Hardware Platform based on TMS320C6416 for Wireless LAN (802.11g)

N. Blanco[*], *D.Exposito*[*],*O. Venard*[**], *C.Ripoll*[**], *G.Baudoin*[**]

[*]Universidad de Valladolid – Facultad de ciencias - Dpto. Ing. Sistemas y autimatica
47011 Valladolid
{nuriablancod,davi_ex}@hotmail.com

[**]ESIEE-ESYCOM, Signal Processing and Telecommunications Department
BP99, 93162 Noisy Le Grand CEDEX
{o.venard,c.ripoll,g.baudoin}@esiee.fr
fax : +33-1-45-92-66-99 , tel : +33-1-45-92-66-91

This paper presents the development of an hardware platform made of a Test and Evaluation Board embeding a TMS320C6416 clocked at 500MHz, a daughter board allowing sampling frequency up to 80 Ms/s which has a FPGA, XCV50E, that could be used for high speed signal processing on the samples. The analog front end is realized with evaluation boards from different manufacturers. We also present the software architecture of an OFDM application with emphasis on the data transfer management and use of the hardware resource of the TMS320C6416, especially the EDMA.

The final realization is fruitfull either for test bench of algorithms such as presdistortion or for the test of new architecture of the Emitter or for educational purpose, because each elementary function either at radio frequency or digital baseband are split in specific hardware.

This work has been done thanks to an ELITE project sponsored by Texas Instruments.

INTRODUCTION

This paper introduces the design of a generic software design radio platform for wideband wireless transmission. We will focus on WLAN-ISM application, typically OFDM modulation system such as 802.11g. This platform has provision to be used for pre-distorsion to linearize power amplifier behaviour for wide dynamic range wideband modulated signal such as ODFM.

The platform is based on a TEB6416 board from Texas Instruments, an AD/DA board AED101 from Signalware and an analog front-end made from evaluation-board from different manufacturers (RF microdevices, Silicon labs,…).

This paper is organized as follows : we present the global architecture of the emitter-receiver in the first part, in the following section we introduce the main analog building blocks of the analog front-end and how they can be parameterized by the DSP board. We then describe the implementation of the base-band modulation on the DSP. Afterwards we conclude and give some perspectives.

ARCHITECTURE OF THE EMITTER-RECEIVER

The chosen architecture is the traditional super-heterodyne with IF at 374 MHz. Although it is not the state of the art architecture for software radio [4], this architecture is suitable for testing on wide-band modulation : OFDM, MC-CDMA signal processing algorithm such as digital baseband pre-distortion [1].

The emitter-receiver runs either in emitter (Tx) or in receiver (Rx) mode. The digital base-band modulation is implemented thanks to a TEB6416 board which embeds a 16 bits fixed point VLIW DSP running at 500 MHz. The DSP board is connected to a two channels AD/DA daughterboard allowing I/Q sampling at 80 Ms/s, that would be necessary for spectral regrowth due to pre-distortion processing. Furthermore the daughterboard embed digital IO that will be used to control the analog front end (AFE).

The IF stage have two main path : the Tx/Rx path (bottom of fig. 1) and the feedback path (top of fig. 1).

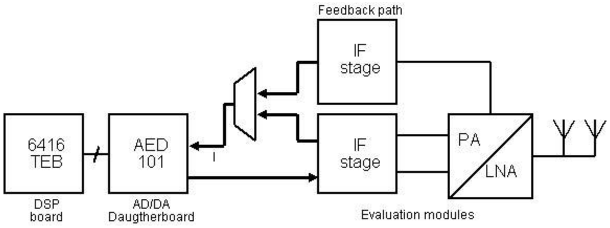


Fig. 1 Overall architecture of the proposed platform

In Tx mode, the Tx/Rx path is used in transmit mode, the base-band I/Q signal is taken at the output of DA converters and send to the power amplifier through IF stage. The output of the PA is send to the antenna and to the feedback path thanks to a coupler. This feedback path will be used for pre-distorter algorithm through IF stage and AD converter. In Rx mode the best antenna is chosen depending on received power and the signal is send to AD converter through low noise amplifier (LNA) and IF stage

ANALOG FRONT END

In this part we will focus on Tx mode. We use the RF2948B spread spectrum transceiver from RF microdevices together with a RF5117 power amplifier which have a saturation output power of 30dBm (fig. 2). Synthesis of RF and IF frequency is done thanks to Si4136 from Silicon labs. The transceiver provide the following functions : PSK modulator at IF frequency and up-conversion to RF frequency.

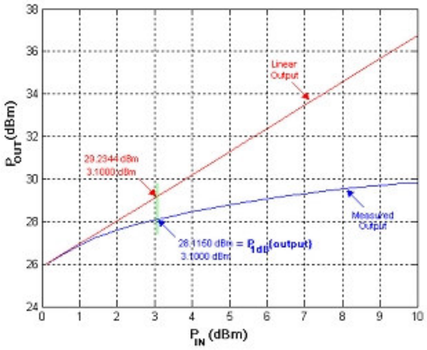


Fig.2 AM/AM curve of the PA in its non linearity region

This AFE is highly programmable: the DSP can set the following values through the daughterboard:

- The Power Amplifier (PA) power can be provided internally or driven externally.
- The PA gain can be set either externally or internally drive gain from 22dB to 26dB in 256 steps.
- Gain of the IF transceiver stage is programmable from -3dB to 13dB in 256 steps
- The Low-Noise Amplifier (LNA) gain can be set either to –6dB or 10dB.
- Gain of the IF receiver stage is adjustable between 0dB to 70 dB with 256 steps.
- Other parameters can be set by software: (antenna selection, …).

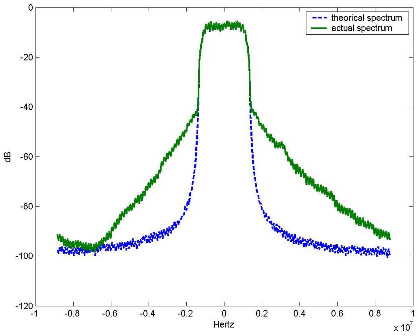


Fig.3 Theoritical and actual OFDM spectrum

For better power efficiency, we would like to use PA above its linear behaviour. But non linearities of PA widen the bandwidth of the modulation (figure 3) and increase the adjacent channel power ratio (ACPR). Digital predistorsion is a mean to avoid such drawback [1,2]. But this widen the spectrum in base band implying higher sampling frequency, almost 80 Ms/s in case of 10 Mhz bandwidth OFDM.

DEVICE DRIVERS

All the function needed to drive this Analog Front End have been embedded in a device-driver library written in C. The device driver functions have been embedded in a library and are split in two classes, primary functions and configuration functions. Primary functions have to change the value of some Daughterboard's registers in order to control the AFE components. Configuration functions make use of the primary functions to establish the correct program flow in each situation:

- Configuration functions are:
  - *Void Init_CtrlBoard(void)*: Mainly sets all the RF components in their default mode.
  - *void Rx_Config(void)*: Sets the control board in a reception configuration. This is the first function to call when the system is going to receive data.
  - *void Tx_Config(void)*: Sets the control board in a transmission configuration. This is the first function to call when the system is going to transmit data.
- Primary functions are:
  - *void Select_Mode (Op_Mode selection)*: Selects operation mode (Rx or Tx).
  - *void Select_Antenna (Antenna_Number Antenna)*: Selects which antenna will be used.
  - *void Select_Pa_Vcc (PA_Vcc Supply)*: Selects PA power supply.
  - *void Select_PA_Gain (Gain_PA mode_gain, Action_Gain power, int steps)*: Sets the PA gain.
  - *void Select_Tx_VGC_TranscPA (Action_Gain power3, int steps3)*: Sets the gain the transmit link in the transceiver.
  - *void Select_LNA_GS (Gain_LNA integ)*: Selects LNA gain.
  - *void Select_Rx_VGC_TranscPA (Action_Gain power2, int steps4)*: Sets the gain of the receive link in the transceiver
  - void Select_Rx_VGC_TranscCouplage (Action_Gain power4, int steps2): Sets the gain of the receive amplifier of the coupling link.

DIGITAL BASEBAND PROCESSING

The digital end of this hardware platform is based on a Test and Evaluation Board (TEB6416) from Texas Instruments that embeds a TMS320C6416 running at 500MHz allowing a crest computing power of 4500 MIPS and a daughter card that is mapped in the CE2 space of the EMIFA bus.

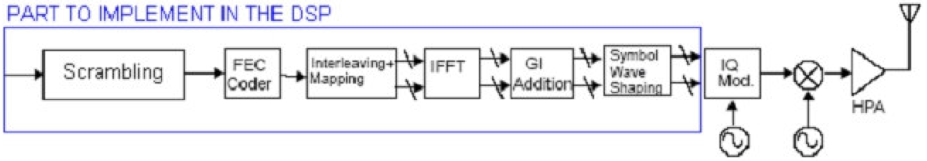


Fig4. DSP blocks for OFDM modulation

As main features, the daugther card offers two 12 bits ADC running at 80Ms/s and two 12 bits DAC able to run at 100 Ms/s and a Xilinx FPGA which realize full speed interface between DSP and AD/DA converters. This FPGA could also embeds high-speed signal processing on the data stream coming from or going to the converters. The driving of the AFE is also done through this FPGA which contains regiters driving logical pins. Interface between DSP and AD/DA converter is done thanks to EDMA. For the DA path, the DSP send a block of 256 data thanks to EDMA, Those data are written in daughterboard fifo and then they are transferred to DA converter at sampling frequency which is driven by the FPGA. Data coming from the AD converters are send to daughter board fifo, when the count of sample is reached an interrupt ts sent to the DSP. This count is set-up by user in FPGA registers.

OFDM modulation is implemented in DSP according to 802.11g standard [3]. It includes the building blocks shown on figure 4. Because we are mainly interested at first in the spectral behaviour of the modulated signal the steps taken to perform the modulation were the construction of the frame format, of the OFDM symbols, the modulation of the subcarriers (16-QAM or BPSK depending on the symbol considered), the IFFT, and the guard interval addition cyclical extension. The overall system is build around real-time tools of DSP-BIOS. The IFFT has been implemented thanks to the radix 4 64-points FFT from DSPLIB. The IFFT can be realized thanks to a FFT according to the following algorithm:

- Swap real and imaginary part of the input samples.
- Compute the FFT.
- Swap real and imaginary part of the output.

In the OFDM modulation, swapping of the real and imaginary part at the input can be done without overhead. If mapping is realized thanks to a look-up table. For the input, all we have to do is to switch quadrature and in phase components at the output of the mapper. At the output of the FFT we only have to read the memory according to the desired pattern. So the number of cycles to compute the IFFT is the same as for the FFT, that is 496 cycles against 421 cycles which is the theoritical value thanks to the following expression (N=64).

$$\text{ceil}\left[\log_4(N)-1\right]*(5*N/4+25)+5*N/4+26$$

496 cycles is equal to 1µs with a 500MHz clock, this is, roughly speaking, one quater of the available time. The reference timing for OFDM is given hereafter:

- T_{FFT}, IFFT period 3.2µs
- T_{GI}, Guard interval 0.8µs
- T_{SYMBOL}, T_{FFT} +T_{GI} 4µs
- T_{SHORT}, Short training sequence 8µs
- T_{LONG}, Long Training sequence 8µs

For the different kind of symbol in a frame we get the following processing loads : PLCP preamble (12 symbols) : 13352 cycles, Signal Field symbol: 7568 cycles, PSDU symbol: 10288 cycles. Those results does not allow real time processing for this implementation and need further optimization. The chosen test synbols sequence is given figure 5.

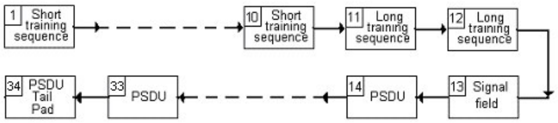

1 Short training sequence
10 Short training sequence
11 Long training sequence
12 Long training sequence
34 PSDU Tail Pad
33 PSDU
14 PSDU
13 Signal field

Fig. 5 The symbols sequence used

After having stated on the processing, we will focus on the organisation of data manipulation and transfer on transmit link. Both the whole system's architecture and the modulation implementation had to be adapted to the DSP and daughterboard's architecture in order to send the data. Figure 6 shows how the adaptation was performed:

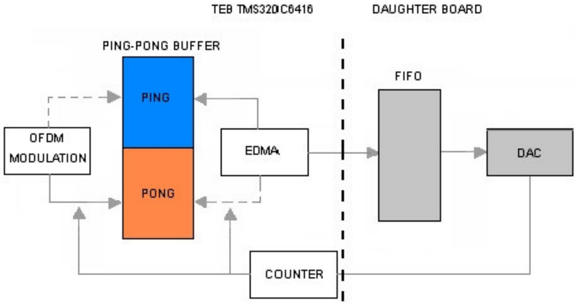


Fig.6 DSP blocks for OFDM modulation

The resources used are depict on figure 6 as the figure 7 describes the time sequence of operations.

The software that performs the modulation writes the resulting samples in buffer "ping", while an EDMA transfer from the buffer "pong" to the daughterboard's FIFO, whose digital samples are converted into analogue by means of a DAC converter (Figure 6).

Figure 7 shows that the master time-base is implicitly the DA sample rate: each time a sample is transferred from the daughterboard FIFO to the DA converter, a pulse is sent to the timer1 of the DSP which is decremented, when it reaches zero, an interrupt is generated. The EDMA is configured with two chained configuration tables, one to transfer data from buffer ping to the fifo and the other to transfer data from buffer pong to the fifo.

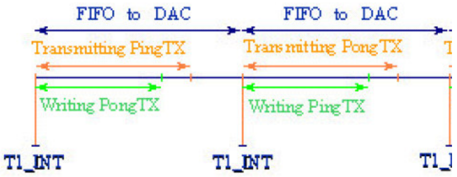
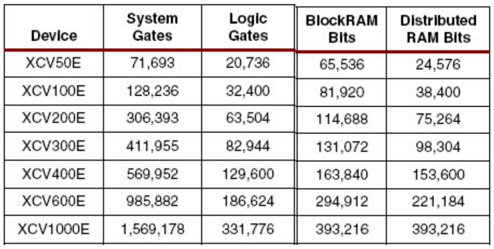


Fig.7 Timing for data transfer

Timer1 interrupt flags an End Of Count event which is used in two ways, first an interrupt service modifies the pointer to the input buffer to be filled by the application which output data to DA converter (here the modulation application seen above); second it triggers directly a new DMA transfer to the FIFO, freeing the DSP of this task. On figure 7, the arrow starting from each "T1_INT" and labelled "Transmitting P(i/o)ng TX" indicates the EDMA duration, while the arrow labelled "Writing P(o/i)ng TX" indicates the modulation processing duration. The time between the end of this last arrow and the next timer1 interrupt is the margin of the application regarding real-time processing.

The daughter board also embeds an FPGA (figure 8), XCV50E is the default., but the board can be upgraded with a XCV1000E allowing more logical and RAM resources. This board has then provision for high speed signal processing at the sample rate (80 Ms/s) which corresponds to a sample period of 13 ns. This FPGA family has the following computation performance, it is able to compute a 16x16bits addition in 4.3ns and a 16x16 multiplication in 5.1 ns. Some processing can then be realized before (AD link) or after (DA link) the sample are handled by the DSP. Such processing could be predistortion [1,2]. The need for predistortion processing comes from the non-linear behaviour of the power amplifier (figure 2). The distortions involved by this non linear behaviour create spectral regrowth in the adjacent channel (figure 3) and deformation of the constellation.



Device	System Gates	Logic Gates	BlockRAM Bits	Distributed RAM Bits
XCV50E	71,693	20,736	65,536	24,576
XCV100E	128,236	32,400	81,920	38,400
XCV200E	306,393	63,504	114,688	75,264
XCV300E	411,955	82,944	131,072	98,304
XCV400E	569,952	129,600	163,840	153,600
XCV600E	985,882	186,624	294,912	221,184
XCV1000E	1,569,178	331,776	393,216	393,216

Fig.8 Xilinx FPGA that can be embedded on the daughter board

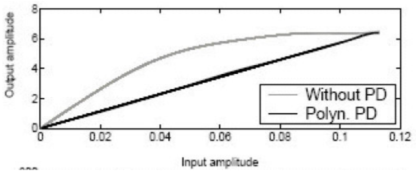


Fig.9 Output vs Input amplitude at the power amplifier with and without pre-distortion processing.

Figure 9 illustrates the distortion on the amplitude. Actually it depends on the Peak to Average Power Ratio (PAPR, around 18dB for OFDM) and on the histogram of the magnitude at the input of the PA.

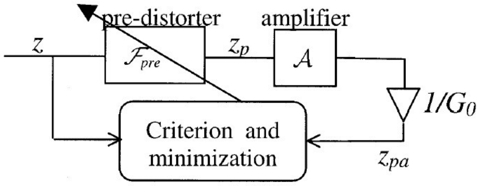


Fig.10 Common architecture for pre-distorsion

If the PA is considered as a memoryless system, the predistorter can be implemented by a look-up table. It generally acts as a complex gain depending on the input amplitude.

CONCLUSION

The hardware platform obtained result in a generic architecture that can be used to implement any wide-band modulation. The same platform could be used for research on software design radio and design of new architectures for emitter. It also well-suited for educational purpose in the field of radio digital communications.

REFERENCES

[1] R. Marsalek, P. Jardin, G. Baudoin, "From post-distorsion to pre-distorsion for power amplifiers linearization", IEEE - Communications Letters, N° 7, Vol. 7, pp. 308-310, July 2003.

[2] G. Baudoin, R. Marsalek, P. Jardin, "A new approach for LUT-based digital predistorters adaptation", EDS'2003, Proceedings of Electronic Devices and Systems Conference, 9 - 10 September 2003 Brno University of Technology, Czech Republic.

[3] 802.11g Part11 amendment 4 - IEEE computer society (SS95134) – 27 june 2003

[4] P. Loumeau, J.F. Naviner, H. Petit, L. Naviner, P. Desgreys "Analog to digital conversion: technical aspects", Annales des Télécommun. Vol.57 n°5-6,pp.338-385, 2002