University of Huddersfield Repository

Al-Nedawe, Basman, Sibley, Martin J.N. and Mather, Peter

Microelectronic implementation of error correcting codes for dicode pulse position modulation

Original Citation

Al-Nedawe, Basman, Sibley, Martin J.N. and Mather, Peter (2010) Microelectronic implementation of error correcting codes for dicode pulse position modulation. In: Future Technologies in Computing and Engineering: Proceedings of Computing and Engineering Annual Researchers' Conference 2010: CEARC'10. University of Huddersfield, Huddersfield, p. 184. ISBN 9781862180932

This version is available at http://eprints.hud.ac.uk/id/eprint/9335/

http://eprints.hud.ac.uk/

Computing & Engineering
System Engineering
Research Group

MICROELECTRONIC IMPLEMENTATION of ERROR CORRECTING CODES for DICODE PULSE POSITION MODULATION

B. Al-Nedawe, Dr. M. Sibley, Dr. P. Mather

University of HUDDERSFIELD

ABSTRACT

Dicode pulse position modulation (DiPPM) was proposed by Sibley as a more advantageous format than digital PPM. DiPPM can be efficiently implemented as it employs two slots to transmit one bit of pulse code modulation PCM. Thus a PCM conversion from zero to one provides a pulse in slot reset (R) and from one to zero provides a pulse in slot set (S). No signal is transmitted if the PCM data is unvarying. The line rate is two times that of the original PCM, a significant reduction in speed compared to digital PPM. As the bandwidth requirement is much smaller than digital PPM, DiPPM could be used in dense wavelength division multiplexing (DWDM) systems. DiPPM systems suffer from three types of pulse detection errors wrong-slot, false-alarm and erasure. This project will use a field programmable gate array (FPGA), to build an error correction code circuit (Reed Solomon Code) and dicode PPM (coder & decoder). This will be expected to reduce the error sources in DiPPM. According to initial results the conclusion is that the Reed Solomon error correction coded system offers improvement over uncoded DiPPM, when operating at the approximately 4/5 code rate.

BACKGROUND

1. DICODE PULSE POSITION MODULATION SYSTEM

In dicode signalling data transitions from logic zero to logic one are coded as +V and transitions from logic one to logic zero are coded as -V. As shown in figure below, a zero signal is transmitted if there is no change in the PCM signal. The positive pulse can be regarded as setting the data to logic one (pulse SET), whereas the negative pulse resets the data to logic zero (pulse RESET). In dicode PPM, these SET and RESET signals are converted into two pulse positions in a data frame. Thus a PCM transition from zero to one produces a pulse in slot S and a one to zero transition generates a pulse in slot R. If the PCM data is constant, no signal is transmitted. (Although two guard slots have been used in this system, to reduce the effects of inter-symbol interference (ISI), this depends on the channel characteristics. If there is minimal ISI, zero guard slots could be used.)
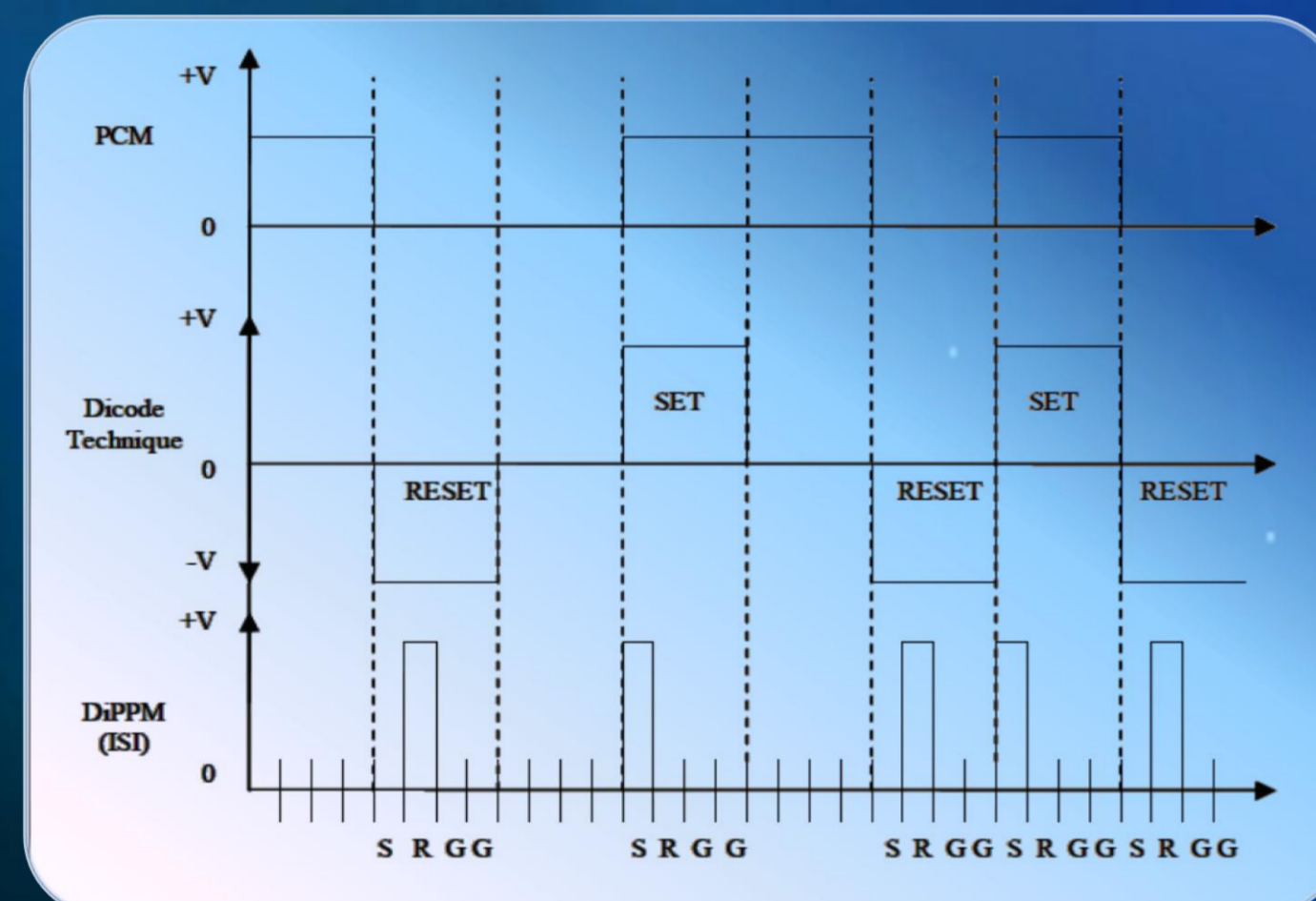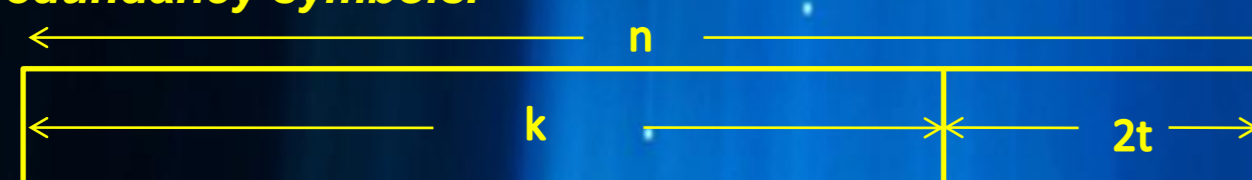
2. REED SOLOMON CODE

In 1960, Irving Reed and Gus Solomon published a paper in which they described a new type of error-correcting codes that are now called Reed Solomon (RS) codes. RS codes are a powerful class of non-binary forward error-correcting that can correct the maximum possible number of errors with the minimum data overhead. RS codes are the most commonly used error codes in practice. It is used in many applications like magnetic and optical data storage, wireline and wireless communications, and satellite communications. RS decoder has the ability to correct up to t symbols that contain errors in a codeword, where t can be defined as:

$$t=(n-k)/2$$

Where:
n= Number of codeword symbols.
k= Number of data symbols.
2t= Number of redundancy symbols.

AIM & OBJECTIVE

The main aim of this project research is to implement a DiPPM and Reed Solomon Code (coder & decoder) using FPGA to reduce the sources of error in DiPPM.
The following main objectives will be considered:
1) Finding the Reed Solomon (RS) optimum parameters.
2) Design RS encoder and decoder circuits.
3) Implement RS and DiPPM (encoder & decoder) circuit by programming the FPGA.
4) Confirm theoretical predictions with measurements.
5) Compare the result with another type of coding.

Sources of Error in DiPPM

DiPPM systems suffer from three types of pulse detection errors:

❖ **Wrong slot error**.
Wrong-slot errors occur when noise on the rising edge of a detected pulse causes the pulse to appear in adjacent time slots.
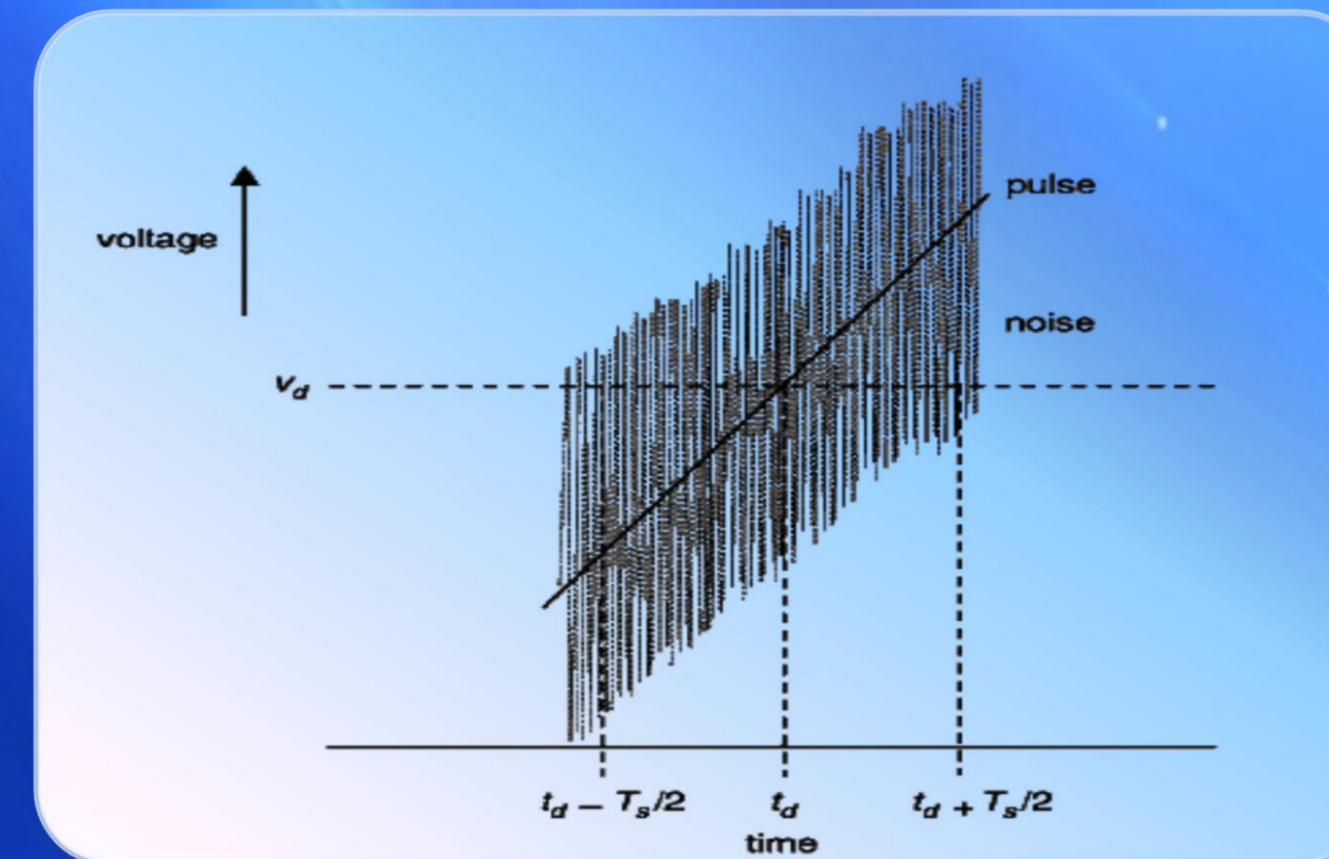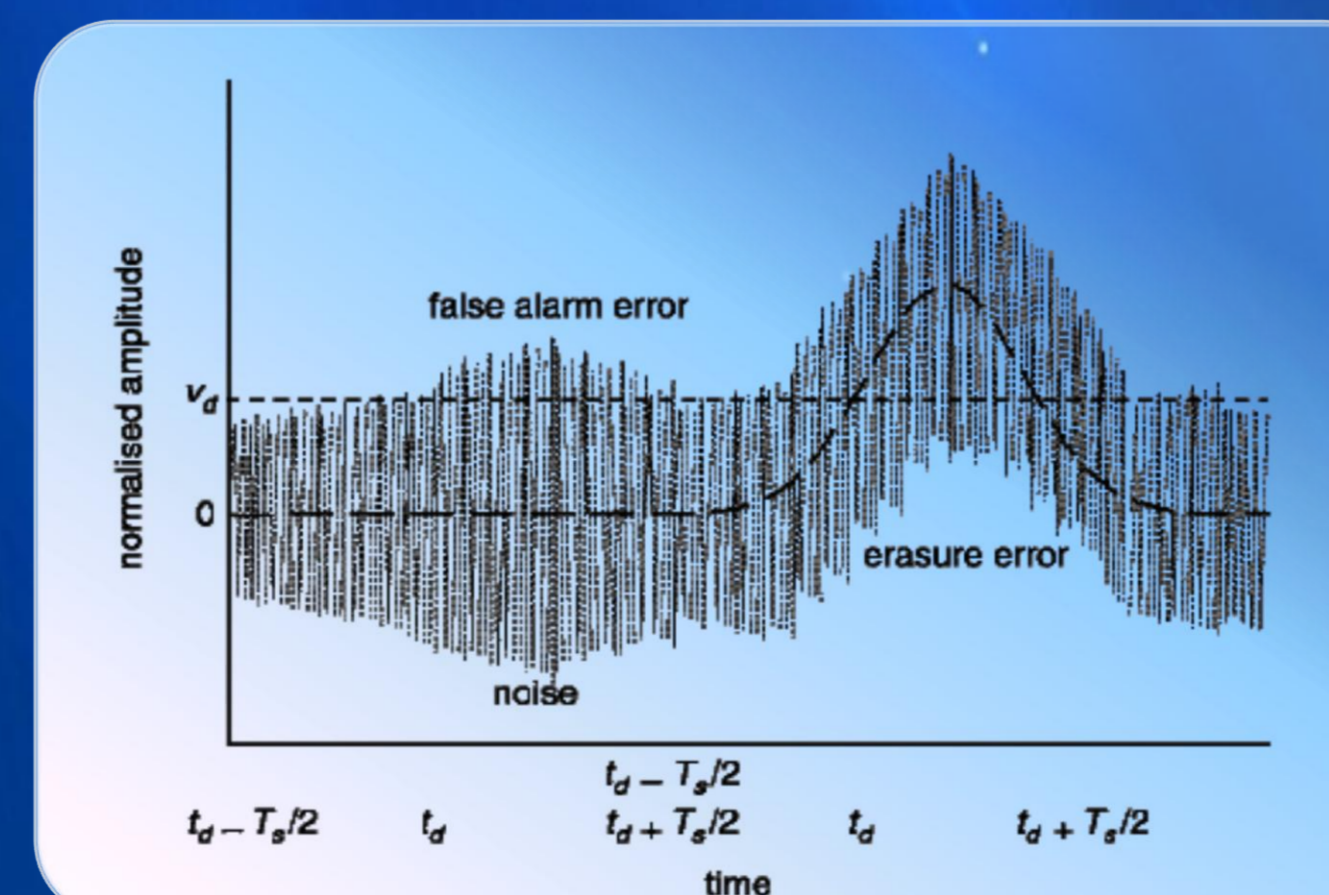
❖ **Erasure errors.**
Erasure errors occur when the noise is so great as to reduce the peak signal voltage to below the threshold level.

❖ **False alarm errors.**
Due to ISI, there can be a signal voltage in the slots surrounding the one containing a pulse. Noise on this voltage can lead to a threshold crossing event when no pulse is presented; so called false alarm errors.

SYSTEM MODEL

A system model tries to simulate some characteristics of a system. The model corresponds to the forward error correction (FEC) communication scheme, which is dependent on a RS error-control code. The behaviour of each block of the model is described in Mathcad software.
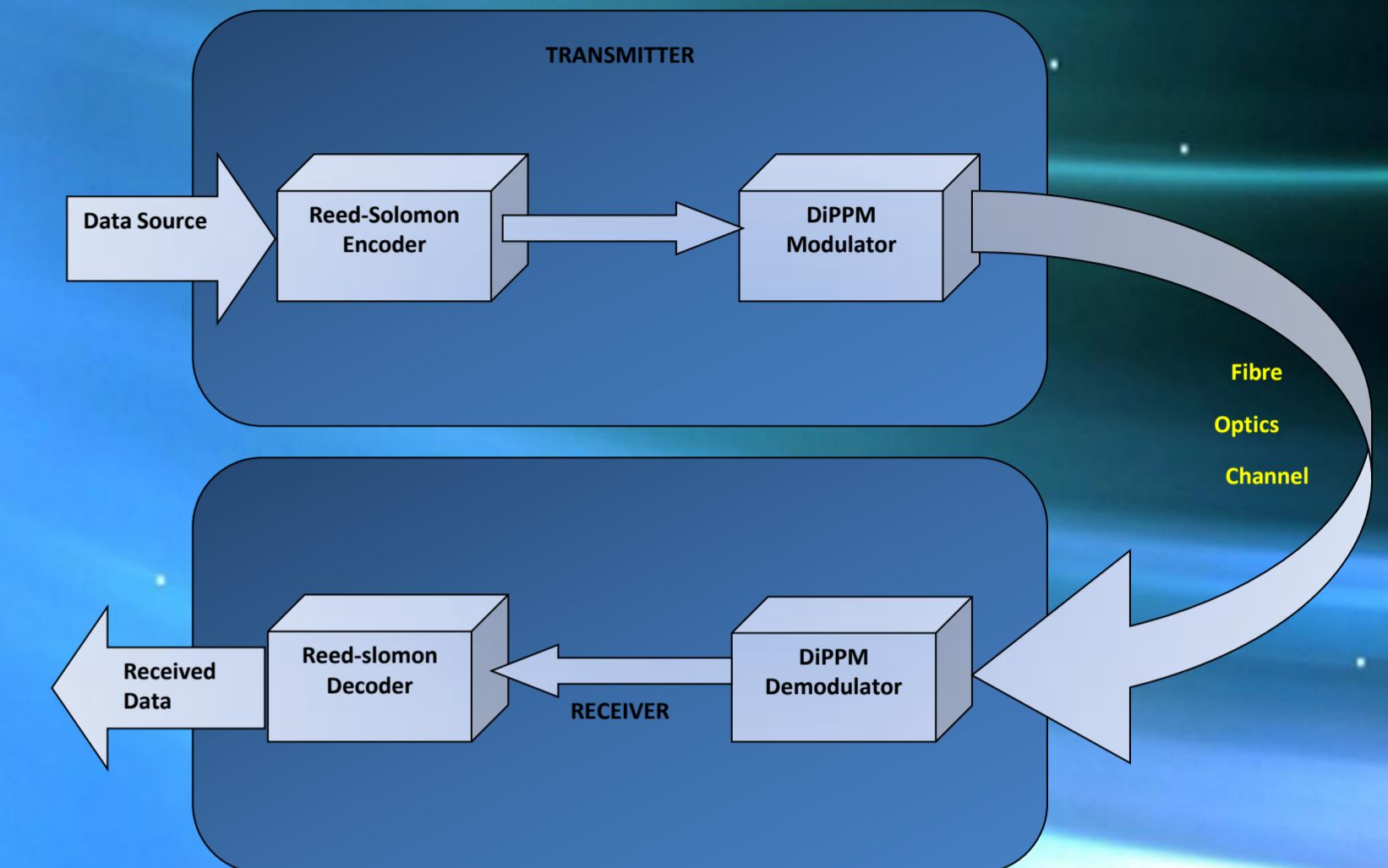
CONCLUSIONS

This research will concentrate on employing Reed Solomon (RS) codes to code the Dicode Pulse Position Modulation System (DiPPM). RS is expected to reduce the source of errors, which a DiPPM system suffers from. In this report a literature review has been written to explain the problems of DiPPM system and trying to find a solution: The general block diagram of the system has been given. According to initial results the conclusion is that the Reed Solomon error correction coded system offers improvement over uncoded DiPPM, when operating at approximately (4/5) code rate, and this will lead to a practical RS system (coder & decoder). Finally, the overall system will be implemented using FPGA and the results will be compared with maximum likelihood sequence detection (MLSD).

FUTURE WORK

❖ Determine a suitable algorithm for Reed Solomon Code.

❖ Design RS encoder and decoder circuits by choosing the proper parameters.

❖ Implement RS, DiPPM encoder and decoder circuit by programming the FPGA using Very high speed integrated circuit Hardware Description Language (VHDL).

❖ Verify theoretical predictions with measurements.