On the use of Reed-Solomon codes to extend link margin and communication range in low-power wireless networks

M. K. Khan†, K. Mulvaney†, P.Quinlan†, S. O'Mahony†, C.Billon†, N.Kearney†, J.Lopez† and T. Conway‡

† Wireless Transceiver Group
Analog Devices Inc., Cork, Ireland.
e-mail :kalim.khan@analog.com

‡ ECE Department
University of Limerick, Limerick, Ireland
e-mail: Thomas. Conway @ ul.ie

Abstract — **Reed-Solomon (RS) codes are commonly used to correct burst errors in applications such as storage devices, data transmission, bar-code readers, and satellite communications. In this paper, we evaluate the use of RS coding to improve link margin and communication range in a low power wireless network. Performance metrics are also evaluated for latency and processing time required to perform the RS coding and decoding algorithms**

The wireless transceiver used for these measurements is based on the ADF7023 from Analog Devices, which is a high performance, ultra-low power, RF IC designed for a broad range of wireless telemetry applications that includes: smart metering, wireless sensor networks, home automation and IEEE 802.15.4g. The transceiver also features an ultra low power on-chip 8-bit RISC processor that performs a number of Physical and MAC layer functions such as: radio control, packet management, 128/256 bit AES and FEC such as Reed-Solomon codes.

Keywords **– Low power sub-GHz RF Devices, Reed Solomon Code, ADF7023, Wireless Sensor Network**

I INTRODUCTION

Wireless communication links are prone to channel impairments such as signal fading and interference from neighbouring channels as well as unwanted high power in-band and out-of-band blockers. Reed Solomon (RS) FEC (Forward Error Correction) coding [1-3] is commonly used to improve the resilience of wireless links to these kinds of channel impairments due to its excellent burst error correction properties.

In this paper, we investigate the use of RS FEC coding to improve link margin by enhancing the radio receiver's sensitivity performance, where random bit errors tend to dominate at low receiver SNR and minimum input power.

Link margin can be improved by increasing the output power of the transmitter or by reducing the receiver's noise figure. However, both of these solutions require a significant increase in either transmit power or receiver power which significantly degrades battery lifetime in a low power wireless network.

For these systems, applying FEC techniques to the radio link payload offers a better system trade-off without causing a significant power penalty. For example, an FEC coding gain of 3dB will result in a corresponding improvement in receiver sensitivity and extends the communication range by approximately 40% using a line of sight link-margin model.

The transceiver used in this evaluation is the ADF7023 [4] RF IC which is an industry leading RF transceiver designed to operate in the sub-GHz ISM (Industrial Scientific and Medical) frequency bands. The transceiver is equipped with a low power RISC processor [5] for packet handling and processing of data packets. The flexible architecture of the design enables the on-chip RISC processor to perform various on-chip algorithms such as image rejection calibration, Advanced Encryption Standard (AES) and FEC.

A block diagram of the ADF7023 is shown in figure 1. The serial peripheral interface (SPI) bus is used to access on-chip memory blocks. The microcontroller or user interface software can be programmed to download data or firmware modules using a SPI interface. One of the firmware modules available for download into Program RAM is a Reed-Solomon Codec. This module can be used with a range of Analog Devices RF Transceiver products such as ADF7022, ADF7241, and ADF7242 [6].

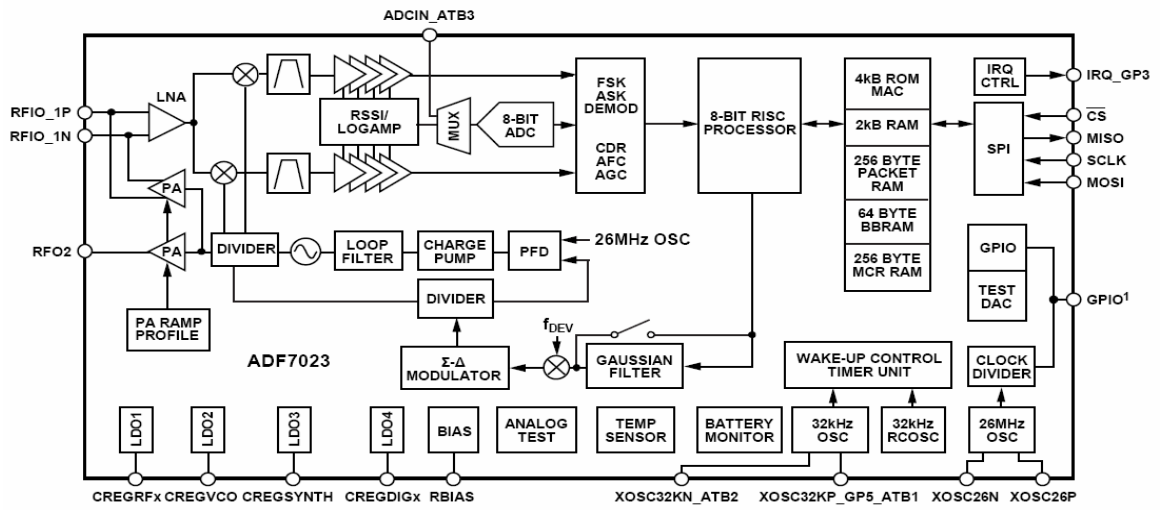

Figure 1 ADF7023 Block diagram

The firmware module enables the RISC processor to perform RS encode and decode operations. The instruction set is executed from Program RAM and the clock speed of the processor can also be dynamically increased to allow faster execution of tasks to minimise system latency. Furthermore, hardware acceleration blocks are provided in the RISC processor to perform Galois Field *GF (q)* multiplication, addition and inversion operations to further minimise encode/decode execution time. A number of packet RAM locations are reserved to store intermediate data for operations such as GF(256) multiplications, additions and syndrome calculations used in a Berlekamp Massey decoding algorithm.

The paper is organised as follows. The general description of RS coding and the ADF7023 implementation details are described in section II. Section III describes the basic measurement setup and RS codes used to perform PER measurement using the device evaluation board. Section IV describes the packet structure, PER results of RS codes and RS encode/decode timings. Finally section V concludes the paper.

II REED SOLOMON CODES ON ADF7023

Reed-Solomon codes are non-binary systematic block codes in which the encoder appends a number of check-sum bytes to the packet payload. A block diagram of the system channel model is shown in figure 2.

The encoder takes in k bytes of data and produces n bytes of encoded data. The check bytes (n-k) are appended to the messaged block to produce an RS (n,k) code rate, R = k/n. The total number of error bytes that can be corrected is t = (n-k)/2 and the implementation investigated is capable of correcting up to 5 error bytes. The theoretical concept of the RS codec is based on algebraic operations over mathematical structures known as finite field or GF (q). This implementation is based on a GF (2^8) where each symbol is represented as 8-bits and all encode and decode operations use 8-bit symbol processing.

a) Reed Solomon Configuration

RS code (n , k) over the field GF(2^8), i.e., n <= 255, with GF(2^8) field is based on the irreducible polynomial $z^8 + z^3 + z + 1$. The RS code is constructed using a primitive element B = z+1, from GF (2^8) represented as the byte value binary 00000011.

The RS generator polynomial is

$$G(x) = (x\text{-}B)\ (x\text{-}B^2)\ (x\text{-}B^3)....(x\text{-}B^{n\text{-}k\text{-}1}) \qquad (1)$$

For ease of implementation the value of n-k has been restricted to be an even number, i.e. 2,4,6,8,10 corresponding to 1,2,3,4,5 symbol error correction capability.

The RISC processor encodes the data bytes stored in Packet RAM when the CMD_RS_ENCODE is issued. The check bytes calculated are appended at the end of locations where the uncoded data bytes are stored. Once, the encoding operation finishes, the user issues a TX command to transmit the data. The RISC processor transmits the data using G\FSK modulation scheme.

On the receiver side of the communication link, when preamble and synchronization fields are detected, the encoded data bytes are stored in a pre-defined packet RAM location and the device transitions to the PHY_ON state. The RISC processor then performs a decode operation to detect and correct errors if they exist in the received packet. To use the on-chip RS codec a number of additional parameters and commands are required, which are listed in the RF IC datasheet [4]. Table I shows some of the RAM locations used to store RS code related information.

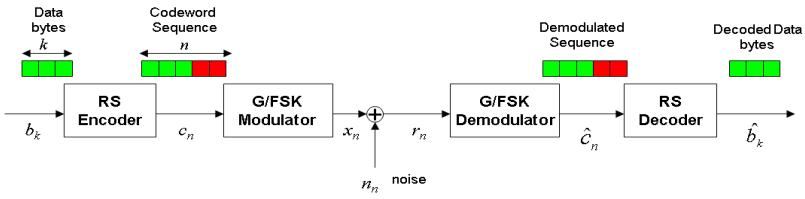

Figure 2 Reed Solomon Codec in Communication System

RAM Location	Description
214	Store RS command to be executed
215	Address of the start of *k* data bytes for the encoding or n encoded data for decoding
216	The value of *k* i.e. the number of data bytes in the codeword
217	The value of *n* i.e. the number of data bytes plus check symbols in the codeword
218-255	Store RS encode/decode internal operation results

Table I: RS Code Specific Locations in Packet RAM

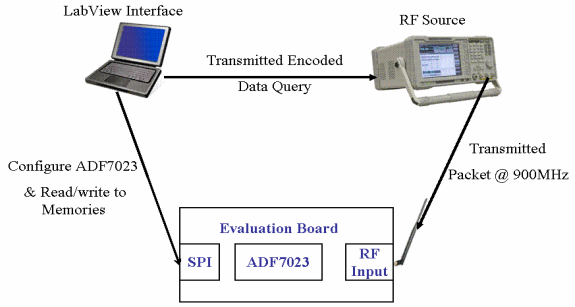

Figure 3 RS Code Evaluation Setup

III. Reed Solomon Codes Evaluation procedure

For performance evaluation purposes, a Labview interface is used to compare the Packet Error Rate (PER) of RS FEC coding with un-coded data packets. The diagram of the test measurement setup using the customer Evaluation board is shown in figure 3. In the setup the controller interface goes through the sequence and programs the device into various configuration modes e.g. Phy_ON, Phy_OFF, Phy_RX states as defined in the data sheet [4]. The program configures the device into RX mode and then the RF source transmits encoded data using FSK modulation. The packet is received and stored in the Packet RAM location , then the device goes into Phy_ON state and a CMD_RS_DECODE command is issued to perform RS decode operation. The interface controller compares the decoded data with the transmitted encoded data bytes and generates packet error rate information at various RS code rates.

The RS encode and decode operations are performed at the maximum processor speed of 26 MHz. The current consumption due to this off-line processing in the digital baseband is independent of the data rate of the system but it depends upon the length of the packet, maximum error correcting capability *t* and the code rate *R*. For RS(38,28,t=5) listed in table II, the extra encoding operation at the transmitting stage represents less than 2% of the total current required to transmit the encoded data bytes. While at the receiving end it represents less than 5% of the total current to receive and decode the packet. The device is also programmed to generate interrupts [4] which indicates encode/decode operations are finished. These interrupts are routed to the general purpose input/output (GPIO) pins [4] and are used to calculate the various timing measurements of the RS coding algorithm.

Encoded Bytes Length *(n)*	Information bytes length *(k)*	Check bytes *(m=n-k)*	Max bytes Correcting Capability *(t= m/2)*	Code Rate *(R=k/n)*
38	28	10	5	0.74
36	28	8	4	0.77
34	28	6	3	0.82
32	28	4	2	0.875
30	28	2	1	0.933

Table II: RS Codes used for Evaluation

IV. Evaluation Results and Timing Measurements

For target applications a message length of up to 10-40 bytes is typically used. In this work a packet length of 28 bytes was chosen for these measurements and tested over a range of 1 to 5 correction bytes, corresponding to a code rate from 0.933 to 0.74 respectively.

The RS coding gain is measured using the code rates defined in table II. The preamble and synchronization frame error tolerances are also varied to evaluate the effect of these parameters on the effective coding gain. The parameter PML (preamble match level) is used to control the level of preamble qualification tolerance in the receiver. Also, Sync. Tol. (synchronization error tolerance) is used to control the receiver's qualification tolerance to the byte synchronization word in a packet. Both PML and Sync. Tol. are programmable using the serial peripheral interface (SPI) on the device [4].

The device is configured to receive encoded data packets from the RF source in the 928MHz ISM band using 2-FSK modulation at 38.4 kbps, 100kbps and 300 kbps. For example, RS (38, 28) packet with t=5, is shown in figure 4. The preamble length is set to 100 bits and a 24-bit synchronization field: 0xB58936 (Hexadecimal representation) is used for byte synchronization.

Figure 5, shows the Packet Error Rate performance (in percentage) with respect to the RF input received power at 100 kbps at various code rates. The figure shows that the receiver's 1% sensitivity level is improved significantly by using RS codes. It also shows that as more correcting bytes are applied the RS coding gain is increased asymptotically. The RS codes have shown good performance improvement at the receiver's sensitivity level where random bit errors tends to dominate.

From figure 5, we observed that using the maximum error correcting capability of t = 5 did not achieve the theoretical coding gain limit of approximately 4dB, based on an Additive White Gaussian Noise (AWGN) channel model. Analysis of the packets errors showed that the packet loss was dominated by the combinations of preamble and synchronization frame detection errors. In order to achieve the coding gain limit it was necessary to relax PML and Sync. Tol. The results using new optimized settings are shown in figure 6. The figure shows PER curves for a number of combinations of PML and Sync. Tol. settings. As seen from the figure, by relaxing PML and Sync. Tol. the coding gain of approx. 4dB was achieved which results in significant improvement in the link margin and the communication range in low power wireless networks.

However, relaxing PML and Syn. Tol. will increase false packet detection at the receiver. To analyse this effect measurements were taken of the occurrence of false packet using various combinations of PML and Sync. Tol. These results are shown in figure 7. The figure shows that relaxing Sync. Tol. from 0 to 2 increases the false packet occurrence from 0.6 packets/hour to 50 packets/hour. However, in low power wireless applications these wireless nodes are duty cycled such that they are active only for very short periods while transmitting and receiving telemetry data. For low power wireless applications this represents a good system performance trade-off. Furthermore, these tolerances settings can be adapted and optimized to accommodate time varying channel conditions.

Table III summarises the measurements obtained at various data rates using the RS (38, 28, 0.74, t=5) code. The overall coding gain of the system achieved is limited by PML and Sync.Tol. settings. At 100kbps, a coding gain of 4.1dB is achieved using the receiver tolerance settings of PML=7 and Sync.Tol=2. This coding gain can be used in a number of ways to improve the system performance parameters such as communication range and battery lifetime.

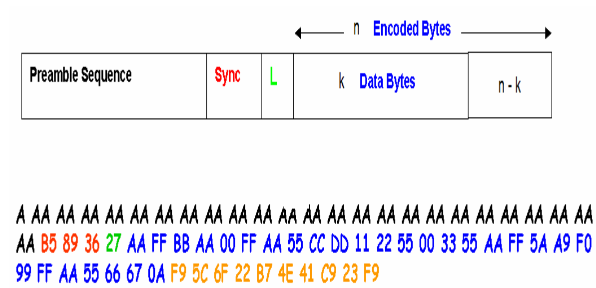

Figure 4 Example Packet Structure

Estimating the battery lifetime depends on several factors but for most applications using power-cycled wireless nodes it is typically dominated by the power dissipated during transmitter's active time or the receiver's active time in addition to battery self-discharge and system leakage current.

FSK Data Rate	RS Code	Code Rate	Preamble Error Tolerance	Sync Error Tolerance	Coding Gain (dB)
38.4 kbps	(38,28,t=5)	0.74	2	0	1.3
			2	1	2.8
100 kbps	(38,28,t=5)	0.74	2	0	2.1
			2	1	3.5
			6	1	3.9
			7	2	4.1
300 kbps	(38,28,t=5)	0.74	2	0	2 dB
			2	1	3.4 dB

Table III. Reed Solomon Coding Gain Summary

The 4.1dB RS FEC coding gain can be used to extend the effective communication range by a factor of approximately 60%, without an increase in the transmit output power. Increasing the ADF7023's transmitter output power by 4.1dB in an uncoded system will also deliver a similar increase in communication range. However, in low power duty-cycled applications, the FEC option can be more power efficient. As an example, in a 0.5sec duty-cycled wireless application, using a 1Ahr AAA battery with a 7uA system leakage current, an improvement of 7% in battery lifetime is achieved using an RS (38,28, t=5) FEC code at 100kbps.

A further improvement in the battery life-time can be obtained by increasing the RS FEC code rate which reduces the overall effective coding of the wireless system. For example, in a 0.5sec duty cycled system, using RS (34,28, t=3) code with rate 0.82 at 100kbps improves a 1Ahr AAA battery life-time by 16% and achieves a coding gain of at least 2dB.

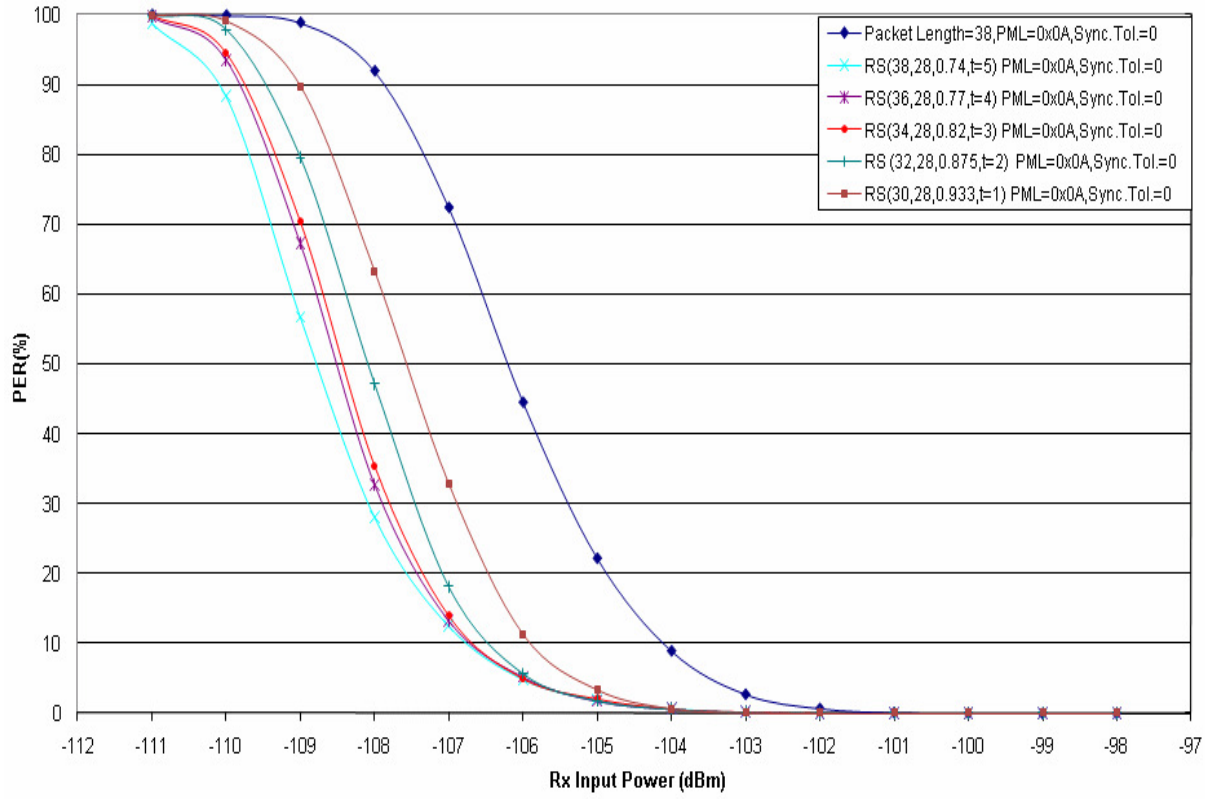

Figure 5 RS Code, Packet Error Rate vs. RF Input Power
RF= 928MHz, FSK Data Rate = 100kbps, Fdev =25KHz, Preamble Errors allowed=2
Sync. Error allowed =0, Number of Packet = 10000/Power Level

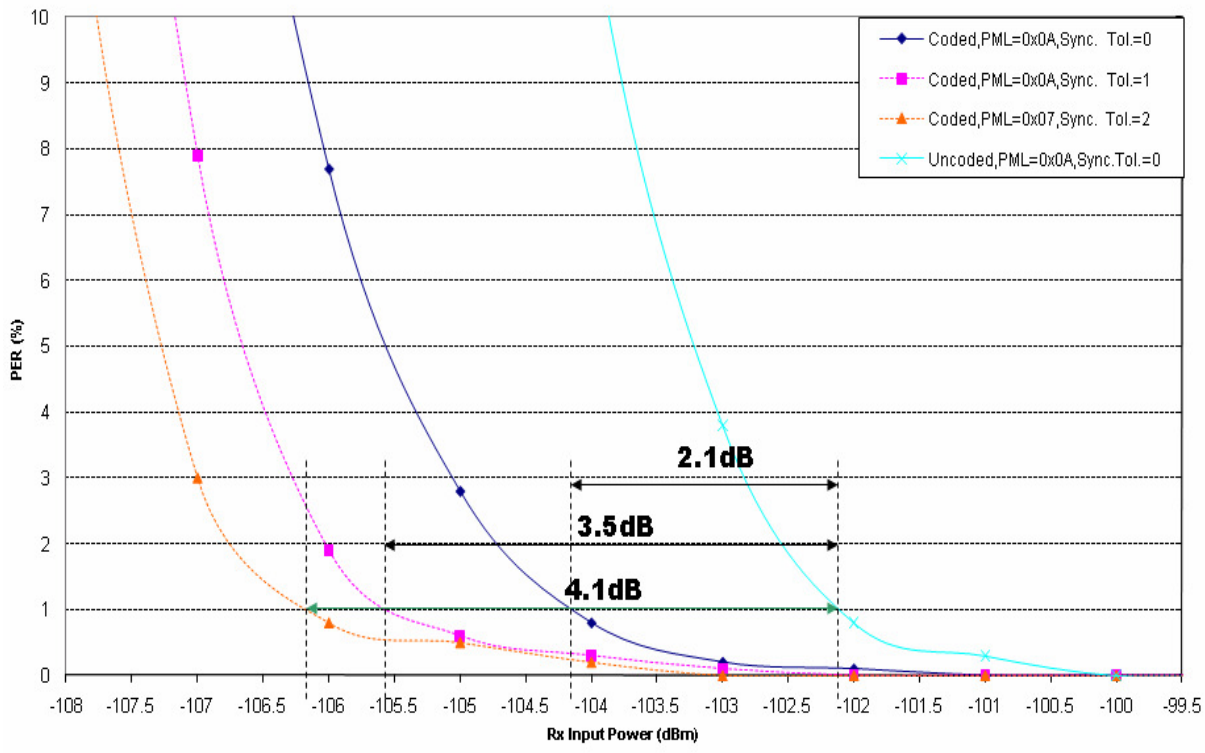

Figure 6 RS Code (38, 28.0.74, t=5), Packet Error Rate vs. RF Input Power
RF= 928MHz, FSK Data Rate = 100kbps
Synchronization Error Tolerance Impact on Coding Gain

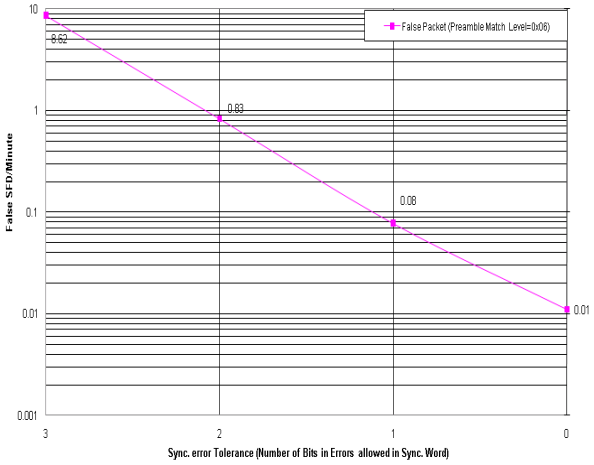

Figure 7. Sync Frame Detection (SFD) 1% error Rate & False SFD detect/ Minute versus Sync. Error Tolerance & Preamble Match level at 100 kbps

RS encode and decode timing measurements using various data bytes lengths are shown in Table IV and V respectively. Table IV shows that the encode time depends upon the length of the data bytes and increases linearly with the number of data bytes.

The RS decode time shown in Table V depends upon the encoded packet length and the number of bytes in errors. For example, RS code of length 128 bytes and t= 5 require 1.7 ms to decode while RS code of length 38 bytes is decoded in approximately 660usec.

Encoded Bytes *(n)*	**Encode Time (msec)**
32	0.34
38	0.43
64	0.8
128	1.7
210	2.87

Table IV. Reed-Solomon Codes Encode Time

Encoded Bytes *(n)*	**Time (µsec) t=0**	**Time (µsec) t=1**	**Time (µsec) t=2**	**Time (µsec) t=3**	**Time (µsec) t=4**	**Time (µsec) t=5**
38	141	476	524	571	614	659
128	350	954	1023	1090	1156	1221
210	540	1390	1471	1554	1645	1724

Table V. Reed-Solomon Codes Decode Time

Throughput of the RS code depends upon the time it takes to decode the received data. For example, Table VI shows the throughput of the receiver capable of correcting 5 bytes in errors, i.e. t=5.

Encoded Bytes Length *(n)*	**Data Bytes Length *(k)***	**Decode Time (µsec) *t*=5**	**Data Throughput (kbps)**
38	28	659	340
128	118	1221	773
210	200	1724	928

Table VI. Reed-Solomon Codes Throughput (kbps)

V. Conclusions

This paper presents the measurement results of Reed Solomon code using ADF7023 RF Transceiver devices. The use of RS FEC codes provides a significant improvement in link margin and communication range without a significant increase in transceiver power. The paper also shows an improvement in battery life-time using RS FEC coded system in comparison with uncoded system transmitting at higher power.

The theoretical RS coding gain of 4 dB can be achieved by optimizing the receiver's tolerance parameters PML, Sync. Tol. Due to the fact that these wireless nodes are in-active for extended periods of time in low power wireless networks, the receiver is less susceptible to false packet detection errors. This makes the use of RS FEC coding schemes very attractive in low power wireless networks.

Acknowledgements

The authors would like to thank T.O'Connell and V. Foley (Analog Devices) for the measurements support.

Reference

[1] E. R. Berlekamp, Algebraic *Coding Theory.* *M*cGraw Hill, New York, 1968.

[2] S. Lin and D.J. Costello, *Error Control Coding: Fundamentals and Applications.* Prentice Hall, 1983.

[3] S. B. Wicker, *Error control systems for digital communication and storage.* Prentice Hall, 1995.

[4] www.analog.com/adf7023

[5] T.Conway, R.Conway, K.Mulvaney, S.OMahony, M.K.Khan, C.Billon, H. Shanan and P.Quinlan. Low power application specific processor for ISM band transceiver, *Irish Signals and System Conference* (*ISSC 2010*), pages 272-277, University College Cork, June 23-24, 2010, Cork, Ireland.

[6] Guido Retz, Hyman Shanan, Kenneth Mulvaney, Shane OMahony, Miguel Chanca,Pat Crowley, Charley Billon, Muhammad Kalimuddin Khan, Jesus Lopez and Philip Quinlan. Radio Transceivers for Wireless Personal Area Networks Using IEEE802.15.4. IEEE *Communications Magazine*, pages 150–158, Sept 2009.