CDSP:2015
COMMUNICATIONS & DIGITAL SIGNAL PROCESSING

Graduate
Degree Level: Ph.D.

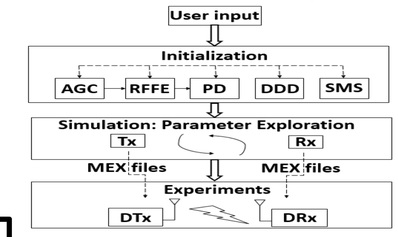
Northeastern University

Implementation of a MATLAB-based Self-Configurable Software Defined Radio Transceiver

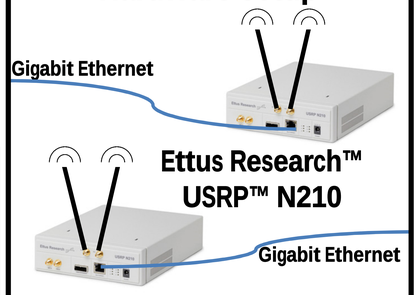
Benjamin Drozdenko (bdrozdenko@coe.neu.edu), Ramanathan Subramanian (rsubramanian@coe.neu.edu), Prof. Kaushik Chowdhury (krc@ece.neu.edu), Prof. Miriam Leeser (mel@coe.neu.edu)

Abstract

- Software defined radio (SDR) requires deep knowledge of the operating environment and low-level coding
- A bi-directional transceiver in MATLAB that allows automated selection of parameters
- Facilitates real-time operation and timing consistency using MATLAB Coder and MEX
- A model-based design, where state and block diagrams drive code development
- One machine (DTx) sends DATA, gets ACKs and another (DRx) gets DATA, sends ACKs



Hardware Setup



Algorithm & Code Structure

```
function dRxd=transceive(d2Tx){
  dRxd = step(hRx);
  step(hTx,d2Tx);
}
function main() {
  while !flagTerminal {
    dRxd = transceive(d2Tx);
    if (state==1.1) {...}
    elseif (state==1.2) ...
  }
}
```

Background

- *IEEE 802.11b* Physical (PHY) and Medium Access Control (MAC) layer frame structure with some modifications
- *Differential Binary Phase Shift Keying (DBPSK)* modulation scheme improves accuracy, requires only coarse frequency compensation without any closed-loop techniques
- *Universal Software Radio Peripheral (USRP)* has tunable oscillator and sampling rate
- *MEX:* MATLAB code, generated into C code, compiled into executable, called as function

Machine 1 — Designated Transmitter (DTx)

1. Energy Detection

1.1 Wait DIFS | 1.2 Detect Energy | 1.3 MAC Contend

1.1 DTx waits for a fixed interval of time before sensing the channel state.
1.2 DTx either backs off or transmits depending on whether the channel state is busy or not.
1.3 DTx contends for channel access.

64 bits					
PLCP SYNC (128 bits ≡ 2 frames)					
SFD		SIGNAL	SERVICE	LENGTH	CRC
Frm Ctrl	Dur/ID	Dest Addr		Src Addr	Sequence #
MAC Frame Body and FCS (16,128 bits ≡ 252 frames)					

2: Transmit DATA Frame

Entry: Prepare 802.11b DATA frame (256 USRP frames)
During: Prepare new USRP frame (64 bits ≡ 1408 samples)
Exit: Wait SIFS

2 CRC Gen → SMS → RCTF

3: Receive ACK Frame

3.1 Search SYNC | 3.2 Read Header

3.1 RFFE → SYNC Det
3.2 RFFE → DDD → CRC Det

Machine 2 — Designated Receiver (DRx)

1. Receive DATA Frame

1.1 Search SYNC | 1.2 Read Header | 1.3 Read Payload

1.1 RFFE → SYNC Det
1.2 RFFE → DDD → CRC Det
1.3 RFFE → DDD

2: Transmit ACK Frame

Entry: Prepare 802.11b ACK frame (4 USRP frames)
During: Prepare new USRP frame (64 bits ≡ 1408 samples)
Exit: Wait SIFS

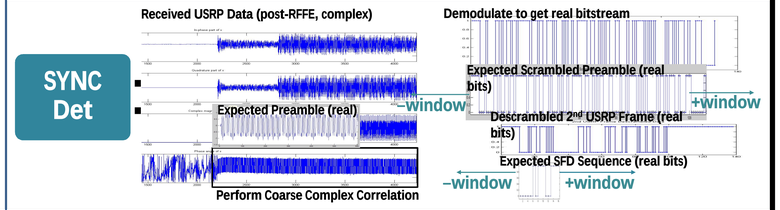
64 bits					
PLCP SYNC (128 bits ≡ 2 frames)					
SFD		SIGNAL	SERVICE	LENGTH	CRC
Frm Ctrl	Dur/ID	Rcv Addr		MAC FCS	

2 CRC Gen → SMS → RCTF

3: Wait DIFS

DRx waits for DCF Inter-frame Space (DIFS) duration before re-entering DRx State 1

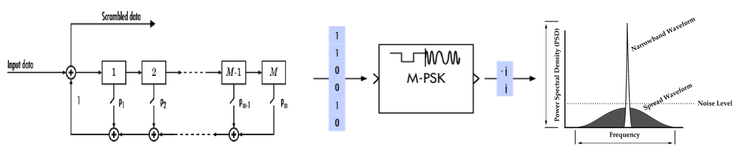
SYNC Det

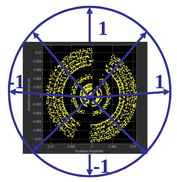


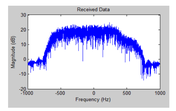
SMS : Scrambling → Modulation → Spreading

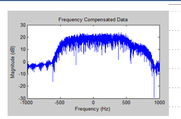
RFFE (RF Front End) : Automatic Gain Control → Freq Offset Estimation → Freq Offset Compensation → RCRF

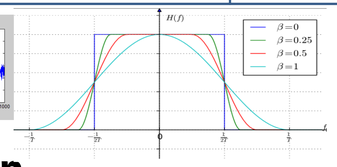
DDD : Despreading → Demodulation → Descrambling





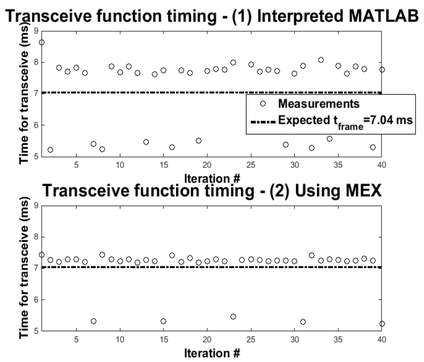


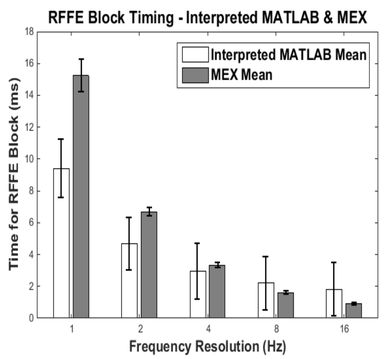




Results

- Transceive function execution times adhere more closely to expected slot time of 7.04 ms when compiled into MEX.
- RFFE Block execution times depend mostly on Frequency Resolution parameter; compiling into MEX provides no speedup due to large FFT sizes.
- AGC parameters control how well a signal can be recovered under various attenuation levels. By parameter sweep, a step size of 0.5 and an update period of 704 minimizes frame misdetection.





Conclusion

- System designed and built with slot time-synchronized operations adheres to our desired frame time and is able to reconfigure parameter values as needed.
- Using MEX is essential for realizing timing with little deviation from the frame time. In addition, MEX improves the speed consistency of our system blocks, most notably RFFE, which can vary its frequency resolution parameter.

Future Work

- Complete design of the MAC functions.
- Implement transceiver system design on the Xilinx Zynq-7000 System-on-Chip (APSoC).

References

[1] I. F. Akyildiz, S. Mohanty, M. C. Vuran, and V. Won-Yeol, "NeXt generation/dynamic spectrum access/cognitive radio wireless networks: A survey," Computer Networks, vol. 500, no. 13, Sept. 2006.
[2] Ettus Research, Inc., "USRP N200/N210 Networked Series."
[3] IEEE Std 802.11-2009, "Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications."
[4] MathWorks. "Comm. System Toolbox Documentation."
[5] T. Collins, "Multi-Node Software Defined Radio TestBed". NEWSDR 2014.
[6] J. Mitola III and G. Q. Maguire, Jr., "Cognitive radio: making software radios more personal," IEEE Personal Communications Magazine, vol. 6, nr. 4, pp. 13–18, Aug. 1999.

Acknowledgements

This work is supported by MathWorks under the Development-Collaboration Research Grant A#: 1-945815398. We would like to thank Mike McLernon and Ethem Sozer for their continued support on this project.