



**HAL**  
open science

# FPGA Based High Data Rate Radio Interfaces for Aerospace Wireless Sensor Systems

Julien Henaut, Daniela Dragomirescu, Robert Plana

► **To cite this version:**

Julien Henaut, Daniela Dragomirescu, Robert Plana. FPGA Based High Data Rate Radio Interfaces for Aerospace Wireless Sensor Systems. The Fourth International Conference on Systems (ICONS 2009), Mar 2009, Gosier, France. p.173-178. hal-00469481

**HAL Id: hal-00469481**

**<https://hal.science/hal-00469481>**

Submitted on 6 Apr 2010

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# FPGA Based High Data Rate Radio Interfaces for Aerospace Wireless Sensor Systems

Julien HENAUT

LAAS-CNRS

Université de Toulouse  
7, avenue du Colonel Roche,  
F-31077 Toulouse, France  
jhenaut@laas.fr

Daniela DRAGOMIRESCU

LAAS-CNRS

Université de Toulouse; INSA  
7, avenue du Colonel Roche,  
F-31077 Toulouse, France  
daniela@laas.fr

Robert PLANA

LAAS-CNRS

Université de Toulouse; UPS  
7, avenue du Colonel Roche,  
F-31077 Toulouse, France  
plana@laas.fr

## Abstract

*This paper gives an overview of development and prototyping of a radio interface designed for high data rate Wireless Sensor Networks (WSN) using a top down approach. An aerospace application of this work is presented to underline the importance of rapid and flexible prototyping. Single Carrier Frequency division multiple Access (SC-FDMA), with a very low Peak Average Power Ratio (PAPR), is a promising alternative to the OFDM scheme for highly power constraint applications, however, it is still not well explored. The chosen radio interface will be tested by means of Electronic System Level (ESL) synthesis and Field Programmable Gate Array (FPGA) prototyping. The ESL development is based on the tool "Synplify DSP", which allows quick prototyping and makes it possible to do real measurements, thus increasing the knowledge about the propagation channel to update the model and algorithm.*

## 1. Introduction

To evaluate a system's compliance with its specified requirements, Hardware System Testing is conducted on the complete and integrated system. This phase is essential in all industry branches, especially in the very regulated and critical aerospace world.

In the final phase of the development of an airplane, flight test equipment gathers and analyzes data during flight to evaluate the flight characteristics of the aircraft and validate its design, including safety aspects. One of the most critical tests is the measure of the pressure around the wings during flight. All new aircrafts are computer designed with the use of virtual wind tunnels. So very accurate measures have to be done on the aircraft to validate the model before the aircraft can be industrially produced.

In the case of satellites, vibration and mechanical stress are two critical phenomena a satellite endures

during launch. This is leading to the necessity for accurate ground tests using strain gauges or thermal sensors before allowing a launch.

All such systems used by aircraft and satellite manufacturers today are wired systems. Sensors put around the wings or inside the satellite compartments are wired to a concentrator inside the cabin or the operator's room. Although good performances are observed in terms of measurement accuracy, these systems have strong drawbacks. The two most important ones are the weight and the cost of both the systems and their installation. An additional drawback concerning its use on aircrafts is due to the installation of a system that increases the weight of the aircraft and immobilizes it during many weeks due to the routing of every cable inside the wings. The cost and the complexity of such systems don't allow a great number of measurement points. This is why research is done on a wireless measurement system based on wireless sensors networks.

This paper is organized into five sections. Section II introduces the specifications for WSN applications. Once the requirements are clearly identified, a top down design methodology (section III) and the reasons why it has been chosen (section IV) are presented. The model that allows quick FPGA implementation is presented in section V and the results for different FPGAs in section VI. Finally conclusions are outlined in the last section.

## 2. Requirements

One of the first points of an application-centered development is the examination of the user's needs. This point is even more important when the specifications introduce very strong constraints. In our case, a large number of unknowns oblige us to choose a rapid development cycle to increase our knowledge of the application's environment. From the vast among

of specifications four major constraints have been extracted.

First of all, the wireless systems should allow an increase in the number of points of measure. Thus, the first specification for both users (aircraft and satellite manufacturers) is to dispose of more than 400 points of measure. In addition to the large amount of data points, aerodynamicists need frequently (every few microseconds) updated data. Thus, the application must provide a very high data rate. Thirdly, if one measurement point gets lost, the whole data set recorded during the test becomes useless. No loss can be tolerated, the transmission therefore has to exhibit an extremely low error rate. Finally, as the wireless sensors are spread on aircraft wings or inside a satellite, they have to be self-powered. This limitation introduces severe constraints in the design of wireless sensor nodes. Due to the high data rate the node has to communicate constantly and one cannot introduce a sleeping mode to minimize the power consumption of the radio interface. As the radio interface remains one of the bottlenecks to implement truly low-power WSN, designers have to rethink the entire radio architecture. An outline of the issue of high data rate in WSN is presented in [1].

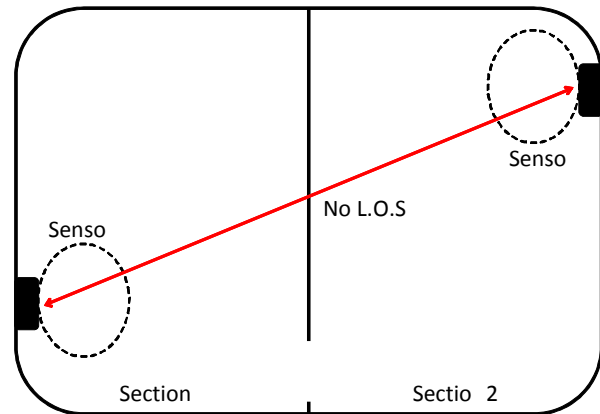
To summarize wireless measurement allows much cheaper and easier deployment than wired sensor measurements. However, the constraints around the application are so strong and critical, that a new design methodology is indispensable.

### 3. An Efficient Top Down Approach

In the design of a new wireless communication system, the first thing to characterize is the propagation channel, yet, no study of such a system could be found in literature. Also the modeling is quite involved, because of the complexity of the structure, the environment at 12000 m and the presence of metal mass. The link is made between two sensors (with thin antennas) stuck on the wing. The channel is open so there is no multi path propagation. But as the link is made across the wing, the measured attenuation of patch antennas in the line of sight is -20 dB at the emitter and -20dB at the receiver. So there is -40dB of attenuation without any considerations of distance. The influence of this kind of channel on phase modulation in an ultra low power system have not been studied.

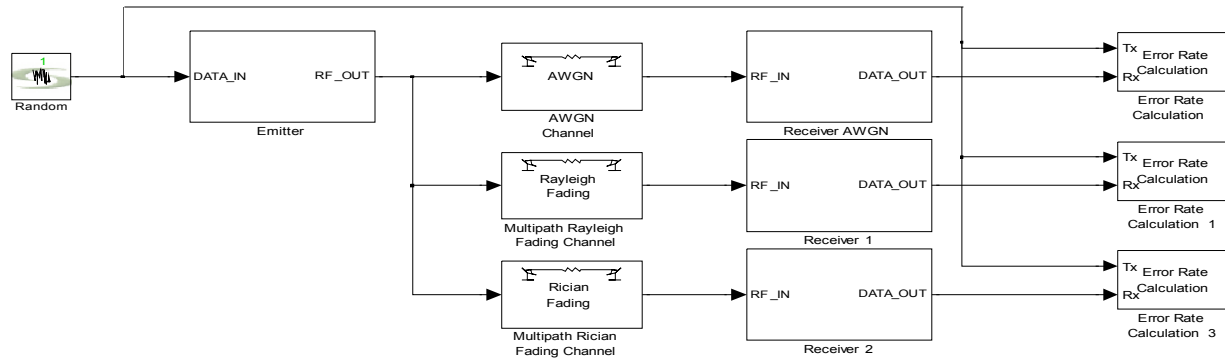
The channel inside a satellite, where closed space propagation takes place, has been studied only recently [2]. There are multiple propagation paths and the more the delay spread is long the lower the maximum data

rate is. The radiation characteristics of the antennas are not problematic if the cavity is considered as multimode (figure 1). This is the case when the side length of the cavity is higher than the wavelength [2]. Neither the impulse response of the inside of a satellite nor the impact of numerous sources in this kind of environment has been studied yet.



**Figure 1: Communication between two patch antennas inside the satellite**

To achieve the specifications, ultra wideband modulation has to be considered and, in both cases, their behavior is not well known. To suppress this lack of knowledge, measurements are needed. They can be performed with equipment such as pulse generator, OFDM modulator and High Bandwidth Oscilloscope. However, these equipments use standardized modulation schemes. As in our application, some key specifications like the number of elements or the achievable error rate cannot be satisfied by existing communication standards. We have to develop our own test equipment that has to be reconfigurable in order to assess different algorithms. To sum up, a very quick development cycle is needed, with easy feedback between the realization and the model. This process will help two different parties: The one working on algorithms can then validate each of their choices with physical data and measurements and the one working on the channel model can use the developed transceivers to increase their knowledge and improve the model. The choice of the radio interface can be done while taking into account both channel and environment.



**Figure 2. Model for the evaluation of the emitter and the receiver**

As an inexpensive solution with short development time is targeted, Field-Programmable Gate Arrays (FPGAs) present a suitable implementation choice. Their advantages include the ability to be re-programmed once the test is performed. But, they draw more power than Application-Specific Integrated Circuits (ASIC). Thus, the real power consumption of the wireless sensor network cannot be evaluated using FPGAs.

The present design is developed on an FPGA, and is then transferred to an ASIC to evaluate the real power consumption. Thus, the design flow has to be both compatible with FPGA prototyping and ASIC development.

The complexity of an ultra wideband radio interface is high especially due to digital signal processing elements (like Direct or Inverse Fast Fourier Transforms: FFT/IFFT). To simplify the design of complex systems in FPGAs, there exist libraries of predefined complex circuits that have been tested and optimized to speed up the design process [3]. These predefined circuits are called Intellectual Property (IP) cores. They are often very expensive which could be a decisive drawback if many functions have to be tested. Furthermore, all the cores are at Register Transfer level (RTL) level. So, each of them is very well optimized but put together in a complex algorithm the result could be a disaster in terms of timing. In our critical application, this IP approach cannot be satisfying. In an attempt to reduce the complexity of the design process to achieve previously detailed requirements, we will raise the abstraction level of the design.

For all these reasons, the only development technique feasible is an Electronic System level (ESL) synthesis. ESL Synthesis refers to tools and methods enabling high-level descriptions and automatic efficient RTL generation. This allows a top down approach based on the specifications mentioned before.

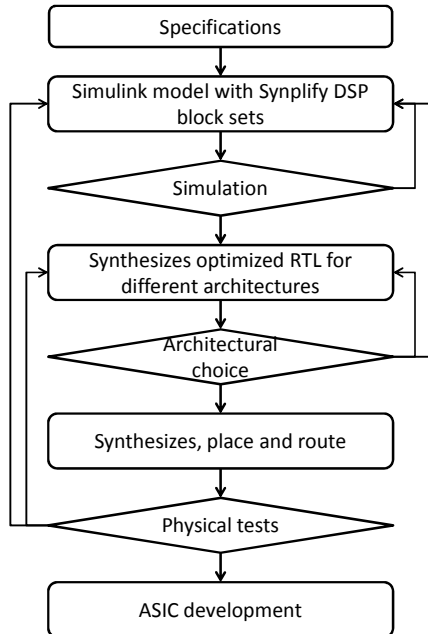
#### 4. Rapid Prototyping with Synplify DSP

The software “Synplify DSP” from Synplicity is a powerful ESL synthesis tools. It automatically creates optimized logic implementations for both FPGAs and ASICs. A good description of this tool can be found in [4] and [5]. The Synplify DSP software combines a high-level modeling library offering multi-rate and vector functions with a synthesis engine. The modeling library is used in Simulink (from The Mathworks).

The first step of the ESL synthesis development is the construction of an algorithm in a Simulink model file. Even if blocs are not synthesizable, they can be used to develop the model. A good example is the use of a channel model Additive White Gaussian Noise (AWGN, Rayleigh, Riceian or ultra WideBand UWB channel) to evaluate the performances of the algorithm before implementation. In figure 2, a model for the evaluation of the emitter and the receiver design with Synplify DSP blocks and a Bit Error Rate measurement in the three channels is given. The RTL generator will use the quantization and sampling rate of each block to determine the clock frequency needed for the FPGA. Then, the Synplify DSP synthesis engine can synthesize the design. The RTL produced (VHDL or Verilog) must be processed using the standard logic synthesis and the place and route flow. When the design is implemented on the FPGA, measurements and complementary tests can be done to improve the model. This design flow is summarized in figure 3. One of the most interesting parts in the ESL design flow is the possibility of exploring many different architectures. The synthesis is defined for a target device and optimizations depend on this target. One can quickly explore a variety of architectures and target technologies. Very low power implementations with a complex design are targeted here. So a choice has to be done between low power FPGAs like Igloo family from Actel, “simple FPGAs” like Spartan III from Xilinx or more powerful FPGAs like Virtex IV or

V from Xilinx. With the ESL synthesis all these FPGAs could be tested and show how they offer significant tradeoffs in speed and area. Some results of these comparisons are presented in the last part.

Although the Synplify DSP user interface seems to be very simple, the possibilities of optimization are huge and constitute the power of this tool. In addition to VHDL optimizations, Synplify DSP will also automatically explore more fundamental micro-architectures at the block level and produces robust VHDL code, and not just several instantiated cores.



**Figure 3. Design flow to decrease time between development and physical tests**

In addition to the baseline architecture (optimizations are done only at block level), the program also performs more complex optimization.

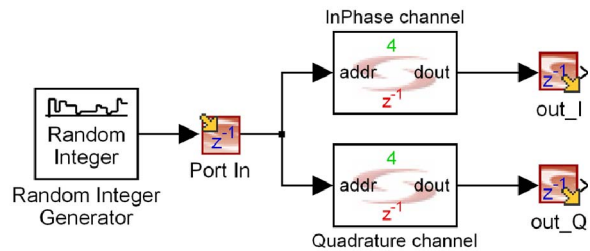
The first one is called retiming. It consists in the insertion of additional pipeline stages, if needed, to meet the sampling rate constraints. This operation increases the size and power consumption due to the addition of buffers to the design. Retiming modifies the architecture to stick to the performance goals. It is very useful when facing high speed requirements.

The second option is called folding. It consists in serialization using faster clocks with resource sharing and scheduling. The folding option allows the design to share hardware at the expense of lower throughput. However, it can often result in large area reduction. This option is helpful when the size of the design is a critical point. With the graphical interface, users can choose different level of folding or retiming on different platform and then find the best tradeoff for

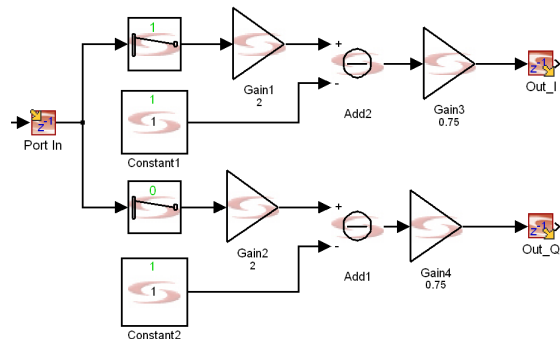
their application. We save months of hand coding for pipelining and folding by automatic optimizations. The ESL synthesis with Synplify DSP approach allows us to optimize the model at high level, to choose the best FPGAs and to perform physical tests in parallel.

## 5. The Model

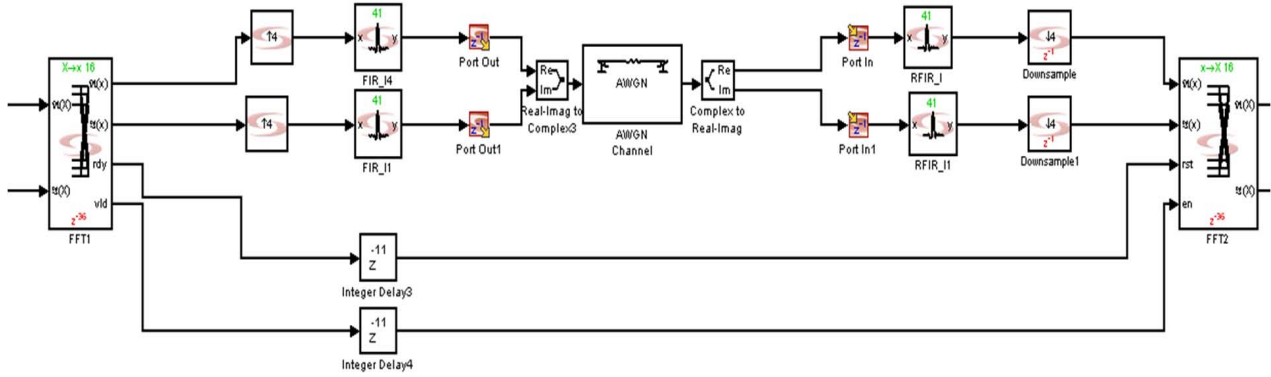
The first step in our development is the definition of a modulation scheme. In [1], two modulation schemes used for high data rate are compared: Multi carrier approaches, represented by the popular Orthogonal Frequency Division Multiplexing (OFDM) and single carrier methods represented by Single Carrier Frequency Division Multiple Access (SC-FDMA). SC-FDMA, with a very low Peak Average Power Ratio is a promising alternative to OFDM scheme for highly constrained application. But, as detailed in [1], the transceivers for OFDM and SC-FDMA are almost the same and it is very easy to switch between the two. We chose to develop an OFDM transceiver model, and then to convert it to SC-FDMA. This paper will present the first part of this process with the presentation of the first OFDM implementation in figure 5.



**Figure 4a. Model of the Q-PSK modulator by using Rom memory cells**



**Figure 4b. Model of the Dual Channel Q-PSK modulator by using ADD and Gain cells**



**Figure 5. Model of the transceiver without the sources and the Q-PSK modulator on each ways**

The heart of the design is a sixteen point Inverse Fast Fourier Transform (I-FFT), which does the multi carrier modulation. The extremely low power requirements in our application demand a careful design for this block that takes into consideration the processing speed and the numerical precision. Sherratt [6] models the UWB transmitter in order to ascertain the minimum requirement for numerical processing. He concludes that the FFT processing (as the IFFT processing in the transmitter) requires only 6 bits of external and 12 bits of internal numerical precision to the FFT. Here, we have decided to design the emitter with a full precision on both internal bits and external bits. The size of the blocks is huge but the chosen FPGA can do the complex signal processing. This leaves us the possibility to evaluate physically the impact of this choice and adjust the precision.

The size (16 point I-FFT) has been chosen to be able to evaluate the feasibility and performances of the modulator with a simple IFFT and is not a limitation because the design flow allows to change the IFFT's size very easily. Some comparisons between IFFTs of different sizes regarding frequency and area will be done soon.

The IFFT is fed by 16 Quadrature-Phase Shift Keying (Q-PSK) modulators having two possible architectures presented in figure 4a and figure 4b. A random bit generator will generate the two bit code words that are mapped following gray coding. The modulator could be done either by two ROM blocks representing In Phase (I) and Quadrature channel (Q) or by a succession of Add and Gain blocks. The results for the maximum achievable frequency and the size of the two architectures are presented in table I. The second architecture (Figure 4b) has been chosen, because the number of equivalent gates (linked to the size) is lower.

**Table I: Performances summary of the two architectures of the QPSK modulator**

Architectures	Achieved frequency	LUTs used before place and route job	Gates used after place and route job
Ram blocks	450 MHz	2	107
Dual channel	450 MHz	0	67

By using 16 channels with a modulator for each one, the data of each way is sent on one of the 16 carriers. Any of these modulators can be changed to send data on a less complex modulation scheme, like Binary-PSK to ensure a better transmission, or on a more complex scheme, like 8-PSK, to allow a higher data rate. After an up-sampling by four, the model adapts a pulse shape to limit the bandwidth. The digital-to-analog conversion (DAC) of the I and Q signals and the mixing with the sine wave carrier are performed outside the FPGA (the outputs are defined with the Port\_Out blocs visible on the figure 5). The demodulator uses almost the same blocks. Two matched filters, using down sampling by four, feed a 16 point FFT block which performs the demodulation. The demapping is performed by state machines which are defined very easily with Synplify DSP tools.

## 6. Results for the Emitter Design

As the model presented is developed with Synplify DSP tool, many architectures and target technologies could be explored. The table II summarizes all the data collected for baseline implementations (without optimizations). Without optimization, the emitter could not be tested on an FPGA, even with very powerful ones like Xilinx Virtex5. To meet the timing constraints, Synplify DSP offers a retiming optimization that inserts new registers along the data path, moves existing ones and maintains consistent latency across parallel paths. The results are presented in table III.

**Table II: Performances summary for baseline implementations on different architectures**

FPGA	Speed	Max Estimated Frequency (MHz)	Achieved Freq. (MHz)	Speed Achieved?
Virtex2	-4	200	109.6	NO
Virtex4	-11	200	83	NO
Virtex5	-1	200	110	NO
Virtex5	-3	200	136	NO
Stratix3	-3	200	162	NO

**Table III: Performances summary for retiming implementations on different architectures**

FPGA	Speed	Max Estimated Frequency (MHz)	Lat. Add.	Achieved Freq. (MHz)	Speed Achieved?
V2	-4	200	16	162.1	NO
V4	-11	200	16	255.5	YES
V5	-1	200	16	309.6	YES
V5	-3	200	16	395.8	YES
Stratix3	-3	200	16	190.9	NO

After the retiming procedure, the estimated frequency could be reached on Virtex5 or Virtex4 FPGA. But as the Virtex 4 has few DSP48 elements, the number of Lookup Tables (LUT) needed in the FPGA is around three times the effective LUTs in the Virtex 4. So, some folding factors are added to reduce the area. Table IV presents the result of both folding and retiming optimization on a V4 and V5. The folding factor allows to use less LUTs, but not to reach the targeted speed of the system.

In summary the architectural (baseline) and micro architectural (retiming) optimizations together enable a quick implementation based on our model and result in the choices of using a Xilinx Virtex5 FPGA and a fully parallel architecture with extra latency of 16. Using multi-channel techniques will increase the frequency and so is not considered here.

**Table IV: Performances summary for folding implementations on different architectures**

FPGA	Max Freq. (MHz)	Folding Factor	Achieved Freq. (MHz)	LUTs
V5	200	0	309.6	23958 (83%)
V5	200	1	119.6	2962 (10%)
V5	400	2	132.8	4428 (15%)
V5	600	3	85.1	2986 (10%)
V4	200	1	103.2	9484 (77%)
V4	400	2	201	4264 (34%)
V4	600	3	175.6	3885 (31%)

## 7. Conclusion

When developing innovative communication systems, the quicker we have physical implementation of the transceivers, the more optimizations can be done. Even in a research projects, the time to development has to be reduced. In contrast to industry, one doesn't speak of "time to market" but of "time to improvement". ESL synthesis gives time to develop more powerful algorithms with physical considerations and measurements. The hand coding time is reduced and when the physical results are not satisfying, developers can modify algorithms. Hypothetical analysis can be put into practice very easily. This method has been applied to a highly constrained aerospace application based on wireless sensor network. By using a high level modeling that allows multi architecture tests, the results of our simple model shows the feasibility of an OFDM system. Future work will concentrate on the design of the SC-FDMA model and quantify the improvement compared to the OFDM modulator.

## 8. References

- [1] Henaut, J. Lecointre, A. Dragomirescu D. et al., "Radio Interface for high data rate wireless sensor network", Proc. of the 6th International Conference on Computing, Communications and Control Technologies (CCCT), Orlando FL USA, 2008, pp 104-109.
- [2] Terral, M. Pourtau, J.C., 'Reducing EMC verifications for future telecommunication Platforms, Assessment of oversized cavity theory limits synthesis', Alcatel Alenia Space Unclassified Document, 2007.
- [3] L'Insalata, Nicola E. Saponara, S. et al., "Automatic Generation of Low-Complexity FFT/IFFT Cores for Multi-Band OFDM Systems", Proc. of the 10th Euromicro Conference on Digital System Design Architectures, Methods and Tools, Washington DC USA, 2007, pp 361 – 368.
- [4] www.synplicity.com [Accessed 12/01/08]
- [5] Balakrishman, S. Eddington C., 'Efficient DSP algorithm development for FPGA and ASIC technologies', Proc. of the international conference on very large scale integration, 2007, pp 168-171.
- [6] Sherratt, R. S. and Makino S, "Soft-bit and Numerical Precision Requirements on Multiband Full-Rare Ultra-Wideband (UWB) Receivers", Proc. Of the 8th IEEE International Symposium on Consumer Electronics, 2004, pp 257-261.