknowledgment process is illustrated in Fig.4-2 indicating that the ground station shall send the acknowledgement packets after the reception of the test signal “Hello I am KufaSat!”.





*Figure ‎0‑9: Sending, receiving and acknowledgement tests.*

**Conclusions**

The System was designed using COTS components which makes it easy to modify and upgrade the system components. Furthermore, the system was designed as a software defined communication system in terms of frequency, Tx power, modulation and data rate parameters to achieve a highly adaptive communications system that can be easily incorporated into any CubeSat design. A new method of generating the Beacon within the same TNC was invented to reduce size, cost and complexity. Finally a secure and reliable data communications link achieved by introducing a new packet information coding protocol.

The system can be developed in future by increasing the transmission power capabilities either by introducing an efficient RF amplifier circuit or upgrading the COTS transceiver by more advanced space qualified equipment.

**References**

The CubeSat Program, “Cubesat Design Specification”, California Polytechnic State University, Regulations and Standards Document, 2009. Available at http://www.cubesat.org/ [online]

Muri, P. & McNair, J. “A Survey of Communication Subsystems for Intersatellite Linked Systems and CubeSat Missions”, Journal of Communications, 2012.

Klofas, B. Leveque, K. & Anderson, J. “A Survey of CubeSat Communication Systems”, Presented on Annual CubeSat Developers Workshop, 2013.

Mahdi, M. C., “Orbit Design and Simulation for KufaSat Nano- Satellite”, Artificial Satellite, Vol. 50, No. 4 - 2015.

Evans, B. “Beginning Arduino Programming, Writing Code for the Most Popular Microcontroller Board in the World”, 2011.

Hope Micro-electronics CO., LTD, “RFM22 Specification Data sheet, PDF Document”.

DeFelice, H. Michael, K. Robinson, S. “$50SAT\_Eagle\_2 Communications documentation”, version 1.2, 26 July 2014, Available at http://www.50dollarsat.info/ [online].

**الخلاصة**

في الوقت الحاضر عده جامعات حول العالم تهتم بمشاريع متعلقه بتصميم وتجميع وتشغيل القمر الصناعي النانوي والمعروف ايضا بالمصطلح (CubeSat) لغرض زيادة الخبره البحثيه عند الطلاب والباحثين في هذا المجال.القمر الصناعي CubeSat هو معيار جديد ظهر بعد سنه 2000 ميلادي وتم اصدار مجموعه من الضوابط والمقيدات من قبل مختلف المنظمات العالميه في مجال الاتصالات والاقمار الصناعيه لتحديد شروط وضوابط التصنيع لهذا القمر النانوي، هذا البحث يعالج تصميم وتنفيذ نموذج اولي لنظام الاتصالات للقمر الصناعي الذي من الممكن الاستفاده منه في تصميم نظام الاتصالات للقمر الصناعي النانوي العراقي كوفه سات (KufaSat) وھو برنامج قمر صناعي نانوي ینفذ من قبل جامعة الكوفة، تم التصميم باستخدام الاردوينو مايكروكونترول مع الاستعانه بالمرسل والمستقبل الذي اختير لهذه المهمه وهو RFM22. تتمثل الفكره الرئيسية للتصميم بجعل النظام قابل للبرمجه وحتى بعد الاطلاق والتشغيل مع امكانيه تغيير المعاملات المهمه لظمان نجاح عمليه الاتصال ومنها تردد الارسال وومقدار الطاقه المرسله ومعدل نقل البيانات لاجل أن يكون النظام قابل للتكيف مع مختلف المتطلبات المستقبليه كزياده معدل نقل البيانات وغيرها، وتم تزويد المنظومه بنظام تشفير لحمايه المعلومات المهمه والحفاظ على أمان وأستمراريه عمليه الأرسال والأستلام.