SSC22-P4-27

Development of University-friendly CubeSat Bus and Ground Station Architecture Using Software-defined Radios

Jin S. Kang
United States Naval Academy
590 Holloway Rd., MS 11B, Annapolis, MD 21402; 410-293-6411
kang@usna.edu

ABSTRACT

The goal of the research and development presented in this paper is to introduce a CubeSat bus and ground station architecture that is made to be much more approachable to schools and universities. The three main pillars of the effort are low-cost, maintaining flexibility, and lowering the bar of entry. The presented CubeSat bus includes PyCubed board which houses most of the core satellite bus components on a single board. The board can handle main processing, data storage, UHF radio communication, telemetry sensors, and power management. This UHF radio is paired with a software-defined radio (SDR) that serves as the ground station radio. For a faster data rate downlink of payload data, a low-cost SDR (Ettus B205mini) is paired with a RaspberryPi processor. By leveraging the flexibility of SDRs, one SDR at the ground station is agile enough to provide UHF up/downlink for the CubeSat bus comm, as well as receiving S- or X-band payload data downlink. This proposed architecture will enable project teams to rapidly achieve a baseline capability with the satellite bus such that the development schedule and cost can be drastically reduced while providing the students with the full-cycle experience of satellite engineering.

INTRODUCTION

CubeSats have become quite popular among educational institutions in recent years. The main thrust has been led by universities in the past, but more and more middle and high schools are also establishing CubeSat-based STEM programs. One main reason for the popularity of CubeSat-based programs is due to easier access to space, both from hardware and regulation perspective, made possible by establishing CubeSat standards that has become well understood by the satellite community. This has made it much easier for CubeSats to 1) secure launches, and 2) promote rapidly growing commercial market. Both factors further make CubeSat-based programs attractive for STEM education. Development teams that have CubeSat development experiences, however, quickly realize that the actual design and development of these satellites are far from “standard”. While there are many commercial options available for individual components, they often go through revisions too quickly for CubeSat programs to establish a standardized bus setup that can be maintained over multiple missions. Regulation changes also limit hardware options, sometimes resulting in a complete redesign of “standard” satellite bus systems. Cost also quickly becomes prohibitive when more capable hardware needs to be considered.

As CubeSat-class satellites start taking on operational roles, the demands on their performance have been steadily increasing. Accurate pointing capability, high power generation, and high data rate data downlink have now become a common part of CubeSats. Educational and scientific CubeSats developed by university-level programs are also trending towards higher performance satellites, but it is difficult to keep up with the increased cost in commercial components. As an example, a complete student-build CubeSat at the United States Naval Academy (USNA) cost less than $25,000 whereas a typical attitude control system or a high-performance radio can cost close to $100,000 by themselves. For satellite programs that focus on education and training of students and thus require a high development cycle cadence such as the program at USNA, this level of cost is unsustainable. USNA usually has two CubeSat development projects every year, with one satellite launched into space. This type of education focused programs can benefit from a reliable, low-cost standard bus that can deliver a reasonable performance. It also enables students to be able to focus on the payload development and integration, instead of re-inventing satellite bus every year.

The objective of this research and development effort is to design and test a low-cost CubeSat bus that can serve as the standard bus suite for future satellites. In particular, the satellite must be able to provide a higher data rate downlink such that a more sophisticated payloads can be implemented in the future where the payloads can generate higher amounts of data to be downlinked. The paper will detail the components and setup of the proposed CubeSat bus and Ground Station Architecture, focusing on SDR setup for both the satellite downlink and the ground station.

SATELLITE SYSTEM DESCRIPTION

System Architecture Overview

The USNA baseline CubeSat that will serve as the standard bus for future missions is shown in Figure 1. The satellite bus as shown in the Figure will only take up 0.5U (10 x 10 x 5 cm) volume. For payloads that are small and do not require a large amount of power, it can be comfortably integrated into a 1U (10 x 10 x 10 cm) CubeSat. USNA often develops satellites that are 1.5U in size because two USNA satellites can then be launched in a typical 3U-sized launchers. For the first Naval Academy Standard Bus (NASB) satellite to be launched and tested on orbit in 2024, a 3U (10 x 10 x 30 cm) CubeSat has been chosen. This is due to the increased power demand of the payload. Two main payloads are a low-cost software-defined radio (SDR) operating in S-band and a scientific payload developed by the United States Air Force Academy student team. The SDR payload is described in detail in the later section.

A UHF radio using LoRa protocol is used as the main command and control radio. The radio operates in half-duplex mode on a single frequency. The satellite has 90˚ phased double-monopole antennas for omni-directional communication and this will be paired with UHF yagi antenna of the ground station. For data dump operations requiring a higher data rate, an S-band radio is also used by NASB. This radio will operate in transmit-only mode using LoRa protocol and act as the main data downlink radio. A directional patch antenna is integrated into NASB and the receiving antenna on the ground is a 3m parabolic dish antenna.

Main Satellite Bus

The main satellite bus of NASB is based on PyCubed board [1]. PyCubed board is an open-source satellite main board that is developed by Maxwell Holliday and his team at Stanford University. It provides a complete hardware and software solution that is made approachable through easy-to-follow documentation and utilization of CircuitPython. The board has flight heritage and is proven reliable. Also important is the parts selected and used on the board. They are COTS components but were selected based on their reliability and radiation tolerance, resulting in a more robust hardware to be operated in the low-earth orbit [1]. Figure 2 shows general board layout.

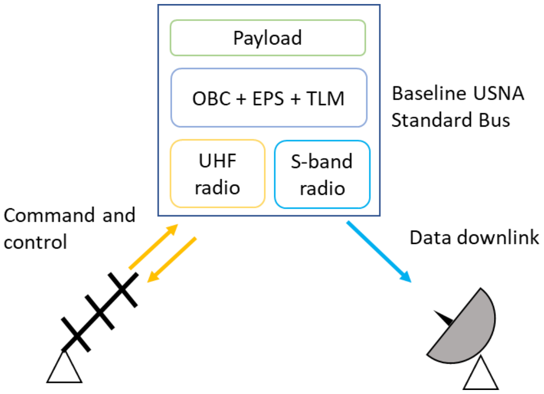

Figure 1: Baseline satellite architecture. (OBC: onboard computer, EPS: electrical power system, TLM: telemetry sensors and data)

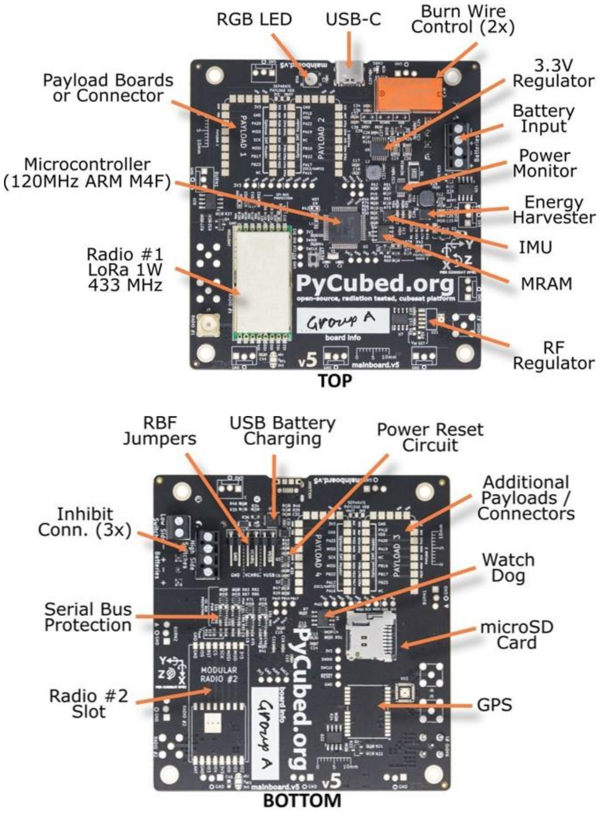

Figure 2: PyCubed board layout [2]

Included IMU and power monitoring sensor provide the basic telemetry data including angular rates, magnetometer readings, temperatures, as well as various voltages and currents readings. Solar panels are directly connected to the board. Peak power tracking and battery charging is also performed by PyCubed so a battery pack can be directly connected to the board to serve as the energy storage device without needing a separate power conditioning and distribution board. An S-band radio is installed in the “Radio #2 Slot” to serve as the main downlink radio. Communication architecture is described in more detail in the following sections. One main function missing is the attitude determination and control system (ADCS). For a 1U configuration, NASB is free-tumbling, and will not use the S-band antenna due to its directionality. For NASB in 3U configuration, magnetic pointing will be used to point the S-band patch antenna towards the ground station at USNA. This passive pointing scheme is also described in more detail in a later section.

The main advantage of this setup is its “ease of use”. The board comes with a flight software that is ready to fly. The software implements task-priority-frequency setup that can be easily modified and edited. The programming language is Python, which is much more approachable for the students who are inexperienced in coding techniques. Components are also much more affordable for university-level programs. A single PyCubed board after all components are populated is approximately $300. Even with an addition of a battery pack, satellite structure, and solar panels, the cost of the satellite can be kept very low such that one or two satellites can be developed and launched every year by typical CubeSat development programs at university level. These are the main reasons why PyCubed was adapted as the core of bus component of NASB.

Communication Architecture

The standard, baseline communication architecture of NASB consists of one UHF radio running at half duplex for the main command and control of the satellite, and one S-band radio used as a transmitter only for the purpose of higher data rate downlink. These are regular hardware radios that use LoRa protocol. LoRa is a proprietary protocol often used by IoT devices that incorporates frequency hopping and error correction to enable long range communication with lower transmission power [3], [4]. Both UHF and S-band radios will utilize Semtech SX1276 and SX1280 IC chip sets respectively. This setup will serve as the baseline satellite bus communication architecture for the future USNA satellites. The plan is to replace the S-band radio with an SDR for added flexibility and capability for the later generation NASB-based satellites. For the first iteration of NASB mission, an SDR will serve as a payload to the satellite. The SDR also implements LoRa protocol in order to characterize its performance and also compare to the performance of a hardware chipset radio.

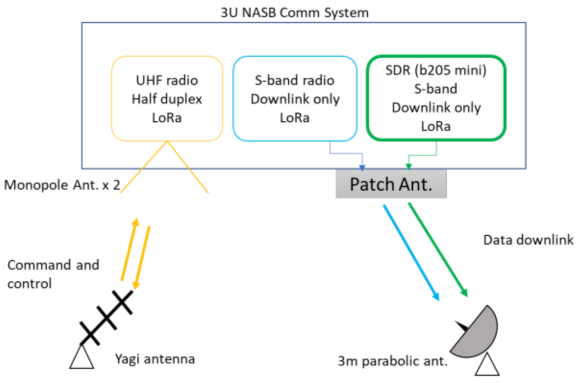

Figure 3: NASB communication architecture

The S-band hardware radio that is the part of the standard suite of NASB will have a direct connection to the data file(s) to be downlinked as the radio physically resides on the PyCubed board. This is shown in Figure 4. The SDR has its own computer that controls it in the form of a RaspberryPi board (more details in the later section). Accordingly, the SDR payload has its own SD card storage capability. The operation will consist of the bus system transferring its data and/or file(s) to the payload to be stored locally. A command from the ground will then be forwarded by PyCubed to the payload to initiate downlink. Figure 5 shows the main components of NASB bus.

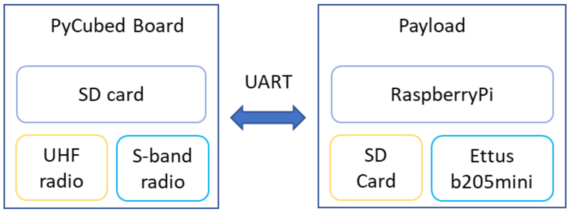

Figure 4: NASB radio setup and data flow

Figure 5: Picture of main components of NASB bus with SDR Payload

Antenna pointing

Having an S-band radio necessitates that the antenna be pointed towards the ground station. To be able to close the link with the ground station with a reasonable transmission power, an antenna with some gain is needed. A patch antenna works well for this application, but the beamwidth of a typical S-band antenna is 120˚ or so, requiring the satellite to point the antenna in the general direction of the ground station. This often leads to a big challenge to CubeSats in terms of cost, volume, and power consumption. As mentioned previously, for satellite programs that produce CubeSats regularly, such increase in cost for having an active attitude control system makes the program unsustainable.

Accordingly, a new passive pointing method was devised for NASB. The satellite is to use a permanent bar magnet to point its z-axis (where the patch antenna is mounted) towards near-nadir. This configuration is shown in Figure 6. As can be seen in the Figure, hysteresis rods are also added in the other two axes to damp out the angular rates. Permanent magnets are used often in CubeSats for passive attitude stabilization but never used for antenna pointing because the magnetic axis will always align with the local magnetic field line, and thus the direction of axis cannot be controlled. However, analysis of the local magnetic field line, along with the simulated performance of the magnetic control system as shown in Figure 6 show that the patch antenna mounted in the magnetic axis can be pointed towards the ground station at the Naval Academy to within 28 ˚ as it passes over the ground station [5]. The pointing stability performance and power analysis done for a 3U CubeSat in a notional polar orbit at 400 km altitude show that an S-band link can comfortably close with an expected downlink performance of 8 MB/day on average as compared to 2.6 MB/day for 9600 bps UHF downlink [5]. This means that NASB can provide S-band communication using USNA ground station at a drastically reduced satellite hardware cost.

SDR COMMUNICATION ARCHITECTURE

There are now a few commercial options available for CubeSat SDRs with prices ranging from estimated $80,000 to $150,000. SDRs provide much flexibility and configurability on the fly, able to deliver a much better performance "on the fly". This makes the use of SDR attractive for CubeSats as well, but there are two main drawbacks for SDRs that are particularly challenging for typical CubeSat programs: cost and power consumption. As mentioned, PyCubed board can be implemented for approximately $300 which include both UHF and S-band radios. A commercially available SDR would cost more than 300 times the rest of the satellite bus electronics. SDRs also operate on FPGAs which usually require much higher power consumption to operate as compared to typical CubeSat processors [6]. The goal of the NASB project is to develop, implement, and test on-orbit a low-cost SDR that can also be operated within a typical smaller CubeSat power budget.

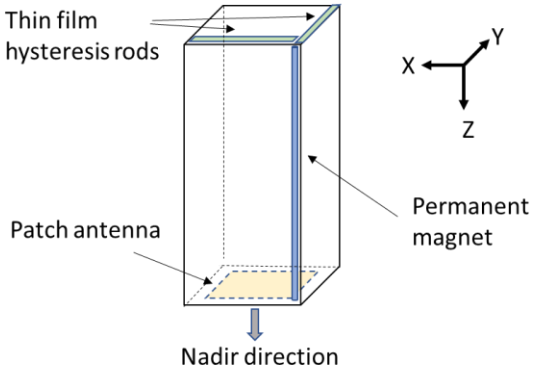

Figure 6: Permanent magnet and hysteresis rod configuration of NASB [5]

SDR Options Considered

There are many ground station projects that utilize low-cost, commercial SDRs. Some of the most widely used SDRs are HackRF, LimeSDR, and PlutoSDR. Example projects implementing these radios are [7], [8] among many and a comprehensive study of CubeSat communication architecture is described in [4]. All of these do not have their own microcontroller for signal processing, and thus needs a separate computing unit for configuring the radio to behave in the desired way. This is often done using ground station computers or smaller processors like RaspberryPis. These radios seem to have been designed without consideration towards low sleep-mode power consumption where the radios still consume a significant amount of power even when idle. While these radios can be implemented on CubeSats, it is clear from the design choices that the radios were meant for ground station side usage more than space side of the satellite communication architecture. YARD Stick One SDR was also considered for its ease of use and low power consumption. This SDR runs on a built-in software such that an external processor for radio configuration is not needed, resulting in further power savings. However, YARD Stick One is not a true SDR in that only certain parameter such as frequency, modulation, etc. can be changed within set parameters and does not provide the true flexibility of SDRs. The frequency range of YARD Stick One is also limited to UHF or lower frequencies.

For performance reasons, Ettus SDRs were looked at more closely to be adapted. Operating frequency range of the previously mentioned SDRs may be more limited, as well as the signal processing performance within the range may not be as good as Ettus SDR devices. From Ettus models that may be suitable for space-side implementation, B205mini provides a better performance at a competitive price, while being able to fit inside 10 x 10 cm footprint. Ettus B205mini was chosen for NASB's future S-band SDR for its robustness, flight heritage, and better signal processing performance. Figure 7 shows a picture of the unit with a metal protection casing that can serve as a heat sink in orbit.

Figure 7: Picture of Ettus B205mini-i [credit: ettus.com]

NASB SDR Architecture Description

A general NASB SDR architecture is shown in Figure 4, labeled as "Payload". Ettus B205mini is controlled by a RaspberryPi unit. RaspberryPi runs GNU Radio flowgraph in operating the SDR. A separate Python script sends and receives data from GNU Radio/SDR and controls the data processing, command handling, and data flow. LoRa protocol is implemented in GNU Radio. RaspberryPi is also connected to PyCubed board via UART connection. SPI would have been preferred but RaspberrPi 4 is not able to operate in slave mode and thus an UART connection was dedicated to communicating with the SDR. An added advantage of having an SDR-controller RaspberryPi onboard is that implementation of an imager becomes quite simple where high-quality imager can be easily implemented on a RaspberryPi in a plug-n-play fashion. As the image files are often large in size, it also has the benefit of having the radio attached to the processor (RaspberryPi) that handles the image files directly, eliminating the need for file transfers from the main PyCubed to the transmitting radio.

GNU Radio flowgraph for NASB was configured to implement LoRa protocol, both for uplink and downlink. This means that NASB will communicate using LoRa for both command and control in UHF as well as data dump downlink in S-band. Other protocols were also considered. As noted in [9]–[11], GMSK modulation seems to work well for LEO communications. In particular, the advantage of the amplifiers being able to work in saturation mode for GMSK is an advantage for CubeSats that are limited in power and budget. CCSDS standard is also very popular among CubeSats. LoRa protocol was chosen for NASB because of it provides a means to close the link easily over a long range with lower transmission power as the protocol utilizes spread spectrum and error coding. It also provides a low-cost solution to radios where LoRa radios are available for tens of dollars and can be used out-of-the-box without any software programming needed. These radios have also been tested in LEO and have performed well [2]. As NASB will be using a passive antenna pointing, a link that is easier to close is more attractive at the cost of lowered data rate. Transmitting at 2.2 GHz frequency, the radio can downlink a 170 kB size picture in about 5.5 minutes in current setting which corresponds to approximately 4.4 kbps of pure data (not including the overhead). As NASB comes standard with an S-band LoRa hardware radio, the SDR payload will be used in parallel to characterize its performance as compared to the commercial hardware radio chipset.

USNA Ground Station

The Naval Academy Ground Station is a part of the Space Systems Engineering Laboratory (SSEL) that focuses on the education and training of students. The ground station is located at approximately 39.0˚ N latitude and 76.5˚ W longitude. This latitude and longitude coincides with a slight rising of the geomagnetic latitude at the location, resulting in the "North Pole" magnetic direction pointing closer to nadir in comparison to other parts of the world (for example, Europe) [5]. This results in the satellite being able to point towards the ground station with off-nadir angle of approx. 28˚, putting it well within the beam width of a typical satellite patch antenna. The ground station at SSEL consists of two sets of UHF/VHF yagi antennas and a 3m parabolic dish antenna, as shown in Figure 8. The control cabinet houses Ettus USRP2 SDRs for signal processing, along with computers that controls the SDRs. The GNU Radio flowgraph used for configuring SDRs are configured for APRS and LoRa protocols, the two main protocols currently used by USNA satellites. Some coax cables will also run down to SSEL Mission Operations Center (MOC) as a backup and also for classroom demo purposes, but most of the control and data handling will be done locally at the control cabinet located at the antenna farm, and communicated down to MOC via fiber lines.

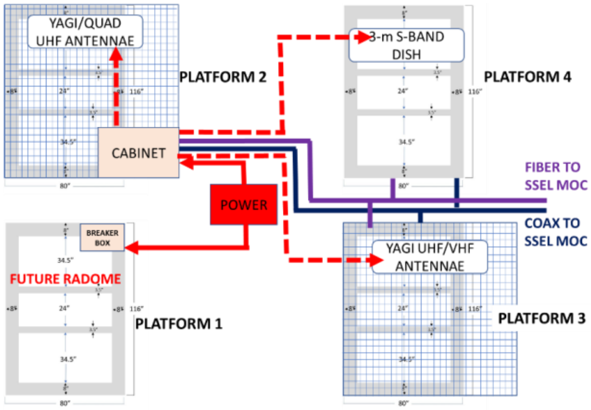

Figure 8: USNA Ground Station Antenna Farm Layout

Power Consumption Considerations

Power draw of the SDR setup continues to be a big concern. With the proposed setup as described above, an SDR for a CubeSat can be implemented with approximately $2,000, which is a significant reduction in cost compared to commercially available CubeSat SDRs. However, power consumption is not reduced as compared to commercial options.

RaspberryPi 4 was initially used for its computing power. It was able to run USRP software (UHD) and GNU Radio without much issue. The CubeSat SDR radio development for NASB was done with this setup. However, when B205mini-i is paired with RaspberryPi 4, after initialization, the SDR system draws 4.5 W at idle without amplifiers. As B205mini does not have a sleep mode, the radio draws about 2-3 W consistently after initialization, regardless of its operation. In order to reduce the power consumption, RaspberryPi Zero was substituted as the main control processor. This brought the power consumption down by about 2 W, but RaspberryPi Zero's processing speed was not fast enough to keep up with the signal processing requirements, resulting in a significantly reduction in performance of the SDR setup. Looking at the power consumption and processing power specifications, the new RaspberryPi Zero 2 seems to be the best choice for the NASB SDR system. It appears to be fast enough to run S-band signal processing tasks while consuming much less power than RaspberryPi 4 boards. However, due to the supply chain issues, RaspberryPi Zero 2 boards were not available and thus the actual testing has not yet been performed. The expected idle power consumption for the proposed setup of RaspberryPi Zero 2 and B205mini is approximately 2.5 W. As a comparison, the lowest power consumption SDR, YARD Stick One paired with RaspberryPi Zero consumes 1.4 W in idle.

CONCLUSION

This paper outlines a CubeSat communication architecture that utilizes software-defined radios. The focus of the proposed architecture is to make it university friendly lowering the bar of entry. This is accomplished by developing a CubeSat SDR system that is low-cost while maintaining all the flexibility that comes with SDR technology. An SDR system that pairs Ettus B205mini SDR with RaspberryPi processor is described. Running LoRa protocol in S-band, the setup can comfortably downlink at about 4.4 kbps in payload data (not including the overhead, etc.). A satellite bus that is based on PyCubed board and is made approachable, resulting in further lowering the bar of entry. The entire satellite, including the SDR radio is estimated to cost less than $5,000, not including the structure and solar panels. This level of cost would enable satellite programs that are focused on education and training, the programs that require CubeSats to be developed at a much more rapid pace. For programs operating in North America, S-band communication with CubeSats is possible without a need for an active attitude control system by implementing the proposed passive pointing architecture, drastically reducing the cost of the satellite bus. The United States Naval Academy is expected to be able to continue its one-CubeSat-a-year development pace by implementing the proposed design as a standard bus such that every student can experience the entire scope of a satellite development cycle from the beginning to the end while focusing more on the payload design, integration, and testing instead of having to spend most of their time on bus development.

REFERENCES

[1] J. Kang, J. Gregory, S. Temkin, M. Sanders, and J. King, "Creating Future Space Technology Workforce Utilizing CubeSat Platforms: Challenges, Good Practices, and Lessons Learned," presented at the AIAA Scitech 2021 Forum, VIRTUAL EVENT, Jan. 2021. doi: 10.2514/6.2021-1437.

[2] M. Holliday, A. Ramirez, C. Settle, T. Tatum, D. Senesky, and Z. Manchester, "PyCubed: An Open-Source, Radiation-Tested CubeSat Platform Programmable Entirely in Python," *Small Satell. Conf.*, Aug. 2019, [Online]. Available: https://digitalcommons.usu.edu/smallsat/2019/all2019/65

[3] C. A. Trasviña-Moreno, R. Blasco, R. Casas, and Á. Asensio, "A Network Performance Analysis of LoRa Modulation for LPWAN Sensor Devices," in *Ubiquitous Computing and Ambient Intelligence*,

Cham, 2016, pp. 174–181. doi: 10.1007/978-3-319-48799-1_21.

[4] N. Saeed, A. Elzanaty, H. Almorad, H. Dahrouj, T. Y. Al-Naffouri, and M.-S. Alouini, “CubeSat Communications: Recent Advances and Future Challenges,” *IEEE Commun. Surv. Tutor.*, vol. 22, no. 3, pp. 1839–1862, thirdquarter 2020, doi: 10.1109/COMST.2020.2990499.

[5] J. S. Kang, J. T. King, C. R. Anderson, and M. H. Sanders, “Performance Analysis of Passive Magnetic Pointing Method for Implementation of Directional Antennas on CubeSats,” *Int. J. Aeronaut. Space Sci.*, Mar. 2022, doi: 10.1007/s42405-022-00448-5.

[6] M. Hessar, A. Najafi, V. Iyer, and S. Gollakota, “TinySDR: Low-Power SDR Platform for Over-the-Air Programmable IoT Testbeds,” 2020.

[7] E. Santos, A. Prieto-Guerrero, and R. Aguilar-Gonzalez, “A Software Development Based on Software-Defined Radio Devices for Transmitting Digital Signals,” Nov. 2019. doi: 10.1109/ICMEAE.2019.00032.

[8] R. P. Hudhajanto *et al.*, “Low Cost Nano Satellite Communication System Using GNURadio, HackRF, and Raspberry Pi,” in *2018 International Conference on Applied Engineering (ICAE)*, Oct. 2018, pp. 1–4. doi: 10.1109/INCAE.2018.8579395.

[9] J. Muslimin, A. L. Asnawi, A. F. Ismail, and A. Z. Jusoh, “SDR-Based Transceiver of Digital Communication System Using USRP and GNU Radio,” in *2016 International Conference on Computer and Communication Engineering (ICCCE)*, Jul. 2016, pp. 449–453. doi: 10.1109/ICCCE.2016.100.

[10] A. Gaysin, V. Fadeev, and M. Hennhöfer, “Survey of modulation and coding schemes for application in CubeSat systems,” in *2017 Systems of Signal Synchronization, Generating and Processing in Telecommunications (SINKHROINFO)*, Jul. 2017, pp. 1–7. doi: 10.1109/SINKHROINFO.2017.7997514.

[11] A. Zeedan and T. Khattab, “A Critical Review of Baseband Architectures for CubeSats Communication Systems,” arXiv, arXiv:2201.09748, Jan. 2022. doi: 10.48550/arXiv.2201.09748.