An Ad Hoc Broadcast Scheme using the Radio Data System (RDS) in the FM Band

Christopher P. Paolini, Christopher Aguirre, Mahasweta Sarkar, and Santosh Nagaraj

***Abstract*— The Radio Data System (RDS) broadcasts short datagrams in the FM band (87.5 to 108.0 MHz). RDS is primarily used by commercial radio stations to periodically broadcast station identification information or metadata regarding the current content being aired. Part 15 of Title 47 of the Code of Federal Regulations governs the RF emission limits of license-exempt FM modulators that can be used by consumers. Unlicensed broadcasts in the FM band are limited by law to a field strength of 250 μV/m at a distance of 3 meters from the antenna, which is approximately equivalent to 0.01 mW or a range of 200 feet. The low cost and low power consumption characteristics of FM transceivers enables RDS as an alternate option for proximity inter-device communication compared to other wireless technologies such as Wi-Fi, Bluetooth, or cellular data service. In this paper, we exploit RDS by designing an ad hoc broadcast scheme for close proximity inter-device communication. We discuss the implementation of the ad hoc broadcast scheme, which allows for communication among multiple transmitters and receivers, and provides larger reliable datagram broadcasts. We conduct and analyze the performance measurements of small and large datagram broadcasts using RDS on personal FM transceivers.**

***Index Terms*— Ad Hoc Networks, Wireless Broadcast Scheme, Radio Data System, Unlicensed FM Band**

I. INTRODUCTION

MOST contemporary smartphones contain ICs that integrate multiple radios on a single chip. Common configurations are ICs with 802.11b/g/n Wi-Fi and Bluetooth® radios, a GPS receiver, and a FM transceiver. The intent of providing an FM transceiver is to allow the consumer to listen to commercial FM radio and broadcast stored MP3 music files a short distance on a locally unused station. For example, while driving in a car, a user can tune their car stereo to an unused FM frequency and listen to MP3 files that have been stored in their phone's non-volatile memory. With the recent explosion of interest in location-based social networking smartphone applications, the drain on a phone's battery through the use of these applications become a concern, since the frequent transmission and reception of datagrams containing social and location metadata through public 802.11 access points or cell networks can place exhaustive power demands on a phone's battery. While the use of Bluetooth is quickly becoming a popular technology for proximity communication, many users continue to keep their Bluetooth radio in OFF mode or in non-discoverable mode, either to conserve battery power or out of security considerations. The FM transceiver, on the other hand, is typically enabled but generally not used as often as the Wi-Fi or Bluetooth radios, which presents an ideal opportunity to use a phone's FM radio as a device for close-quarters, inter-device communication by location-based social networking or proximity marketing applications. Such applications will often periodically transmit social content about the application user, or receive advertising content tailored to the current time and user's location [3].

A mobile network consisting of multiple smartphone users within 200 feet of one another, where users transiently enter into, and leave each other's proximity, represents an *ad hoc* network where all FM devices have equal status and are free to associate with one another, without the use of management infrastructure, such as an access point or router. Since users may be in motion and come within proximity of one another at different times, a FM device will not know beforehand the number or unique media addresses of other FM devices within range at any particular time. Therefore, an inter-device communication scheme based on the periodic broadcasting of datagrams is more appropriate than a unicast or multicast transmission scheme where foreknowledge of station identifiers is required. A broadcast transmission scheme is one in which FM devices transmit datagrams to all surrounding FM devices, simultaneously, within a 200 foot radius [30].

II. THE RADIO DATA SYSTEM (RDS)

A. *Brief History of RDS*

In 1974, Bosch/Blaupunkt, a German car radio manufacturer, developed the *Autofahrer Rundfunk Information* (ARI) System, or in English, the *Broadcast Information System for Motorists*. Stations using the ARI system would periodically broadcast traffic announcements to motorists. The ARI system used a sub-carrier frequency of 57 kHz and amplitude modulation to encode traffic announcements. Field trials of ARI were conducted between 1980 and 1982, and in 1983, the European Broadcast Union officially adopted the system, which was renamed as the Radio Data System (RDS). Between 1986 and 1990, RDS was officially adopted by the European countries Austria, Belgium, Denmark, United Kingdom, and Italy. During these years, car audio manufacturers

---

Manuscript received July 16, 2013; revised August 16, 2013.

C. P. Paolini is with San Diego State University, San Diego, CA 92182 USA (phone: 619-594-7159; fax: 619-594-2068; e-mail: paolini@engineering.sdsu.edu).

C. Aguirre is with San Diego State University, San Diego, CA 92182 USA (e-mail: aguirrecm@gmail.com).

M. Sarkar is with San Diego State University, San Diego, CA 92182 USA (e-mail: msarkar2@mail.sdsu.edu).

S. Nagaraj is with San Diego State University, San Diego, CA 92182 USA (e-mail: snagaraj@mail.sdsu.edu).

Blaupunkt and Philips began mass developing RDS capable car radios. Finally, in 1990, the European Committee for Electrical Standardization (CENELEC) adopts RDS as the European Standard and in 1992 the Electronic Industry Association and National Association of Broadcasters (EIA/NAB) adopts the Radio Data System as a United States standard.

B. RDS Characteristics

Fig. 1 shows a picture of the stereophonic FM baseband signal with RDS located on the third harmonic of the 19 kHz stereo pilot tone at 57 kHz [1]. The RDS subcarrier frequency of 57 kHz was selected to reduce signal interference on the audio channels [2]. With a 57 kHz oscillator and one bit transmitted per clock cycle, the maximum data rate of RDS is 1187.5 bps [10].

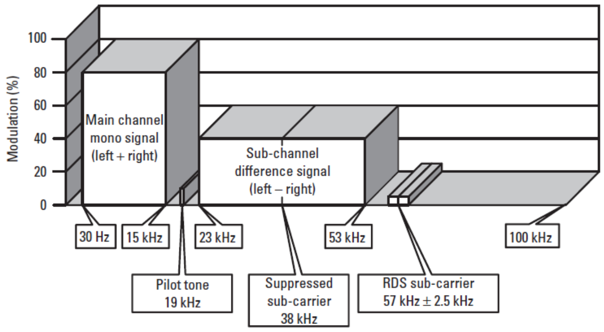

Fig. 1. Diagram of the stereophonic FM baseband signal. RDS is located on the third harmonic of the 19kHz stereo pilot tone at 57 kHz.

Fig. 2 shows a diagram of the coding structure used by RDS, which is comprised of 104-bit groups of four 26-bit blocks. Each block consists of a 16-bit *Information Word* and a 10-bit *Checkword*. The Information Word contains bit codes that define a particular RDS application message while the checkword is used to detect general bit corruption errors and group and block synchronization errors. The RDS designers elected to transmit datagrams using these small block entities because field-testing in Bern/Interlaken showed that mobile reception was more reliable when a radio data stream was broken into small packets.

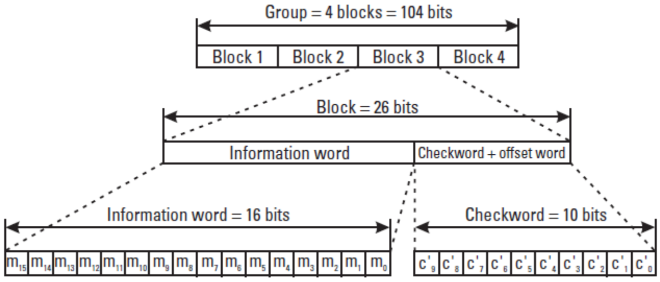

Fig. 2. The datagram coding structure used by RDS can be categorized into groups of blocks.

RDS messages can support a number of different types of applications. While the RDS message format varies with each application, some fields are consistent across all message types. Shown in Fig. 3, these fields include the *PI Code*, a 16-bit *Program Identification* code that typically is used to transmit a radio station's call sign (e.g. "KPBS" or "KFI"), and a 4-bit *Group Type Code* and 1-bit *Bo Flag* that identifies one of 18 types of RDS application messages. In addition, a 1-bit *Traffic Program* (TP) Flag indicates if the current program offers traffic announcements, and a 5-bit *Program Type (PTY)* code that identifies one of 31 different program genres such as news, sports, talk, jazz, weather, etc. Examples of real time traffic announcements include alerting motorists of silent or "no honking" zones, traffic accidents, massive traffic jams, and road side vehicles requiring assistance [5]. RDS is big endian, that is, the order of the transmission begins with the most significant bit in block 1 and least significant bit in block 4.

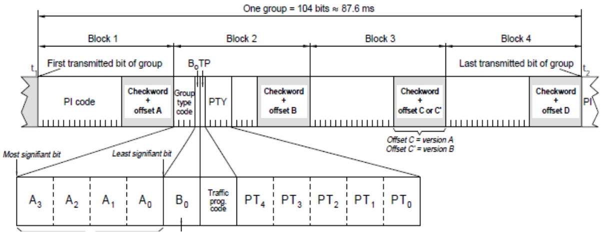

Fig. 3. The RDS message format showing the PI code, Bo flag, TP flag, and PTY code.

C. Additional Present Day RDS Applications

In addition to broadcasting station identification and content metadata, RDS is currently used by emergency warning systems to broadcast information about national disasters, dangerous weather conditions, and hazardous chemical spills. RDS can also be used to send alphanumeric messages and alerts to mobile pocket pagers, as well as remotely synchronize clock and time data on receivers [4,6].

D. Radio Text Messages

RDS supports two types of group structures for broadcasting arbitrary text messages: *Type 2A* that can transmit four characters per group, and *Type 2B* that can transmit two characters per group. Fig. 4 shows the structure of a Type 2A message, which has two 16-bit *Radio Text Segment* fields that can be used to encapsulate four arbitrary ASCII alphanumeric characters. The 4-bit *Text Segment Address Code* identifies the group's text position offset in a larger text message that has been fragmented. The *Text A/B* flag instructs the receiver to append the group fragment to a larger message being assembled from multiple Radio Text Segment groups, or to begin a message by indicating the current group is the first fragment of a new message.

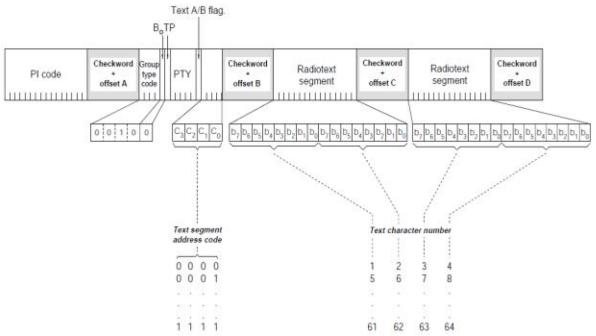

Fig. 4. The RDS Radio Text Type 2A Message structure.

E. Maximum Goodput

If we consider RDS as a data link layer protocol and define *goodput* as number of bits received by the application layer per unit time, the maximum goodput achievable using Radio Text Type 2A messages is 605.17 bps:

$$PI\ (16) + PTY\ (5) + Radio\ Text\ (32) = 53\ bits \Rightarrow$$
$$(1187.5\ bits/s)(53\ bits/104\ bits) = 605.17\ bits/s \quad (1)$$

III. IMPLEMENTATION

A. Hardware Configuration

An application was developed on an embedded single board computer that uses the RDS protocol to periodically broadcast social networking information to mobile FM devices within a 200-foot radius. Implementation of the application was performed on a Texas Instruments (TI) Pandaboard ES Development Platform. The Pandaboard includes a LS Research TiWi-R2 Transceiver Module [14] which encapsulates a TI WiLink™ 6.0 1271 IC. The WiLink™ 6.0 1271 includes an 802.11b/g/n radio which uses the 2.4-GHz band, Bluetooth 2.1 Release, ANT, and FM transmit and receive capabilities. The PandaBoard uses the TI OMAP4430 Multimedia Processor [21,22] that is a chipset widely used in wireless mobile applications and popular e-readers, such as the Amazon Kindle Fire, Barnes and Noble Nook, and Samsung Galaxy Tablet [23,24,25]. Two different external antennas were used with the PandaBoard: a half-wave monopole antenna having length 4.75 feet for use with the unlicensed frequency 103.3 MHz (in San Diego), and an Antenova M10385 Internal SMD FM Antenna [26,27]. The Antenova M10385 is a low-cost, miniature (30mm x 5mm) PCB designed for use in mobile devices such as mobile phones, notebooks, portable media players, and Media Tablets. A spectrum analyzer (SA) was used to measure transmitted power from the PandaBoard using different combinations of the half-wave monopole and M10385. Table I shows the results of this measurement. It was found that the best combination was to use the half-wave monopole for Tx and the M10385 for Rx.

B. Software and Protocol Design

An application-programming interface (API) was written to implement an ad hoc broadcast scheme to use RDS as a data-link layer protocol. The API encapsulates a user's social profile, consisting of variable length ASCII text structures, into one or more RDS Radio Text Type 2A Message structures, which are then periodically transmitted by the code running on a PandaBoard and received by other PandaBoard stations. The API implements social profile message datagram fragmentation and reassembly at the application layer. Datagram error checking at the application layer is also provided, in addition to the checkword facility provided by RDS.

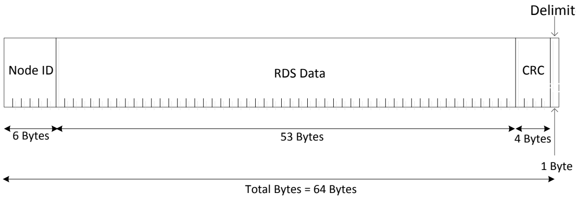

Fig. 5. Format of a 64-bye *broadcast fragment*.

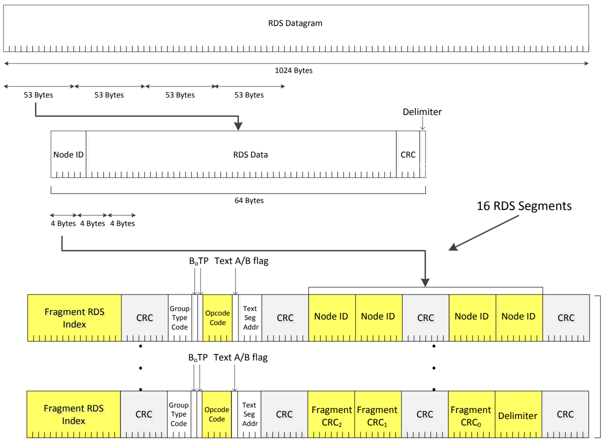

Fig. 6. Fragmenting a 1KB message into multiple 64-byte broadcast fragments which are, in turn, fragmented into RDS Radio Text Type 2A Message frames.

The application layer protocol for message transport uses a 64-byte *broadcast fragment*, the structure of which is shown in Fig. 5. A broadcast fragment consists of a 6-byte header for storing a unique address to identify the broadcasting device, a 53-byte payload field for storing a fragment of a user's social profile, a 4-byte field to hold a CRC-32 checksum, and a 1-byte delimiter equal to the hexadecimal value 0x0D to mark the end of a datagram. Our application layer API uses the broadcast fragment frame to fragment an arbitrarily long social networking profile into 64-byte frames. The API in the application layer performs fragmentation and reassembly of broadcast fragments. As shown in Fig. 6, a user's social networking profile, say 1KB long, is fragmented at the application layer by the API into multiple 64-byte broadcast fragments which are, in turn, further fragmented into multiple RDS Radio Text Type 2A Message frames and then transmitted by the FM transceiver. Each byte broadcast fragment is fragmented into 16 Radio Text Type 2A Messages. The *Fragment RDS Index* field is used by the API to reassemble received Radio Text Type 2A frames into a broadcast fragment. If one or more Radio Text Type 2A frames are corrupted during transmission, the reassembled broadcast fragment will fail a CRC-32 check and the API will discard the broadcast fragment and wait for another broadcasted instance of 16 Radio Text Type 2A frames needed to reassemble the broadcast fragment. If one or more Radio Text Type 2A frames are lost during transmission, a reassembly timer maintained by the API will expire, resulting in the discarding of all received Radio Text Type 2A frames having the same Fragment RDS Index value. The API will then wait for the next broadcast instance of 16 Radio Text Type 2A frames needed to reassemble the broadcast fragment.

IV. PERFORMANCE MEASUREMENTS

A. Measurement Configuration

Two PandaBoards were configured with our API to measure throughput, goodput, latency, and the inter-packet delay of our RDS-based broadcast scheme. Performance measurements were made between one PandaBoard configured as a transmitter and the other PandaBoard configured as a receiver. Measurements were taken with the two PandaBoards at distances of 10 feet to 200 feet apart, in increments of 10 feet. Two sizes of social networking messages were transmitted: a 1KB message and a 4KB message. All measurements were taken with the two PandaBoards within a line-of-sight.

B. *Throughput Results*

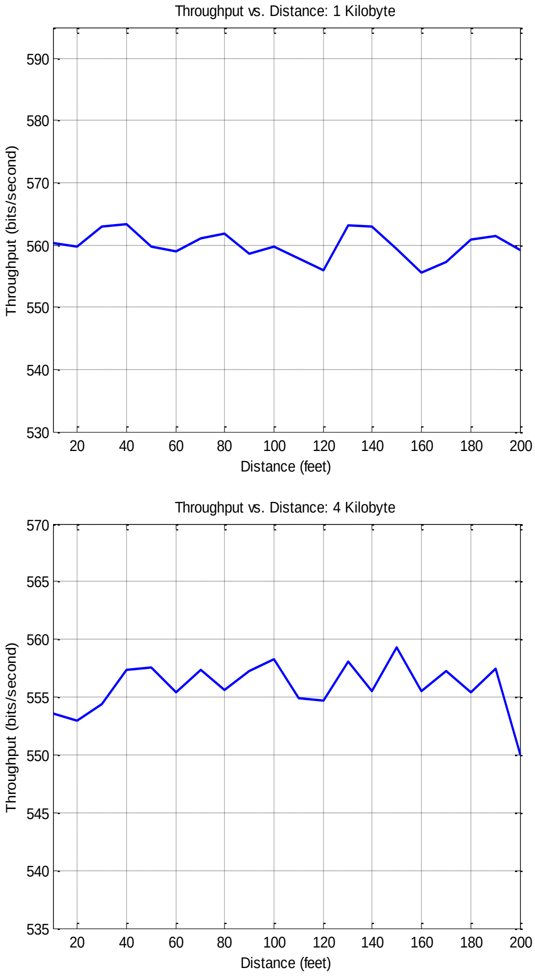

Fig. 7. Measured RDS broadcast throughput on FM station 103.3 MHz for 1 KB (top) and 4 KB (bottom) sized social networking messages. PandaBoards were placed 10 feet to 200 feet apart, in increments of 10 feet, in a line-of-sight configuration.

We define *throughput* as the total number of bits received by the RDS data-link layer per unit time [29]. Fig. 7 shows the measured RDS broadcast throughput on FM station 103.3 MHz for one and four kilobyte sized social networking messages. The one KB graph shows a steady throughput of 558-563 bps with an overall average throughput of 560 bps, while the four KB message throughput graph shows a steady throughput of around 555-558 bps with an overall average throughput of 556 bps.

C. *Goodput Results*

We define *goodput* as the number of bits received by the API running at the application layer per unit time. Fig. shows the measured RDS broadcast goodput for one and four kilobyte sized social networking messages. The one KB graph shows a relatively constant goodput with an overall average of 187 bps, while the four KB message throughput graph shows a steady rate of around 189-190 bps with an overall average of 189 bps. As seen in the Fig. 8, the goodput noticeably diminishes at the 200-foot legal distance limit. Based on the results, the goodput of the larger sized message broadcast appears to be slightly higher than the smaller sized message broadcast. Both message broadcasts appear to slightly decline as the distance approaches 200 feet.

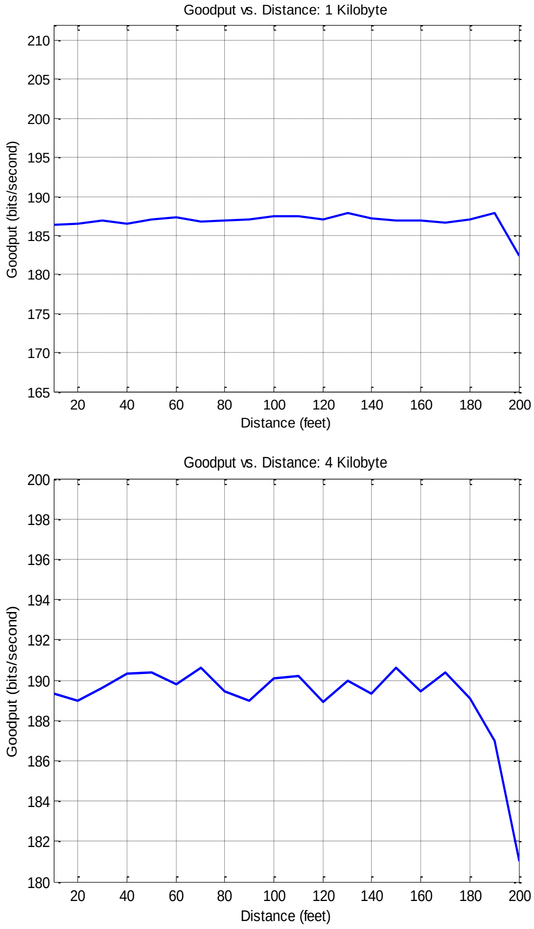

Fig. 8. Measured RDS goodput on FM station 103.3 MHz for 1 KB (top) and 4 KB (bottom) sized messages.

D. *Delay Results*

We define *delay* as the time elapsed from the start of an application layer broadcast fragment being transmitted by a device to the time the broadcast fragment has been received in its entirety by a receiving device. Fig. 9 shows the measured delay for fragments from a one and four kilobyte sized social networking message. The one KB graph shows an overall average delay of 0.75 seconds, while the four KB graph shows an overall average delay of 0.72 seconds. As expected, there is no appreciable difference in broadcast fragment delay for fragments originating from the shorter or longer application layer message.

E. *Inter-packet Delay Measurements*

We define RDS inter-packet delay as the time required to wait between transmitting successive Radio Text Type 2A frames. For our throughput, goodput, and delay measurements, we used a default inter-packet delay of 5 seconds. When a Radio Text Type 2A frame is constructed and submitted to the transceiver for transmission, a certain period of time is required for the transceiver to transmit the frame. Attempting a subsequent Radio Text transmission too soon will result in data corruption, which is detected by a checkword failure. Fig. 10 shows a bar chart of the measured goodput, as a function of inter-packet delay, for

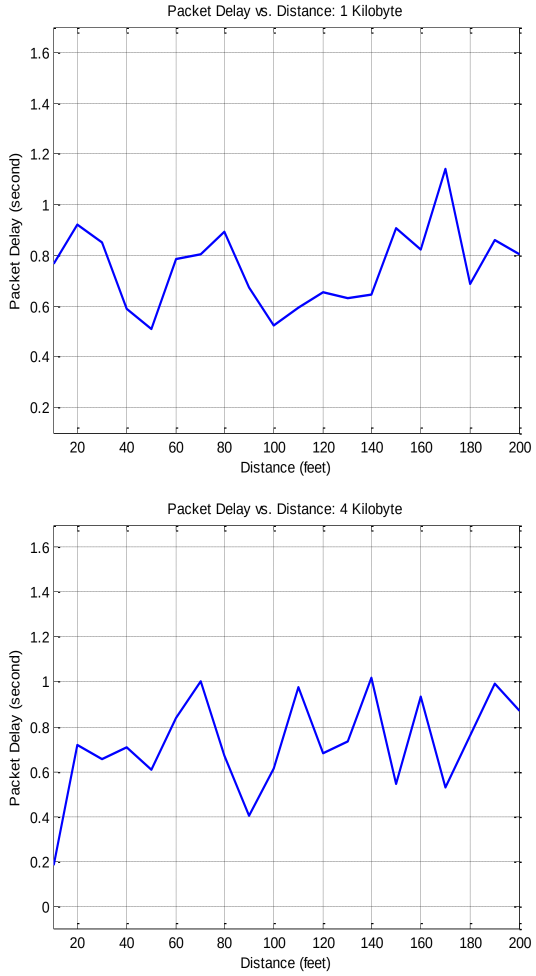

Fig. 9. Measured broadcast fragment delay on FM station 103.3 MHz for 1 KB (top) and 4 KB (bottom) sized messages.

one PandaBoard transmitting Radio Text Type 2A messages to a receiving PandaBoard placed 120 feet apart. As the chart shows, with a 1-second delay, no Radio Text Type 2A messages were found to pass a checkword test. Higher goodput was seen where the inter-packet delay was configured to be 5 seconds or greater.

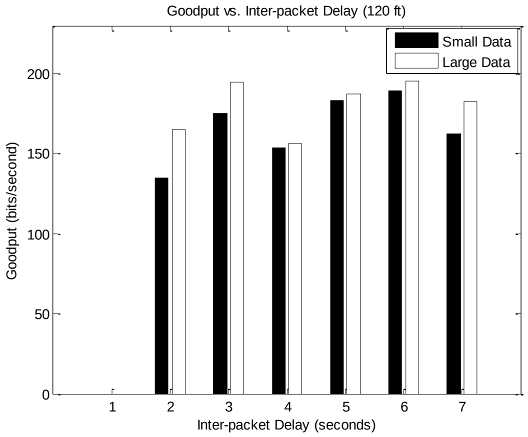

Fig. 10. RDS Radio Text Type 2A message goodput as a function of inter-packet delay, measured at a Tx to Rx device distance of 120 feet. With a 1-second delay, no Radio Text Type 2A messages were found to pass a checkword test.

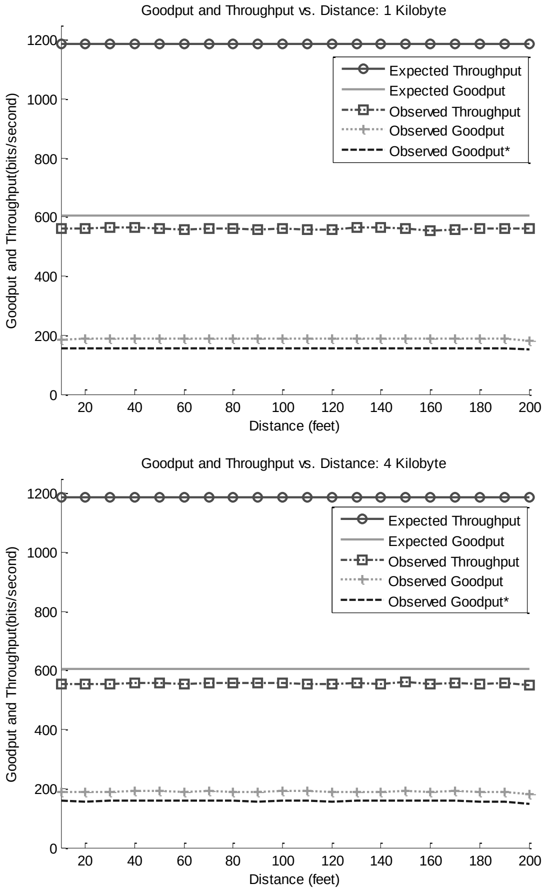

Fig. 11. A comparison of expected vs. observed network performance, as a function of distance, for 1 KB (top) and 4 KB (bottom) sized social networking messages.

V. CONCLUSION

An ad hoc broadcast scheme is proposed that uses the Radio Data System (RDS) of FM radio in a new, novel way for digital, short distance social media device communication. The design of this ad hoc broadcast scheme allows larger reliable broadcast data transfers and can support multiple broadcast environments. Performance measurements were taken between a single transmitter and a signal receiver. A summary of measured throughput and goodput is summarized in Fig. 11. With an expected, theoretical throughput of 1187.8 bps, the actual measured throughput was found to be much less: 560 bps for one KB sized messages, and 556 bps for 4 KB messages. Similarly, with a theoretical maximum goodput of 605 bps, the actual goodput was found to be significantly less: 187 bps for one KB messages and 189 bps for 4 KB messages. Recall our definition of goodput is the number of bits delivered to the application layer per unit time. If we modify our definition of goodput to consider only reassembled datagrams that pass the datagram CRC in the application layer, the goodput drops to 155 bps and 157 bps for one and four KB messages, respectively, as shown by the cyan lines in Fig. 11. Thus, using unlicensed FM to transmit datagrams via RDS with a theoretical throughput of 1187.8 bps, one will observe an effective application message data rate of about 13% of the maximum theoretical throughput. Our findings show that when using the monopole antenna as the transmitter antenna and the 10385 Antenova antenna module as receive antenna, the goodput and throughput does slightly decline at 200 feet, but overall, it appears that the distance does not impact the goodput or the throughput within the FCC legal distance range. It is important to note that the transmitter and the receiver were placed in almost ideal conditions: direct line of sight, no obstructions and/or, interference, and the transmitter and receiver were stationary when measurements were taken. In conclusion, using RDS to broadcast social networking data on smartphones between people within close proximity is possible, but the significant reduction in effective data rate must be taken into consideration.

TABLE I
ANTENNA POWER MEASUREMENTS

Transmit Antenna (PandaBoard)	Receive Antenna (Spectrum Analyzer)	Measured Power (dBm)
Antenova	Antenova	-84.93
Antenova	Monopole ½ λ	-89.30
Monopole ½ λ	Antenova	-62.64
Monopole ½ λ	Monopole ½ λ	-70.21

Federal Communications Commission (FCC) Low Power Broadcast Operation Regulations limit maximum effective radiated power to -50 dBm which given an approximate maximum coverage radius of 200 ft.

REFERENCES

[1] EIA/NAB (1998), National Radio Systems Committee: United States RBDS Standard, Specification of the Radio Broadcast Data System (RBDS).

[2] D. Kopitz and B. Marks, RDS..The Radio Data System. Artech House, 1998

[3] Hang Yu , Ahmad Rahmati , Ardalan Amiri Sani , Lin Zhong , Jehan Wickramasuriya , Venu Vasudevan, Data broadcasting using mobile FM radio: design, realization and application, Proceedings of the 13th international conference on Ubiquitous computing, Beijing, China, September 2011, pp. 17-21.

[4] Liqun Li , Guoliang Xing , Limin Sun , Wei Huangfu , Ruogu Zhou , Hongsong Zhu, Demo: a sensor network time synchronization protocol based on fm radio data system, Proceedings of the 9th international conference on Mobile systems, applications, and services, June 28-July 01, 2011, Bethesda, Maryland, USA

[5] Agarwal, A.; Goel, N., "GPS-RDS enabled location based smart transit," Intelligence in Next Generation Networks, 2009. ICIN 2009. 13th International Conference on , vol., no., pp.1,5, 26-29 Oct. 2009, doi: 10.1109/ICIN.2009.5357105

[6] Kuang-Hao Lin, "Implementation of radio data systems decoder," Information, Communications and Signal Processing, 2009. ICICS 2009. 7th International Conference on , vol., no., pp.1,4, 8-10 Dec. 2009, doi: 10.1109/ICICS.2009.5397614

[7] Jui-Yuan Lin; Kuang-Hao Lin; Shuenn-Yuh Lee, "Digital RDS demodulation in FM subcarrier systems," Circuits and Systems, 2004. Proceedings. The 2004 IEEE Asia-Pacific Conference on , vol.1, no., pp.221,224 vol.1, 6-9 Dec. 2004, doi: 10.1109/APCCAS.2004.1412732

[8] Berenguer-Sau, J.; Villanueva-Fernandez, A., "A broadcast synthesized FM modulator with a radio data system encoder," Broadcasting, IEEE Transactions on , vol.42, no.2, pp.117,121, Jun 1996, doi: 10.1109/11.506828

[9] Berezin, S.; Nephin, D.; Wang, J., "FM RDS for smart charging of PEVs," Transportation Electrification Conference and Expo (ITEC), 2012 IEEE , vol., no., pp.1,5, 18-20 June 2012, doi: 10.1109/ITEC.2012.6243422

[10] de-Ridder-de-Groote, M.J.; Prasad, R.; Bons, J.H., "Analysis of new methods for broadcasting digital data to mobile terminals over an FM-channel," Broadcasting, IEEE Transactions on , vol.40, no.1, pp.29,37, Mar 1994. doi: 10.1109/11.272418

[11] Glazar, B.; Jankovec, M.; Topic, M., "FM RDS PnP radio receiver," EUROCON 2003. Computer as a Tool. The IEEE Region 8 , vol.1, no., pp.114,117 vol.1, 22-24 Sept. 2003, doi: 10.1109/EURCON.2003.1247990

[12] Matulic, J.; Sisul, G.; Modlic, B., "Realization of RDS encoder with atmel microcontroller," Applied Electromagnetics and Communications, 2007. ICECom 2007. 19th International Conference on , vol., no., pp.1,4, 24-26 Sept. 2007, doi: 10.1109/ICECOM.2007.4544486

[13] Roppel, C.; Perez, A.O., "An efficient implementation of an FM/RDS software radio," Education and Research Conference (EDERC), 2010 4th European , vol., no., pp.245,249, 1-2 Dec. 2010

[14] C.Aguirre (2012, November 12). FM Specification.[Online] Available e-mail: jbablic@lsr Message: Re: FM Specification

[15] Li, C.C.; Lin, C.E.; Tsai, C. F.; Chiang, S. C., "An airborne Collision Avoidance System for low altitude flights using Radio Data System," Digital Avionics Systems Conference, 2008. DASC 2008. IEEE/AIAA 27th , vol., no., pp.6.A.2-1,6.A.2-6, 26-30 Oct. 2008, doi: 10.1109/DASC.2008.4702863

[16] Wang, J.; Cheng, F.; Wang, J.; Shen, G.;Xiaofan, J., “Genius-on-the-go: FM radio based proximity sensing and audio information sharing,” Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems, 2012. SenSys 2012. ACM, pp.363-364, doi: 10.1145/2426656.2426710

[17] Chen Wei; Sheng-Wen Hong, "An FM subcarrier data broadcast on bus transportation status indication system," Broadcasting, IEEE Transactions on , vol.47, no.1, pp.76,79, Mar 2001, doi: 10.1109/11.920784

[18] Rahmati, A; Zhong, L; Vasudevan, V; Wickramasuriya, J; Stewart, D, “Enabling Pervasive Mobile Applications with the FM Radio Broadcast Data System, “Proceedings of the 13th international conference of Ubiquitous computing, 2011. UbiComp’11, AMC, pp. 247-256, doi: 10.1145/2030112.2030146

[19] M. Buchanan (2009, Oct 13) In-house Radio.app in the works for iPhone and iPod touch[Online]. Available: http://gizmodo.com/5471805/windows-phone-7-series-everything-is-different-now

[20] M. Buchanan (2010, Feb 10) Windows Phone 7 Series: Everything is Different Now[Online]. Available: http://9to5mac.com/2009/10/13/in-house-radio-app-in-the-works-for-iphone-and-ipod-touch/

[21] Pandaboard Development Board References [Online]. Available: http://pandaboard.org/sites/default/files/board_setup_v3.png

[22] OMAPTM 4 PandaBoard System Reference Manual, Revision 0.4, September 22, 2010, DOC-21010

[23] Barnes and Nobles Nook Tablet Specifications (16GB & 8GB Models). [Online]. Available: https://nookdeveloper.barnesandnoble.com/devices/nook-tablet

[24] Amazon Kindle Device and Feature Specifications. [Online]. https://developer.amazon.com/sdk/fire/specifications.html

[25] Samsung Galaxy Tab 2 10.1” 16GB Android 4.1 Tablet Specifications [Online]. http://www.bestbuy.ca/en-CA/product/samsung-samsung-galaxy-tab-2-10-1-16gb-android-4-0-tablet-with-ti-omap-4430-processor-silver-gt-p5113/10206558.aspx

[26] Internal SMD FM Antenna Module Product Specification, Product Specification 10MD-0038-PS, August 12, 2010

[27] M10385 FM Module Evaluation board, 2010

[28] A. Jurison (2011, Jan 6) RDS: Injection and Pilot Synchronization [Online]. Available: http://www.rwonline.com/article/rds-injection-amp-pilot-synchronization/4463

[29] A. Tanenbaum, Computer Networks, 4th ed. Upper Saddle River, NJ: Prentice Hall, 2003

[30] Public Notice Permitted Forms of Low Powered Broadcast Operations, Federal Communications Commission, July 19.