A contribution to real-time space weather monitoring based on scintillation observations and IoT

Moisés José dos Santos Freitas (contact author)
Instituto Tecnológico de Aeronáutica - ITA
São José dos Campos, SP, Brazil
freitas@ita.br

Alison Moraes
Instituto de Aeronáutica e Espaço - IAE
São José dos Campos, SP, Brazil
aom@ita.br

Johnny Cardoso Marques
Instituto Tecnológico de Aeronáutica - ITA
São José dos Campos, SP, Brazil
johnny@ita.br

Fabiano Rodrigues
William B. Hanson Center for Space Sciences
University of Texas at Dallas – UT Dallas
Richardson, TX
fabiano@utdallas.edu

Abstract

The ionosphere is a region of plasma in altitudes extending up to 1000 km. Over low latitudes, steep depletions in the ionospheric density, commonly referred to as equatorial plasma bubbles, are generated after sunset as a result of plasma instabilities. Large variations in the index of refraction associated with these plasma bubbles affect the amplitude and phase of trans-ionopsheric radio signals used for communication, remote sensing and navigation, causing the so-called ionospheric scintillation. Therefore, better understanding of the ionospheric conditions is important for various applications. Unfortunately, real-time ionospheric monitoring systems have been limited, in most cases, by the cost and distribution of adequate sensors. In this work, we present and discuss an internet-of-things (IoT) system composed of a mobile application that acquires data from a network of low-cost scintillation monitors (ScintPi) capable of detecting the occurrence of ionospheric irregularities. The system, referred to as Ionik2, provides real-time information about ionospheric scintillation. A mobile app named ScintApp provides time-series of scintillation indices (S_4) and spatial distribution of scintillation using Google Maps. ScintApp also has post-processing capability allowing database queries. Finally, ScintApp is based on a native mobile smartphone application for Android operating systems. Here, we present initial results obtained with an interim prototype distribution of ScintPi monitors. We compare the information provided by ScintApp with observations made by independent instrumentation that have been widely used for ionospheric studies but do not provide real-time data. The results proof the concept of a system capable of providing realt-time information about scintillation events associated with equatorial plasma bubbles.

Keywords: internet-of-things, ionospheric scintillation, equatorial plasma bubbles

1. INTRODUCTION

The ionosphere is a region characterized by a relatively large density of free electrons and ions created, mainly, by solar photoionization (Kelley, 2009). In this layer, instabilities and irregularities over a broad range of scale sizes can develop (Abdu, 2019). These irregularities cause significant temporal and spatial variations in the index of refraction along the ionospheric altitudes. As a consequence, radio waves transmitted by satellites, including those of Global Navigation Satellite Systems (GNSS) suffer signal fadings observed by ground and space-based receivers. The fadings associated with ionospheric irregularities is commonly referred to as ionospheric scintillation (Kintner et al., 2004). Ionospheric scintillation causes cycle slips and, in extreme cases, losses of lock. Therefore, scintillation can deteriorate phase and code pseudorange observables, degrading positioning and navigation.

Fundamental and applied studies of the low-latitude ionosphere could benefit from real-time or near real-time information about the occurrence of ionospheric irregularities. This is particularly true for distributed observations. Implementation of such an observational platform is complex, in most part because of the relatively high cost of most ionospheric sensors. In South America, scintillation monitoring networks have been used for more than a decade. The following networks are worth mentioning, the Low-Latitude Ionospheric Sensor Network (LISN) from Valladares and Chau (2012) and the CIGALA/CALIBRA Project (Vani et al., 2017). These networks are essential for recording the occurrence of equatorial plasma bubbles in total electron content (TEC) measurements and ionospheric scintillation in GNSS signals. These recordings allow the characterization and modeling studies which can help to describe and be used to minimize the effects of the ionosphere on GNSS users.

Internet of Things (IoT) is a paradigm that connects real-world objects to the Internet, allowing objects to collect, process, and communicate data without human intervention (Gubbi et al, 2013). The IoT's vision is to create a better world for humans, where objects (refers to physical objects, the terms object, device, entity, and things are used interchangeably) around us can comprehend our preferences and likeness to act appropriately without explicit instructions (Perera et. al, 2014). The rapid advancements in low-cost sensor manufacturing, communication protocols, embedded systems, actuators, and hardware miniaturization have contributed to the exponential growth of the IoT. Recent advances in smartphones, wireless communication technologies, accessible hardware platforms, associated with the use of cloud computing services, have enabled the development of low-cost applications based on the Internet of Things (IoT), with the capacity to meet the demands of scalability and real-time query processing (Pattar, 2018).

IoT systems assume that devices have access to the internet and use it to collect and exchange data with other devices connected to the network. This requires an infrastructure capable of automatically collecting, distributing, and manipulating large volumes of data as well as quickly aggregating them (Gantz, 2012). Cloud computing services offer a reduction in processing cost, providing better representation and management of data. For sensor data to become usable, it must be analyzed, interpreted, and understood. Since the real-time processing requirements need to be met, eliminating the latency of responses to requests, a strategy that has been adopted is the decentralization of processing in the cloud, taking advantage of the computing power of devices closer to the end-user, such as smartphones, which configures the Edge Computing paradigm (Bonomi, 2014). In addition to reducing network delays, it reduces processing and storage costs and prevents confidential data from leaving the local network (Bonomi, 2014).

IoT is revolutionizing the aerospace industry, both on the ground and in the air. Real-time analytics via IoT are already pushing improvements in quality and manufacturing efficiency in

this sector. For example, IoT-enabled power meters can provide information on energy usage in aircraft production, which could lead to significant cost reductions and a more sustainable operation. According to Airbus, advanced analytics algorithms analyze the energy usage and suggest energy-saving measures which could result in 20% cost saving (Raad, 2021).

Previous studies have proposed architectures focused on the use of low-cost open-source hardware such as Arduino and Raspberry Pi. Monitors like ScintPi, using Raspberry Pi, by Rodrigues and Moraes (2019) and Ionik, using Arduino, from Vani et al. (2021) provide an opportunity for the proposals of platforms capable of distributing near real-time information about ionospheric irregularities and scintillation over a wide range of longitudes and latitudes.

Here, we present the current development of the Ionik2 system whose context diagram is shown in Figure 1. In the Ionik2 system, ScintPi is the sensor for ionospheric scintillation monitoring whose data is connected to cloud computing services. Mobile devices can have real-time access to ScintPi data in the cloud through a mobile app, the ScintApp.

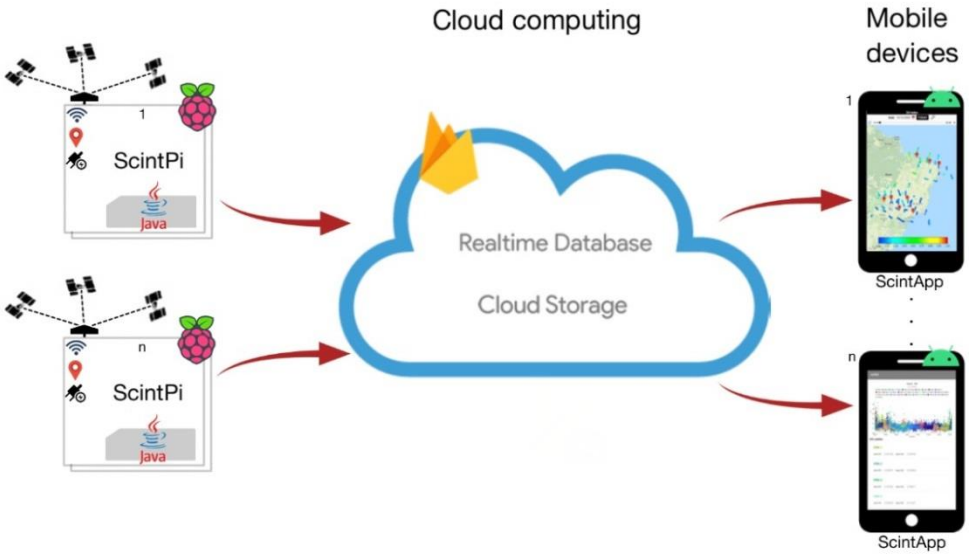


Figure 1: Context diagram of the Ionik2 system formed by ScintPi as scintillation sensors, cloud computing and the ScintApp running on mobile devices.

This article is organized as follows: Section 2 provides a description of the scintillation sensor. It starts by describing ScintPi, which is the source of observations for Ionik2 system. It also describes the functionalities of the firmware, adapted within ScintPi to acquire and send data to a cloud storage service. Then information about the software architecture of Ionik2 system is presented. In Section 3, we introduce the functionalities of the Android mobile software (ScintApp) to visualize ionospheric scintillation data in real-time. Section 4 present results of a prototype system and a comparison with independent observations of large-scale ionospheric structures. Finally, Section 5 summarizes the work and provides concluding remarks.

2. SCINTILLATION SENSOR

The work of Vani et al. (2021) showed the development of Ionik, the first version of a scintillation monitor, which consisted of an Adafruit Ultimate Global Positioning System (GPS) Breakout connected to an Arduino Uno board and that recorded the scintillation data on a memory card. Later, ScintPi of Rodrigues and Moraes (2019) was developed, being an evolution of Ionik, using a Raspberry Pi with network connection features. More importantly,

ScintPi proved that such sensors could be used for low-cost, long-term and uninterrupted monitoring of ionospheric scintillation and space weather conditions. Like commercial scintillation monitors, ScintPi has network compatibility through Internet Protocol (IP), but was not designed to provide data in real time, to serve multiple users simultaneously, to be accessed anywhere through mobile devices or to be scaled up with the inclusion of other types of sensors. These features were the design goals for the Ionik2 system, which will be described in this section with details regarding the real-time ionospheric scintillation acquisition instrumentation with Internet of Things (IoT) and cloud computing features.

2.1 ScintPi

As mentioned earlier, the observational platform of this conceptual network is the ScintPi monitor developed by Rodrigues and Moraes (2019). ScintPi is an ionospheric scintillation monitor based on single-frequency GPS-L1 (1575.42 MHz), the Adafruit Ultimate GPS Breakout and a single-board computer, Raspberry Pi 3 (Upton and Halfacree, 2014), connected via a serial to USB converter as illustrated in Figure 2.

ScintPi is composed of off-the-shelf parts with a total cost of approximately US$100. As shown by Rodrigues and Moraes (2019), ScintPi can provide useful information about ionospheric irregularities, including monitoring its generation, development, and decay. Such a platform can also serve to disseminate information about space sciences and engage students in Science, Technology, Engineering and Mathematics (STEM) education, as in other initiatives such as Watanabe et al. (2020), Yamanoor and Yamanoor (2017) and Zhong & Liang (2016).

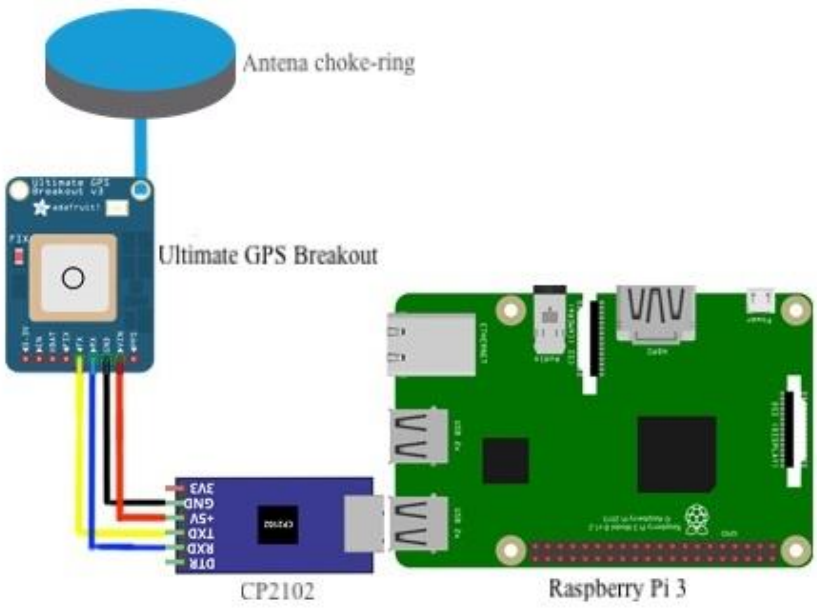


Figure 2: Block diagram of ScintPi platform for ionospheric irregularity monitoring.

The parameter used to identify the occurrence of ionospheric irregularities and scintillation is the S_4 index, which is widely used in ionospheric scintillation studies. This index is given by std(I)/<I>, where I is the intensity of the received signal, std() represents standard deviation, and <> represents average (Yeh and Liu, 1982). The S_4 index is computed every 60 seconds. The intensity values are estimated from the SNR output provided by the ScintPi platform, with $I=10^{SNR/10}$.

Traditionally, scintillation monitors operate with a sampling rate of 50 to 100 Hz as discussed in Paula et al. (2020). The ScintPi platform, however, operates at a sampling rate of 10 Hz. Nevertheless, Rodrigues and Moraes (2019) performed a comprehensive (year-long) comparison of measurements made by ScintPi and a commercial receiver (Septentrio PolaRx5S) that showed the adequacy of the low-cost system for S4 estimates despite the low sampling rate.

Figure 3(a) shows an example of a normalized detrended intensity measurement made by ScintPi. This example shows the received signal with several deep fades (i.e., below -15 dB). As discussed by Salles et al. (2021a), for strong scintillation (S_4>0.7) this level of fading is expected to happen between 1 and 4 times per minute. Such events are potential threats to GNSS receivers as they lead to the occurrence of cycle slips as discussed in details by Portella et al. (2021). It is important to note that this prototype is able to detect these events successfully.

Figure 3(b) shows simultaneous measurements of the Total Electron Content (TEC) made by a commercial receiver (Septentrio PolaRx5S) whose Choke-Ring antenna is located next to the lower cost antenna used by ScintPi. These measurements show significant TEC gradients during the period when scintillation can be seen in Figure 3(a).

Finally, Figure 3(c) shows a comparison of S_4 values estimated from ScintPi measurements and those from a PolaRx5S receiver. This comparison serves to show the very good agreement between the S_4 indices estimated by Septentrio and ScintPi despite the use of a lower cost antenna. We point out that the Septentrio monitor used here is part of the CIGALA/CALIBRA network, and the S_4 from this receiver was calculated with a sampling rate of 50 Hz, and the data was accessed by the ISMR Query Tools from Vani et al. (2017) for comparison purposes. The measurements shown in Figure 3 were made in the night between November 25 and 26, 2021 at Presidente Prudente, Brazil (22.12ºS, 51.40ºW, -15.9º dip latitude).

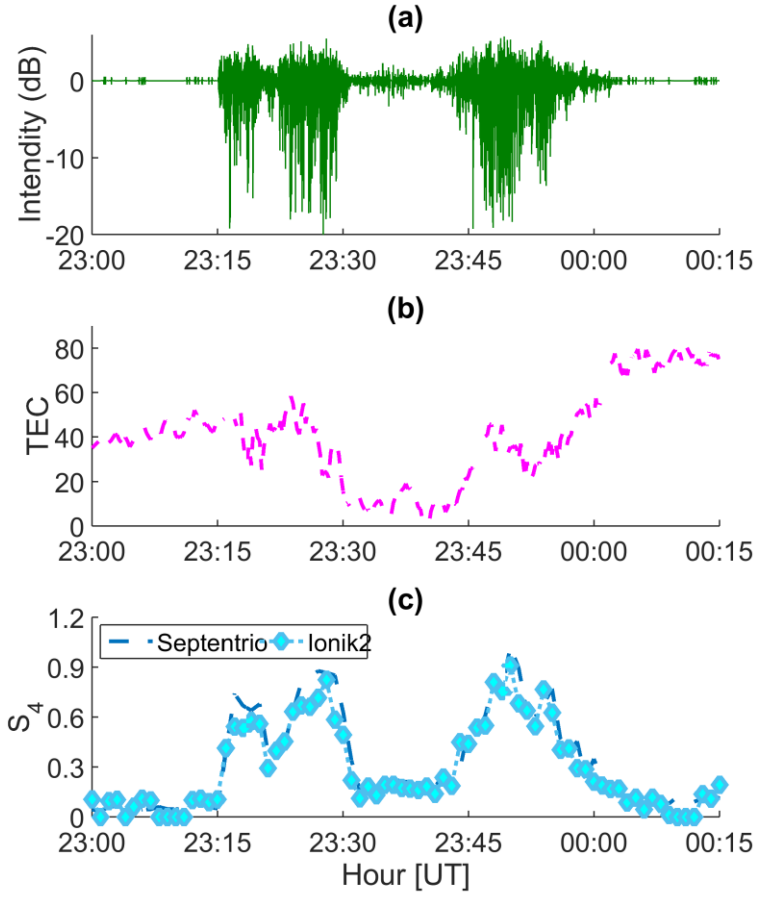


Figure 3: (a): Example of ScintPi SNR measurements of the L1 signal transmitted by the GPS satellite Pseudorandom Noise (PRN) 8. The measurements were made at Presidente Prudente, Brazil (22.12ºS, 51.40ºW) on the night between November 25 and 26, 2021. (b) Code derived slant TEC values provided by a Septentrio PolaRx5S receiver located next to the ScinPi receiver which show an equatorial plasma bubble signature (steep TEC depletions). (c) Comparison of S_4 values measured by the ScintiPi (cyan dots) and Septentrio PolaRx5S (blue dashed line).

2.2 Adaptations on ScintPi for Ionik2

ScintPi monitors have been distributed at different locations of the Brazilian territory. Measurements collected by these sensors were stored in a cloud database that can be accessed through the mobile app, ScintApp. Ionik2 provides real-time access to scintillation information. It can also provide access to information about measurements made in previous days. Figure 4 shows system architecture of the Ionik2 system. This arrangement may be described as an IoT based space weather monitoring network.

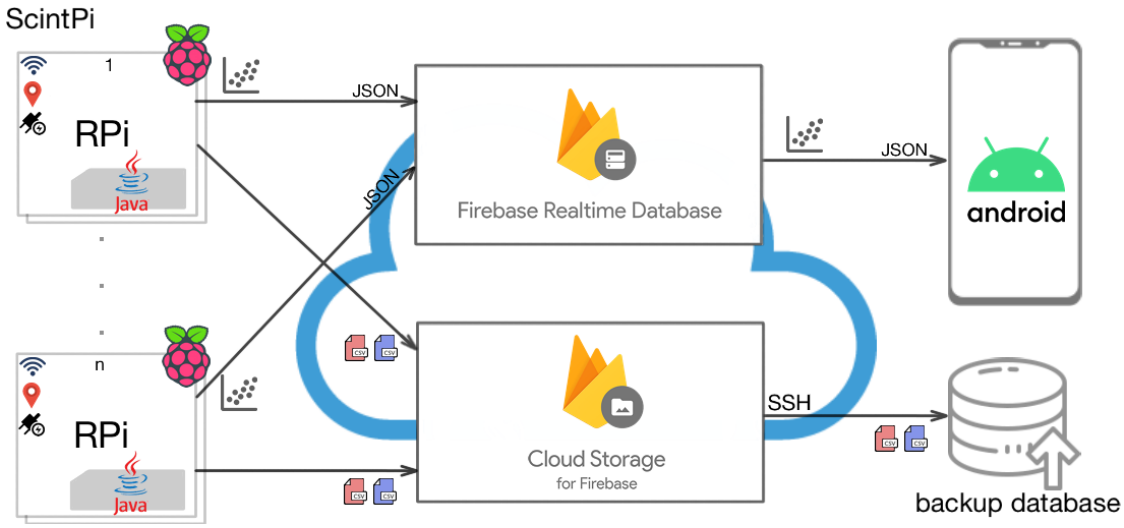


Figure 4: Architecture of the space weather monitoring network named Ionik2.

As part of Ionik2, each ScintPi monitor acquires raw data (SNR values) from the satellites in view, processes this information locally and computes the S_4 every 60 seconds. Additionally, ScintPi also provides time, azimuth and elevation from each satellite. The information collected by ScintPi follows the format described in Table 1. Finally, the raw data are also recorded in the format presented in Table 2.

Table 1: Example of information in the summary file.

Hour [UT]	Minute	Day	Month	Year	PRN	S_4	Elevation [º]	Azimuth [º]	Latitude [º]	Longitude [º]
18	12	01	01	2021	23	0.10430056	44	159	-18.638742	-48.19694
18	12	01	01	2021	29	0.0028517093	41	156	-18.638742	-48.19694

Table 2: Example of information in the recorded raw data file.

Hour [UT]	Minute	Second	Month	Day	Year	PRN	SNR	Azimuth [°]	Elevation [°]	Latitude [°]	Longitude [°]	Height [km]
16	29	53.500	11	25	01	26	44	246	46	-18.638742	-48.19694	0.7
16	29	53.100	11	25	01	25	28	13	42	-18.638742	-48.19694	0.7

Next, Ionik2 integrates the data from ScintPi monitors into a cloud database service. The connection between ScintPi and users is established through Firebase, a multi-service from Google (Moroney, 2017). The following Firebase services were used: Realtime Database which provides low latency database access; and Cloud Storage which allows file uploads and downloads. In order to be incorporated into the Ionik2 system, a ScintPi monitor needs to be authenticated by the Firebase server.

As shown in Figure 4, the ScintPi monitors that are part of the network are connected to the Firebase Realtime Database, where S_4, elevation, azimuth and time parameters are sent in a JavaScript Object Notation (JSON) format. The ScintPi monitors also send raw data and summary information (S_4, elevation, azimuth, etc.) to Cloud Storage. Users with ScintApp installed in their Android devices, can access Realtime Database and obtain graphical representations of their queries. It is also possible to download the backup raw database through Secure Socket Shell (SSH) connection, but currently this option is only valid for registered members.

The implementation of ScintPi embedded software was in Java. Figure 5 presents the algorithm composed of 13 steps. During step 1, the program waits for the receiver to track the GPS satellites. In the step 2, the program performs authentication with Realtime Database and Cloud Storage, both Firebase services. Consequently, the serial port is opened at step 3. With the serial port open, the program starts receiving data from the GPS receiver (NMEA sentences) at step 4. The NMEA sentences can start with 19 different types of prefixes (http://aprs.gids.nl/nmea/), but the only ones that are useful are the types: $GPGSV, $GPRMC and $GPGGA. If the sentence received is different of these aforementioned types the program waits to receive the next sentence in step 4.

All measurements of ScintPi must be time-indexed, with date and time (yyyy-MM-dd HH:mm:ss.sss). For the time to be uniform across all network monitors, it is obtained from the GPS message in step 6, based on the NMEA messages with prefix $GPRMC. When receiving the first sentences, if $GPRMC messages are not available yet, these measurements are rejected, until a time measurement is available. At step 5, the received sentences starting with $GPGSV, containing the PRN, elevation, azimuth and SNR of the viewed satellites are acquired. As previously mentioned, if the sentence starts with $GPRMC, then the receiver latitude, longitude and time are updated (step 6). If the sentence begins with $GPGGA, the receiver altitude is acquired (step 7). Steps 5, 6 and 7 are performed continuously, with the GPS time value updated as available.

At step 8, with each new $GPGSV sentence, the program saves the raw data (PRN, SNR, elevation, azimuth, latitude, longitude and altitude, indexed by time) in a text file. At step 11, the S_4 is calculated using the SNR data from $GPGSV sentences, based on the acquisition of one minute of data. The program stores the computed S_4 in a summary text file. This summary file is composed of PRN, S_4, elevation, azimuth, latitude, longitude, altitude and is indexed by time data for each satellite, as part of step 12. Following, the program sends the summary file

data to the Realtime Database in step 13. In case internet connection is momentarily not available after this data processing, ScintPi keeps the data locally on a memory card, and only sends it when the connection is reestablished.

Upon receiving a new $GPGSV sentence, the program checks the current date on the information. if the day changes from the date previously received, the program now operates on a new date. Thus, text files with summary and raw data are immediately closed and compressed to take up less space in memory, at step 9. Typically, raw files have more than 5.5 million lines, corresponding to 500 Mbytes size file; before being uploaded, the file is compressed to an 18 Mbytes size, approximately. Summary files have approximately 2 Mbytes and 240 Kbytes before and after compression, respectively. Once the text files have been closed and compressed, the program uploads them to Cloud Storage (step 10).

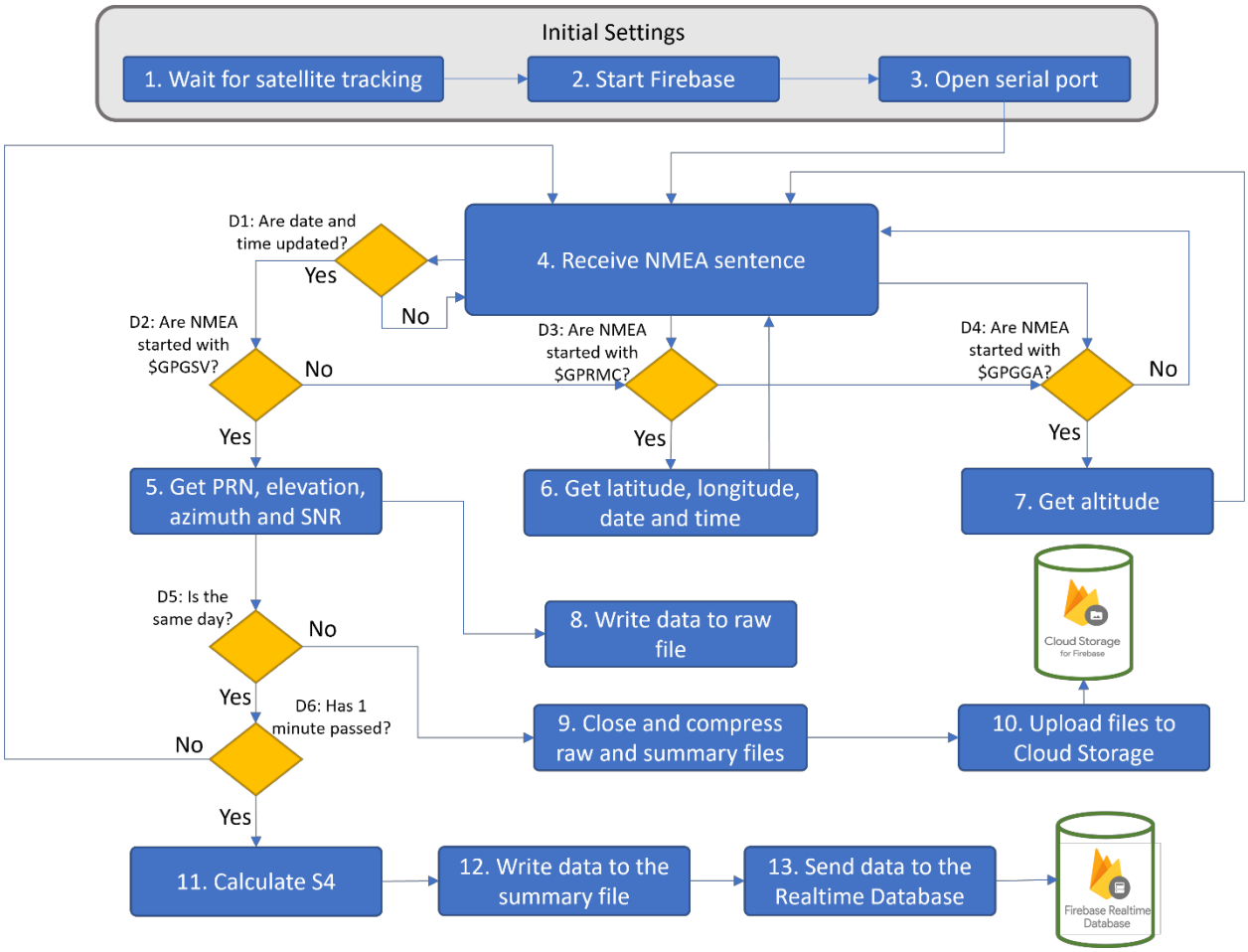


Figure 5: Algorithm embedded in ScintPi to operate on the Ionik2 network.

Before providing more information about the design of the app, we provide additional details about the scintillation data. We provide statistics of scintillation (S4) data that was collected between November 22 and 28 of 2021. Despite a small dataset, it provides a reference for comparison between the proposed system and traditional geophysical instrumentation. These data sets were acquired by two ScintPi, deployed at São José dos Campos (23.2°S, 45.85°W) and Presidente Prudente (22.1°S, 51.4°W) in Brazil. Both sites had Septentrio scintillation monitors, that integrate INCT NavAer project (Monico et al., 2022), for comparison purposes. Figure 6 shows the Complementary Cumulative Distribution Function (CCDF) of S_4

occurrences for both instrumentations, as described in Salles et al. (2021b), for the time between 20:00 and 23:59 LT. It is possible to observe the good agreement between the data, for example, considering the threshold of $S_4 \geq 0.7$ for Presidente Prudente, the probabilities are 1.61 and 1.92 %, respectively, for the measurements of Septentrio and Ionik2. Similarly, considering the same threshold, the values for São José dos Campos were, respectively, 0.77 and 0.72%. This comparison serves to show that the sensors of Ionik2 were properly installed and providing good proxies of scintillation activity. The values found in these analyzes are lower than the values reported for example in Salles et al (2021b), however considering the low solar flux condition during the presented measurements, the percentages between 1.92% and 0.72% might be threatening to critical users such as augmentation systems, as discussed in Figure 10 of Sousasantos et al (2021).

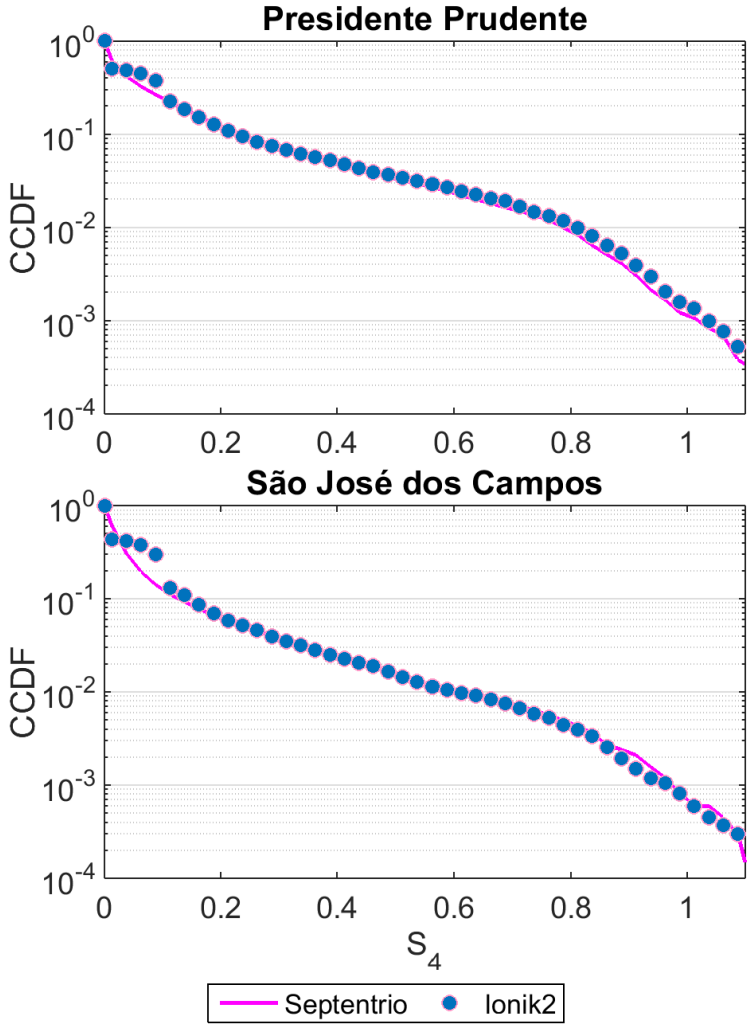


Figure 6: Statistical validation of measurements through the CCDF of S_4 for the period between November 22 and 28 of 2021 at Presidente Prudente and São José dos Campos considering the Ionik2 and Septentrio for the hours between 20:00 and 23:59 LT.

3 SCINTAPP

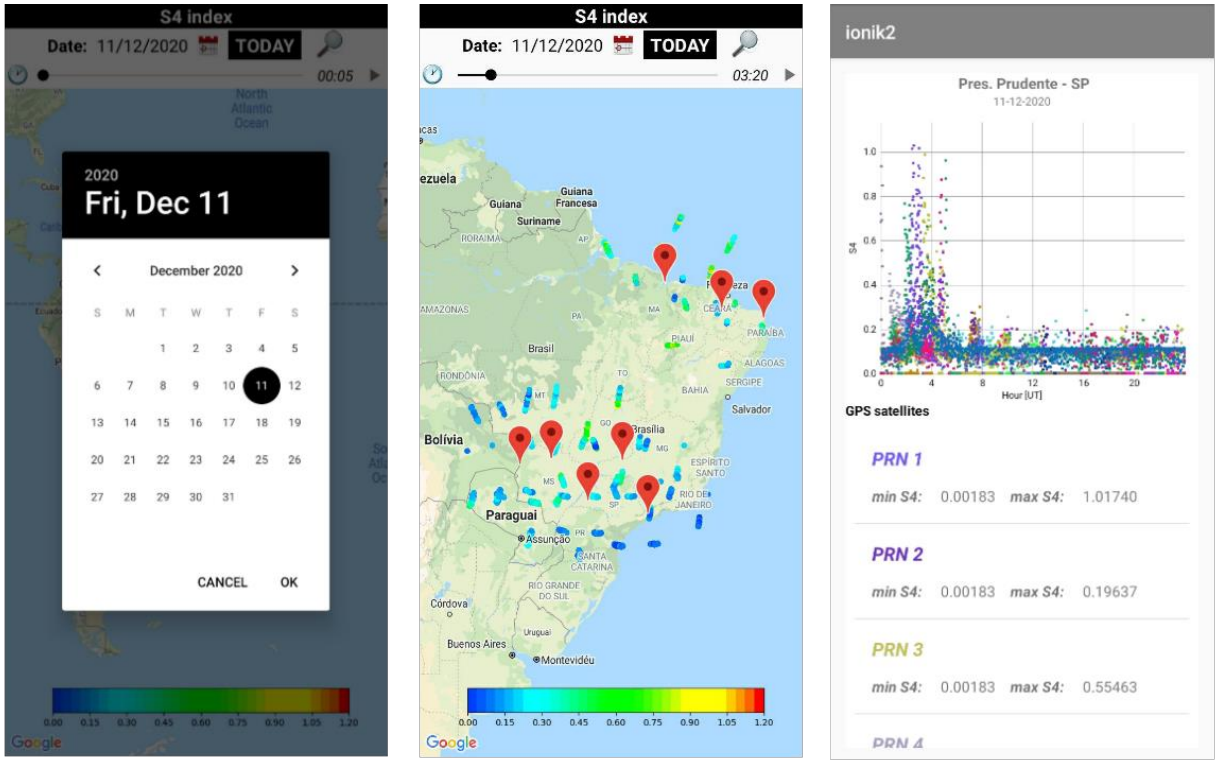
The objective of the Ionik2 system is to make scintillation data, more specifically S_4 indices, available for analysis and visualization through Android devices for multiple users by the mobile app ScintApp. The development of ScintApp used IDE (Integrated Development

Environment) Android Studio for programming with Java language, and Firebase services to assist in receiving notifications and queries to the database.

The first information required from the user to use ScintApp is the date of interest as illustrated in Figure 7a. As an example, after choosing December 11, 2020 as the day of interest, ScintApp provides the screen shown Figure 7b. The screen shows the stations locations (red marks) and ionospheric piercing points (IPPs) at 350 km altitude for the observed signals (colored dots). The IPP is the assumed point in space at which the signal traspasses the ionospheric layer, modeled as a thin shell in this simplified geometry.

Scintillation severity is represented by the color bar shown at the bottom of the screen. By clicking on the station of interest, the user will be presented with the screen shown in Figure 7c, which provides the S_4 index data for the chosen location and date. This screen shows S_4 indices for all the satellites with elevation angle > 20º, such value was chosen to reduce interference effects, mainly multipath. It also shows the maximum and minimum S_4 values for each satellite. The user can request data from a specific satellite. The request will be presented as a time series plot like the one shown in Figure 7d. Figure 7d shows data from PRN49, which is an SBAS satellite, in a case of a strong scintillation event occurring between 02:00 UT and 04:00 UT.

(a) (b) (c)



(d)

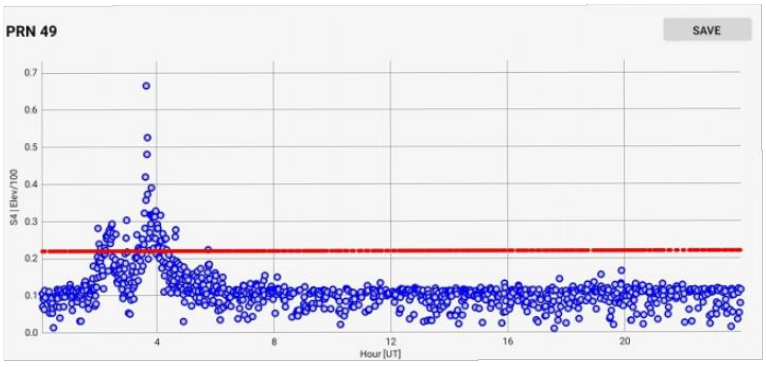


Figure 7: The ScintApp. (a) The first screen where the user may select query for current or past data. (b) The map with the available stations, IPP for the received GPS signals with the bar indicating the strength of scintillation. (c) The S_4 indexes computed for a single station. (d) The S_4 for a specific link queried by the user.

3.1 Functional Requirements

Functional Requirements (FRs) represent the functionalities of the system without which, the system itself is not useful. These functionalities must be fulfilled and exceptions cannot be granted or resorted during the project life cycle. Core functionality addresses the performance of a set of processes. The main purpose of software development is to fulfill this core functionality **(**Gilb, 2005**).**

In the development of ScintApp, 11 FRs were identified, as presented in Table 3. These FRs allow the search of active stations (FR01), determining the pairs (city, state) of each station (FR02), calculate latitude (φ_r) and longitude (λ_r) of ionospheric piercing points (FR04), and then display these points on the map following the color scale of S_4 and the time selected in the slider.

Table 3 – Functional Requirements for ScintApp

Requirement Tag	Requirement Description
FR01	The ScintApp shall search for the active stations on a specific date and add their bookmarks to the map in their respective locations.
FR02	The ScintApp shall determine the pairs (city, state) of each station.
FR03	The ScintApp shall seek all measurements taken by a station on a specific date.
FR04	The ScintApp shall calculate the location - latitude (φ_r) and longitude (λ_r) of the ionospheric piercing points.
FR05	When selecting the marker of a station, the ScintApp shall open the screen that contains the S_4 chart due to all satellites and the list of observed satellites.
FR06	The ScintApp shall receive all satellites observed per station on a specific date and add them to a list.
FR07	The ScintApp shall fetch in all measurements performed by all satellites seen in the station the S_4 scintillation index and plot on a chart.
FR08	The ScintApp shall determine the maximum and minimum values of S_4 over time on each satellite on the specified date.
FR09	When selecting a satellite from the list, the ScintApp shall open the screen that displays the S_4 graph and elevation.
FR10	The ScintApp shall display the S_4 and elevation graph.
FR11	When you press the Save button, the ScintApp shall save the measurements to a text file identified with the satellite ID, date, and station ID.

3.2 Data Requirements

The Data Requirements (DR) definition establishes the process used to identify, prioritize, precisely formulate, and validate the data needed to achieve application objectives (Fernandes and Machado, 2016). For ScintApp, 9 DRs were identified, as shown in Table 4. The DRs are traceable to the Functional Requirements presented in Table 3.

Table 4 – Data Requirements for ScintApp

Requirement Tag	Requirement Description	Traceability to Functional Requirements
DR01	The ScintApp shall allow the user to select a date in the calendar.	FR01
DR02	The ScintApp shall add a station with the following attributes: latitude, longitude, and ID.	FR01
DR03	The ScintApp shall add a pair (city, state) in each station.	FR02
DR04	The ScintApp shall add a measure with the following attributes: azimuth, elevation, S_4, and time.	FR03, FR07, FR10, FR11
DR05	The ScintApp shall add ionospheric points to the map.	FR04
DR06	The ScintApp shall transmit the station ID and date to the screen containing the S_4 chart due to all satellites and the list of observed satellites.	FR05
DR07	The ScintApp shall add a satellite using the satellite ID.	FR06
DR08	The ScintApp shall add the maximum and minimum values of S_4 to its respective satellite from the list.	FR08
DR09	The ScintApp shall transmit the date, station ID, and satellite ID to the screen that displays the S_4 graph and elevation.	FR09

3.3 Non-Functional Requirements

A Non-Functional Requirement (NFR) corresponds to a set of restrictions imposed on the system to be developed, establishing, for instance, how attractive, useful, fast, or reliable it is. Classic examples of this types of requirements are time constraints, restrictions in the development process or adoption of standards **(**Dick et al., 2017). In this development, 2 NFRs were identified as presented in Table 5. The ScintApp shall read the database with Read-Only privileges (NFR01) and provide the ability to track measurements on all active stations in real time (NFR02).

Table 5 – Non-Functional Requirements for ScintApp

Requirement Tag	Requirement Description
NFR01	The ScintApp shall read the database with Read-Only privileges

NFR02	The ScintApp shall provide the ability to track measurements on all active stations in real time

3.4 ScintApp algorithm

Figure 8 shows the schematic representation of the ScintApp algorithm with 8 main steps. Steps 1 to 4 occur on the screens shown in Figures 7a and 7b; steps 5 and 6 appear on the screen shown in Figure 7c; finally, steps 7 and 8 occur on the screen shown in Figure 7d. In step 1, the software loads the home screen and adds an interactive map using the Google Maps for Android service. This service downloads and displays map blocks in response to movement and zoom gestures made by the user and allows the insertion of markers of different shapes and behaviors. As shown in Figure 7a, when the user selects the calendar icon at the top of the screen, the user may choose a date, present or past, for querying the S_4 index. Optionally, pressing the Today button selects the current date. Pressing the search button (magnifying glass), the ScintApp requests the Realtime Database, providing the date chosen as a parameter. The response as a JSON contains the latitudes, longitudes, and identifiers of the monitoring stations active on that date.

In step 2, red markers are inserted at the active station locations, as shown in Figure 7b. The program obtains the cities and states using the latitudes and longitudes of each station, where the stations are located, as titles of the respective markers. This step serves to fulfill the functional requirements FR01 and FR02, as shown in Table 3. After that, ScintApp automatically performs a request in the Realtime Database for each active station for the selected date to obtain time series with S_4, elevation, and azimuth as a return. In step 3, the program uses these time series to calculate the location (latitude and longitude) of the IPPs, thus fulfilling the functional requirements FR03 and FR04. By pressing the Play button (step 4), the insertion of circular markers at the location of the IPPs occurs using a color scheme weighted by the values of S_4. According to the scale shown at the bottom of the screen, a 20-minutes window moves in time with increments of 5 minutes, as shown in Figure 7b. This visualization of IPPs is essential because it presents the temporal and spatial distribution of S_4 values and, consequently, the location of ionospheric irregularities crossed by the GPS signals.

In the transition from step 2 to step 5, when a station is selected, ScintApp requests the Realtime Database, passing as parameters the station identifier and the selected date. A JSON response with the observed satellites occurs, adding a list, as required by FR05 and FR06, according to the presented in Figure 7c. Automatically, the program executes a request to the Realtime Database for each satellite observed in the station, now passing as parameters the date, the station identifier, and the satellite identifier, obtaining time series with S_4, elevation, and azimuth as a response. At this point, the program determines the maximum and minimum values of S_4 calculated from each satellite and adds them to the satellite list, fulfilling the FR08 requirement. The program uses the time series to generate a graph with all the estimated S_4 in the range of a maximum value reached in the chosen night (step 6). Each S_4 measurements available have a trace in a different color, according to the satellite, as required by the FR07 requirement.

The algorithm makes a transition from step 5 to step 7 with a selection of a satellite from the list. A screen change, as presented by Figure 7d, occurs. This transition provides the fulfilment of the requirements FR09 and FR10. The program makes one more request to the Realtime Database to generate the graphic, providing the date, the station identifier, and the satellite identifier. By pressing the Save button, the application saves the measurements in a text file containing the time, S_4, azimuth, and elevation, as per the FR11 requirement.

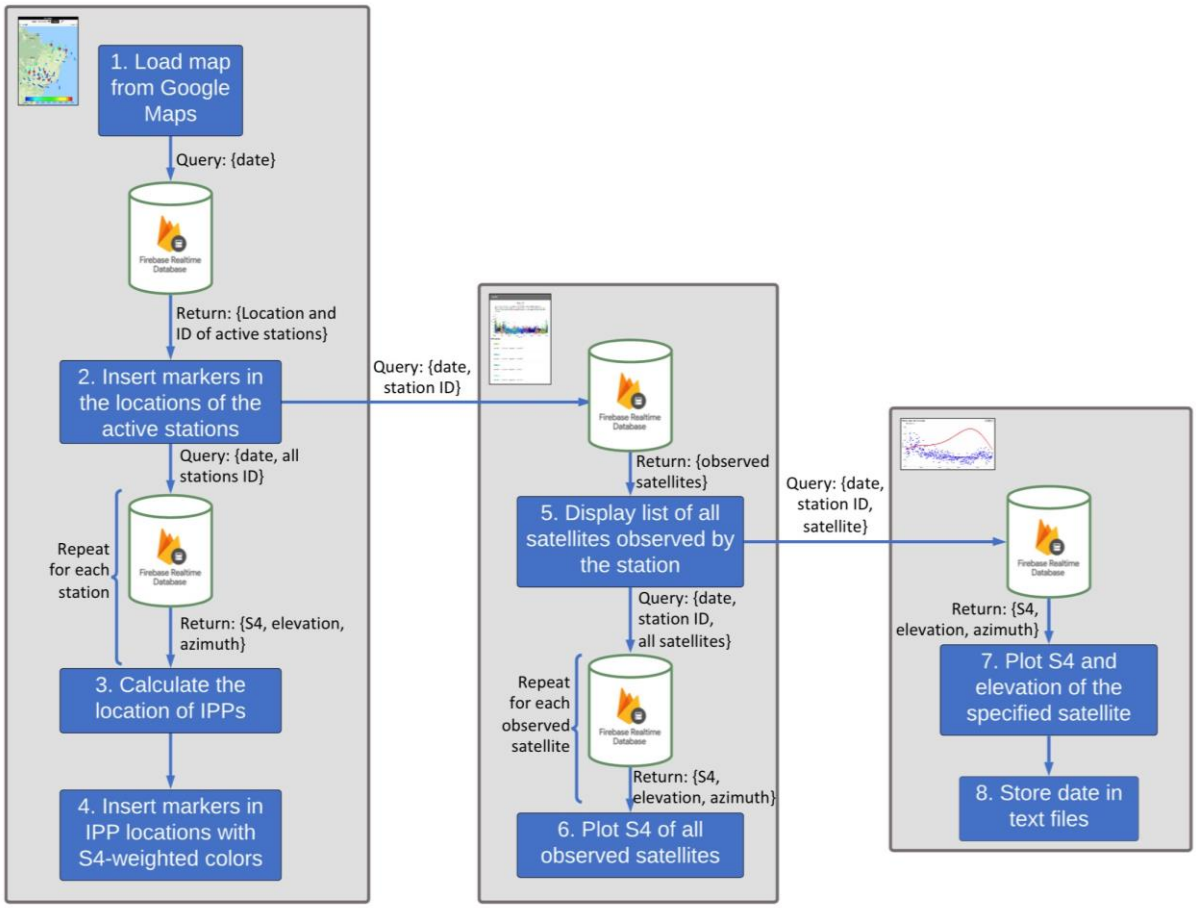


Figure 8: Algorithm of the ScintApp.

4. ON THE USE OF THE APPLICATION

This section illustrates the potential application of ScintApp. We show, more specifically, its ability to monitor the occurrence of ionospheric scintillation and plasma bubbles over the Brazilian sector with a considerable area coverage based on Ionik2. These examples use data acquired on an experimental basis, from the nights of October 23 and November 26, 2021 recorded by a prototype of the system with data from only four ScintPi monitors, provisionally deployed for this purpose. These monitors were installed at Corumbá, Araguari, Presidente Prudente, and São José dos Campos. The coordinates for these stations during the tests are shown in Table 6. In general, the monitors were hosted by universities and research institutes. Table 6 provides the details about the locations. Additionally, Figure 9 shows one example of Ionik2 prototype currently installed at Instituto Tecnológico de Aeronáutica in São José dos Campos.

Figure 9: Android mobile device running ScintApp next to the IoT based scintillation monitor protype Ionik2 installed at Instituto Tecnológico de Aeronáutica in São José dos Campos (23.209ºS, 45.874ºW).

Table 6: Location of Ionik2 receivers .

Station	Latitude	Longitude	Dip Latitude
São José dos Campos – SP	23.207ºS	45.859ºW	-19.59º
Presidente Prudente – SP	22.120ºS	51.408ºW	-15.92º
Corumbá – MS	19.027ºS	57.654ºW	-10.56º
Araguari – MG	18.638ºS	48.197ºW	-14.83º

Figure 10 shows time series of S_4 values provided by the ScintApp for the nights of October 23 (left column) and November 26, 2021 (right column). ScintApp allows users to track the behavior of scintillation over Brazil. In both cases, the time series indicate that strong scintillations occurred on these nights from 00:00 UT up to 04:30 UT, typical period of Equatorial Plasma Bubble (EPB) occurrences in this sector (Sobral et al. 2002).

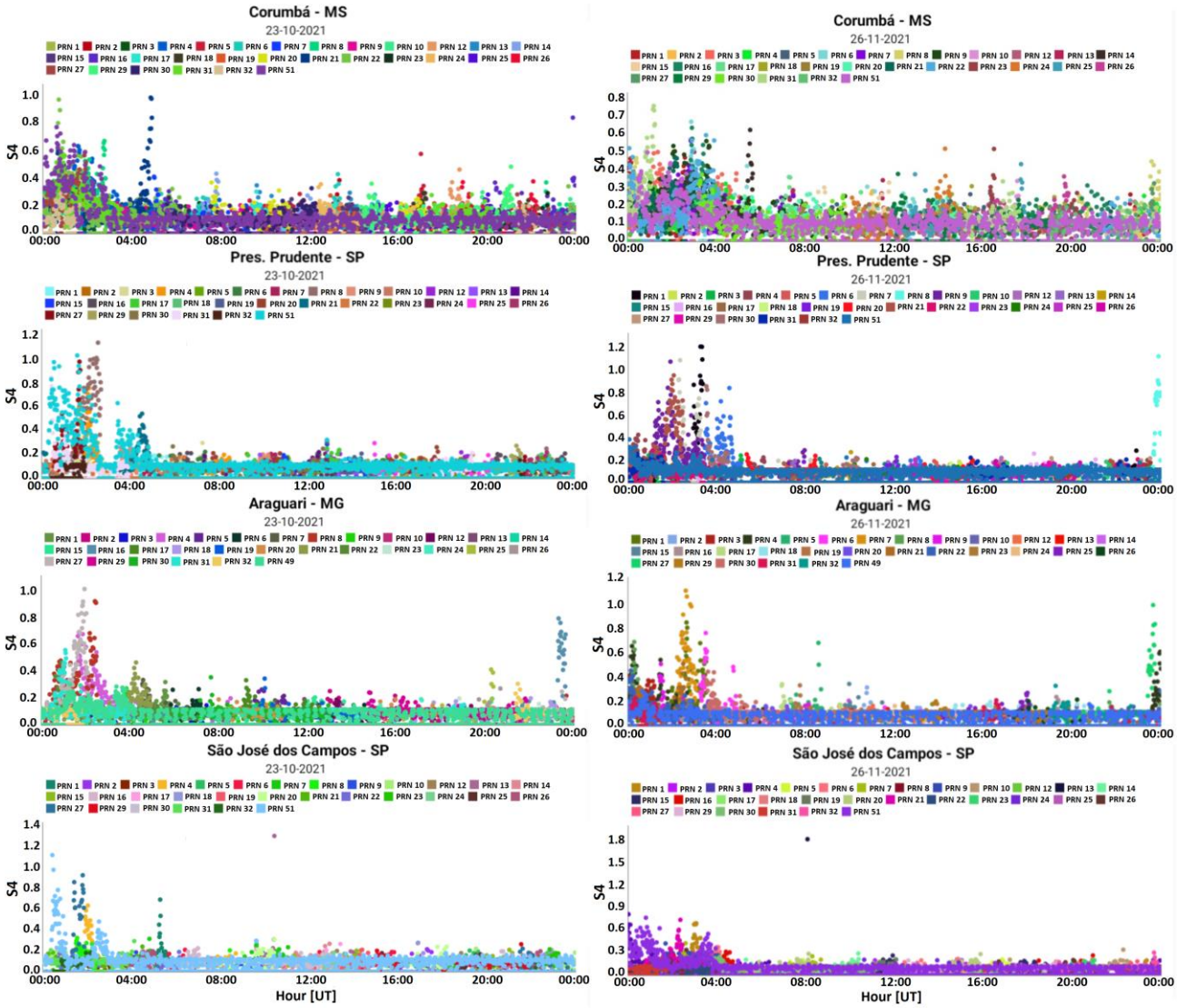


Figure 10: S_4 values for the nights of October 23 (left) and November 26 (right), 2021 as showed by ScintiApp, considering all GPS satellites above 20º elevation tracked by Ionik2 monitors located at Corumbá, Araguari, Presidente Prudente and São José dos Campos.

Continuing to explore the features of ScintiApp, the panels on the left of Figure 11 show the IPPs of the signals monitored by the system for 01:45 on October 23, 2021 and 0:15 UT on November 26, 2021, respectively. Some of the scintillation shown in Figure 10 can also be seen on the IPPs shown in Figure 11 IPPs of this panel for several satellites.

Assisting with cross-validation, the right panels of Figure 11 show the results of nearly simultaneous measurements made by dual-frequency geodetic GPS receivers distributed over Brazil. While these receivers cannot be used to make high-rate measurements of signal intensity and scintillation, they allow us to create a map of the ionospheric TEC over the Brazilian region using post-processing routines. More information about these maps and how they are created may be found in Oliveira et al. (2020). The TEC maps on the right-side panels of Figure 11 correspond to the scenario at the respective hours shown in the ScintApp screen in the left panels. The color bar at the right represents the TEC scale in TECU (1 TECU = $1x10^{16}$ electrons/m^2). The IPP tracks imported from ScintApp with the estimated values of S_4 for the latest 30 minutes centered in the selected time were superposed above the TEC map for comparison. The S_4 values are color-coded according to the horizontal bar at the top of the panel.

The TEC maps show an enhancement located a few degrees of and nearly parallel to the magnetic equator (red line). This TEC enhancement is the equatorial ionization anomaly (Appleton, 1946). More importantly, the TEC maps show at least two (three) main depleted structures on October 23 (November 26). These structures are approximately perpendicular to the magnetic equator in the longitude range between -60º and -45º. Across these regions of severe TEC gradients the scintillation reported by the ScintApp exhibits the largest values, hence indicating that the ScintApp can indeed serve as a functional indicator of plasma bubble occurrence. It is worth mentioning that the IPPs outside the equatorial ionization anomaly do not show considerable increments in the S_4 level.

These examples serve as a proof of concept of a system with the potential to alert GNSS users of scintillation conditions in real-time. Such a system could, for instance, generate notifications about potential availability threats that ionospheric irregularities and scintillation may cause. This type of service can be valid for users such as those in civil aviation, precision agriculture and offshore oil and gas extraction. It is important to mention that the performance of such a system would be directly related to the number of scintillation sensors used in the network. Finally, it is worth mentioning that the prototype system presented here operated in GPS-L1 only and substantial benefits would be obtained if the system operated with multi-frequency and/or multi-constellation capabilities. Data is structured in such a way that adaptation for such features would be straightforward.

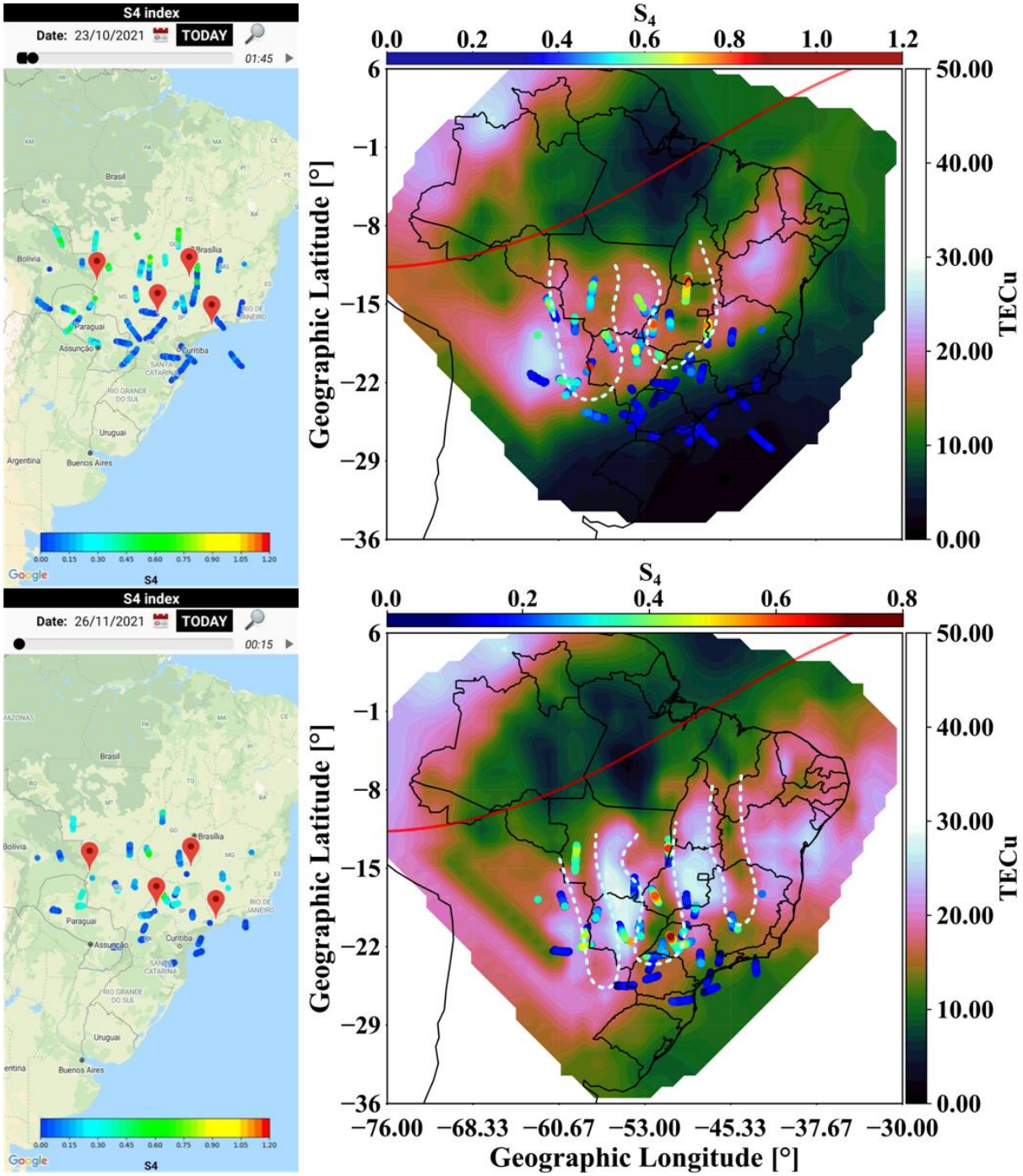


Figure 11: Scintillation maps example for October 23 (top panels) and November 26 (bottom panels), 2021 at 01:45 and 0:15 UT, respectively. Left panels: ScintApp map screens showing the IPP for the received GPS signals with the bar indicating the level of the scintillation. Right panels: Ionospheric TEC maps showing the occurrence of depletions (equatorial plasma bubbles) and the scintillation recorded along these regions. The approximate region of EPBs are highlighted by the white dotted curves.

5. SUMMARY AND CONCLUDING REMARKS

A better understanding of the generation, development and decay of ionospheric irregularities and the impact of these irregularities in communication, navigation and remote sensing systems has been the topic of an extensive number of studies (Kintner et al., 2007; Carrano et al., 2012; Ghafoori and Skone 2015, Bolaji et al., 2019). Systems for monitoring these irregularities in real-time or near real-time, however, have been limited. One of the main reasons is the relatively high cost of observing instrumentation. Additionally, distributed arrays of instruments may provide a wide perspective of the ionospheric environment.

Here, we take advantage of the development of low-cost scintillation monitor (ScintPi) to propose the concept of a system (Ionik2) that would allow real-time monitoring of low-latitude ionospheric irregularities. Visualization and inspection of ionospheric scintillation time series and maps have been proved possible through a mobile application (ScintApp). The first results of a prototype system (with only 4 monitors running provisionally up to now) are presented as a proof-of-concept of the system.

In this initial evaluation, the scintillation maps generated by the mobile app were compared with TEC maps. The results indicate that Ionik2 may successfully track GPS signals and monitor the occurrence of scintillation associated with ionospheric plasma bubbles. Therefore, ScintApp has the potential to contribute to observational studies of ionospheric structures by providing information about ionospheric irregularities with Fresnel scale sizes (hundreds of meters), which cause scintillation in L-band signals and affecting GNSS users.

The next step for the system would be a permanent installation of the prototypes used in this experiment and an expansion of the distribution of sensors and, therefore, an expansion of the coverage of the scintillation maps. Another point being considered is related to the fusion of this system with data from other geophysical sources in the context of cloud computing and data acquisition based on IoT instruments. Another possibility is the use of GNSS receivers on smartphones; a version based on this technology is planned for future work but will require considerations related to the non-stationarity of these sensors.

Finally, we must mention the benefit of creating such a network for undergraduate education. Students can be involved in the development and deployment of scintillation monitors, in improvements in the software for data acquisition and analyses, and comparison of the data with observations made by other types of ground-based instruments and space-based sensors. Additionally, the system can also be used in educational initiatives related to algorithms development, data structures and internet of things (IoT) technologies.

Acknowledgments: MJSF was supported by CAPES (88887.137186/2017-00/0) and and conducted this research in the framework of the INCT GNSS-NavAer grants CNPq 465648/2014-2 and FAPESP 2017/50115-0. AOM is supported by CNPq award number 314043/2018-7. FSR would like to thank support from NSF Award AGS-2122639. The Scintapp is available for download at the following link: https://doi.org/10.5281/zenodo.6466730. In this previous link it is also available all the field measurements acquired in this proof-of-concept experiment carried out in this work. The authors are grateful to Instituto Brasileiro de Geografia e Estatística (IBGE) for providing GPS data from Rede Brasileira de Monitoramento Contínuo (RBMC) for the construction of TEC map from Figure 11. The authors are grateful to I. Tsuchiya, A. Martinon, F. Antreich, A. L. Almeida Silva, R. V, Françozo, J Sousasantos, E. R. de Paula and J. F. Galera Monico for their constant support, comments and suggestions.

REFERENCES

Abdu, M. A. (2019). Day-to-day and short-term variabilities in the equatorial plasma bubble/spread F irregularity seeding and development. Progress in Earth and Planetary Science, 6(1), 1-22. https://doi.org/10.1186/s40645-019-0258-1

Appleton, E. V. (1946). Two anomalies in the ionosphere. Nature, 157(3995), 691. doi: 10.1038/157691a0

Bolaji, O. S., Adebiyi, S. J., & Fashae, J. B. (2019). Characterization of ionospheric irregularities at different longitudes during quiet and disturbed geomagnetic conditions. Journal of Atmospheric and Solar-Terrestrial Physics, 182, 93-100. https://doi.org/10.1016/j.jastp.2018.11.007

Bonomi, F., Milito R., Natarajan P., Zhu, J. (2014). Fog computing: a platform for internet of things and analytics. Big Data and Internet of Things: a roadmap for smart environments. Springer International Publishing, 169-186. https://doi.org/10.1007/978-3-319-05029-4_7

Carrano, C. S., Groves, K. M., & Caton, R. G. (2012). Simulating the impacts of ionospheric scintillation on L band SAR image formation. Radio Science, 47(04), 1-14. https://doi.org/10.1029/2011RS004956

de Oliveira, C. B. A., Espejo, T. M. S., Moraes, A., Costa, E., Sousasantos, J., Lourenço, L. F. D., &Abdu, M. A. (2020). Analysis of plasma bubble signatures in total electron content maps of the low-latitude ionosphere: A simplified methodology. *Surveys in Geophysics*, *41*(4), 897-931. https://doi.org/10.1007/s10712-020-09584-7

de Paula, E. R., Martinon, A. F., Moraes, A. O., Carrano, C., Neto, A. C., Doherty, P., ... & Slewaegen, J. M. Performance of 6 Different GNSS Receivers at Low Latitude under Moderate and Strong Scintillation. *Earth and Space Science*, e2020EA001314. https://doi.org/10.1029/2020EA001314

Dick, J., Hull, E., & Jackson, K. (2017). System Modelling for Requirements Engineering. In Requirements Engineering (pp. 57-92). Springer, Cham. https://doi.org/10.1007/978-3-319-61073-3_3

Fernandes, J. M., & Machado, R. J. (2016). Requirements in engineering projects. Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-18597-2

Gantz, J., Reinsel, D., Arend, C. (2012). The digital universe in 2020: big data bigger digital shadows, and biggest growth in the far east. IDC iView

Ghafoori, F., & Skone, S. (2015). Impact of equatorial ionospheric irregularities on GNSS receivers using real and synthetic scintillation signals. Radio Science, 50(4), 294-317. https://doi.org/10.1002/2014RS005513

Gilb, T. (2005). Competitive engineering: a handbook for systems engineering, requirements engineering, and software engineering using Planguage. Elsevier. ISBN: 9780750665070

Gubbi., J., Buyya R., Marusic, S., Palaniswami M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions, Future Gener. Comput. Syst., 29(7), 1645–1660. https://doi.org/10.1016/j.future.2013.01.010

Kelley, M. C. (2009). The Earth's ionosphere: plasma physics and electrodynamics. Academic press. ISBN: 978-0-12-088425-4

Kintner, P. M., Ledvina, B. M., De Paula, E. R., & Kantor, I. J. (2004). Size, shape, orientation, speed, and duration of GPS equatorial anomaly scintillations. Radio Science, 39(2), 1-23. https://doi.org/10.1029/2003RS002878

Kintner, P. M., Ledvina, B. M., & De Paula, E. R. (2007). GPS and ionospheric scintillations. Space weather, 5(9). https://doi.org/10.1029/2006SW000260

Monico, J. F. G., de Paula, E. R., de Oliveira Moraes, A., Costa, E., Shimabukuro, M. H., Alves, D. M. B., ... & Aguiar, C. R. (2022). The GNSS NavAer INCT Project Overview and Main Results. Journal of Aerospace Technology and Management, 14. J. Aerosp. Technol. Manag., São José dos Campos, v14, e0722, 2022 https://doi.org/10.1590/jatm.v14.1249

Moroney, L. (2017). The firebase realtime database. In The Definitive Guide to Firebase (pp. 51-71). Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4842-2943-9_3

Pattar, S., Buyya, R., Venugopal, K. R., Ivengar, S. S., Patnaik, L. M. (2018). Searching for the IoT Resources: Fundamentals, Requirements, Comprehensive Review, and Future Directions. IEEE Communications Surveys & Tutorials. Survey, 20(3), 2101-2132. https://doi.org/10.1109/COMST.2018.2825231

Perera, C., Zaslavsky, A., Christen P., Georgakopoulos D. (2014). Context aware computing for the Internet of Things: A survey. IEEE Communications Surveys & Tutorials. Survey, 16(1), 414–454. https://doi.org/10.1109/SURV.2013.042313.00197

Raad, H. K. (2021). Fundamentals of IoT and Wearable Technology Design. Wiley. ISBN: 978-1-119-61753-2

Portella, I. P., de O Moraes, A., da Silva Pinho, M., Sousasantos, J., & Rodrigues, F. (2021). Examining the tolerance of GNSS receiver phase tracking loop under the effects of severe ionospheric scintillation conditions based on its bandwidth. Radio Science, 56(6), 1-11. https://doi.org/10.1029/2020RS007160

Rodrigues, F. S., and Moraes, A. O. (2019). ScintPi: A low cost, easy to build GPS ionospheric scintillation monitor for DASI studies of space weather, education, and citizen science initiatives. *Earth and Space Science*, *6*(8), 1547-1560. https://doi.org/10.1029/2019EA000588

Salles, L. A., Moraes, A., Vani, B., Sousasantos, J., Affonso, B. J., & Monico, J. F. G. (2021a). A deep fading assessment of the modernized L2C and L5 signals for low-latitude regions. GPS Solutions, 25(3), 1-13. https://doi.org/10.1007/s10291-021-01157-4

Salles, L. A., Vani, B. C., Moraes, A., Costa, E., & de Paula, E. R. (2021b). Investigating ionospheric scintillation effects on multifrequency GPS signals. Surveys in Geophysics, 42(4), 999-1025. https://doi.org/10.1007/s10712-021-09643-7

Sousasantos, J., Marini-Pereira, L., Moraes, A. D. O., & Pullen, S. (2021). Ground-Based Augmentation System Operation in Low Latitudes-Part 2: Space Weather, Ionospheric Behavior and Challenges. Journal of Aerospace Technology and Management, 13. https://doi.org/10.1590/jatm.v13.1237

Upton, E., & Halfacree, G. (2014). Raspberry Pi user guide. John Wiley & Sons. ISBN: 978-1-119-26436-1

Sobral, J. H. A., Abdu, M. A., Takahashi, H., Taylor, M. J., De Paula, E. R., Zamlutti, C. J., & Borba, G. L. (2002). Ionospheric plasma bubble climatology over Brazil based on 22 years (1977–1998) of 630nm airglow observations. Journal of Atmospheric and Solar-Terrestrial Physics, 64(12-14), 1517-1524. https://doi.org/10.1016/S1364-6826(02)00089-5

Valladares, C. E., & Chau, J. L. (2012). The low-latitude ionosphere sensor network: Initial results. Radio Science, 47(04), 1-18. https://doi.org/10.1029/2011RS004978

Vani, B. C., Shimabukuro, M. H., & Monico, J. F. G. (2017). Visual exploration and analysis of ionospheric scintillation monitoring data: the ISMR query tool. Computers & Geosciences, 104, 125-134. https://doi.org/10.1016/j.cageo.2016.08.022

Vani, B. C., de Oliveira Moraes, A., Salles, L. A., Breder, V. H. F., dos Santos Freitas, M. J., Monico, J. F. G., & de Paula, E. R. (2021). Monitoring ionospheric scintillations with GNSS in South America: scope, results, and challenges. In *GPS and GNSS Technology in Geosciences* (pp. 255-280). Elsevier. https://doi.org/10.1016/B978-0-12-818617-6.00012-3

Watanabe, W., Maruyama, R., Arimoto, H., & Tamada, Y. (2020). Low-cost multi-modal microscope using Raspberry Pi. Optik, 212, 164713. https://doi.org/10.1016/j.ijleo.2020.164713

Yamanoor, N. S., & Yamanoor, S. (2017, October). High quality, low cost education with the Raspberry Pi. In 2017 IEEE Global Humanitarian Technology Conference (GHTC) (pp. 1-5). IEEE. https://doi.org/10.1109/GHTC.2017.8239274

Yeh, K. C., and Liu, C. H. (1982). Radio wave scintillations in the ionosphere. *Proceedings of the IEEE*, *70*(4), 324-360. https://doi.org/10.1109/PROC.1982.12313

Zhong, X., & Liang, Y. (2016). Raspberry Pi: An effective vehicle in teaching the internet of things in computer science and engineering. Electronics, 5(3), 56. https://doi.org/10.3390/electronics5030056