

Low Power Handheld Module for Parametrical Measurement with CPLD Structures

B. R. Donchev, M. H. Hristov

Faculty of Electronic Engineering and Technology
Technical University of Sofia

Kliment Ohridski Str. 8, 1797, Sofia, Bulgaria

Tel +35929653115, E-mail: donchev@ecad.vmei.acad.bg

Abstract

This paper introduces developing of handheld module for parametrical measurement, based on low power programmable logic arrays (XPLA3) from Xilinx. The module has five operating modes: frequency measurement, period measurement, measurement of the phase difference between input signals, measurement of time intervals, calibration mode. On four-digit dynamically managed display is visualized the current measuring result and a light-emitting-diode display controls device operations carried out with six buttons. External crystal oscillator of 20 MHz operating frequency is used as a reference source. The module has interface (RS232) to connect with PC platform or μ -controller with standard transmitting frequency – 9600 Kb/s. In the project, top-down approach for CPLD designing is demonstrated where VHDL (VHSIC Hardware Description Language) is used, which reduces the time for designing.

1. INTRODUCTION

Most commonly digital Integrated Circuits (IC) can be subdivided according to their purpose into standard and specialized.

Standard circuits are the ones, which are produced into large-scale series and have universal purpose.

Programmable structures are an important part of specialized IC. Basically they can be divided into two groups:

- Complex programmable logic devices (CPLD) – A non-volatile programmable device (PLD) based on programmable AND-OR structure.

- Field Programmable Gate Array (FPGA) – Most (Xilinx, Altera) of FPGA are SRAM based that use Look-Up Tables (LUT) to generate logic. A LUT can implement any logic function of its inputs. Also referred to as a function generator.

Benefits of programmable circuits are:

- Decreasing time-to-market for the whole system;
- More large functional capabilities of the system at lower cost;
- Decreasing of necessary area for the PCB;
- Zero fault risk during developing phase.

Programming of these chips up to date is very simplified. It gives to the end user an opportunity on one's own to shape functionality of the product. Thus less time for designing, including multiple programming, is achieved.

No elaborate and time-consuming tests before manufacturing are required.

There are two patterns to bring in the input information:

1. Graphical – helps a designer to build own schematic with predefined elements stored into libraries. In practice, this way is called bottom up designing.

2. Based on behavioral description of the developing circuit model using high-level description language (HDL) – top-down designing approach.

2. DESIGN DESCRIPTION

The goal of this paper is developing of low power handheld module for parametrical measurement with CPLD structure. The block scheme is shown in Figure 1.

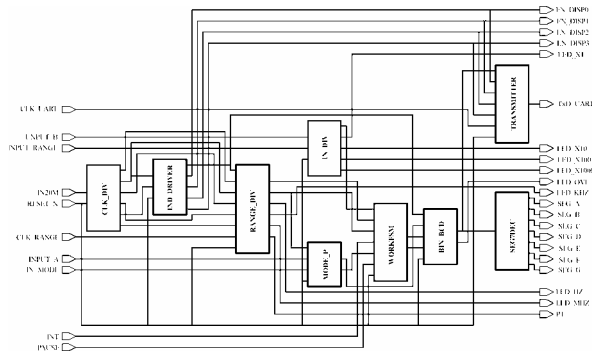


Fig. 1 Top level of the low power handheld module

Table 1 gives input/output interface of the design

Signal name	Direction	Description
IN20M	IN	Input for global reference clock signal
INPUT_A, INPUT_B	IN	Input data signals
RESET_N	IN	Global reset signal
INT	IN	Interrupt signal
IN_MODE	IN	Signal for determination of mode
INPUT_RANGE	IN	Signal for determination of input signal range
CLK_RANGE	IN	Signal for determination of result range
MEMORY	IN	Memory function
LED_OVL	OUT	Signal for overload
LED_Hz, LED_kHz, LED_MHz	OUT	Indications for result range
LED_x1, LED_x10, LED_x100, LED_x1000,	OUT	Indications for input signal range
PT	OUT	Activates the point in LED display
EN_DISP0, EN_DISP1, EN_DISP2, EN_DISP3,	OUT	Signals for dynamic driving display
SEG_A, SEG_B, SEG_C, SEG_D, SEG_E, SEG_F,	OUT	Signals for segment driving. Logic zero ('0') is active level.

SEG_G,		
TxD	OUT	One way data channel

Tab. 1 Input/Output interface

Table 2 gives sub modules of the design

Module name	Description
BCD7SEG	BCD to 7 segment code converter
IN_DIV	Input signal divider
IND_DRIVER	Indication driver
REF_REG	Reference divider for internal clock signals
WORK_REG	Two level working registries
RANGE	Logic for result range driving
WORK_FSM	Main Finite State Machine
MODE	Logic for set/preset of working mode
TxDATA	UART sub module

Tab. 2 Sub modules of the design

Table 3 gives the most significant internal signals of the design

Signal name	Description
MESURE_C	Enables first level of working registries. Logic '1' is active level.
MESURE_CE	Sets first level of working registries of zero value. Logic '1' is active level.

Tab. 3 Internal signals of the design

Mainly the process of measurement of the design is divided into two stages: *stage of impulses counting* and *stage of the result visualization*. In general, for all operating modes the following is true (1):

$$T_{disp} = 2 * T_{measure} \quad (1)$$

where, $T_{measure}$ - time for measurement, T_{disp} - time for visualization.

Visualization is accomplished upon four-digit LED display, dynamically controlled from sub module *ind_driver*. To avoid the so-called "floating" on the display the control logic provides moments between each turning on of the separate digits. In those moments, the indication is off. The "flicker" feeling on the display is achieved by refresh frequency over 100 Hz/digit.

The main frequency according to which the whole design is synchronized is taken from external crystal oscillator (20 MHz). Operating mode, measurement scope and divide factor for an operator through buttons sets up input signals. The design provides “Memory” function. If “Memory” function is activated, the display presents the last measured value.

In cases of overload, there is an indicating signal for that status (LED indication).

The design gives a possibility for PC connection through asynchronous serial interface. Its main function is data converting from parallel to serial form. This enables the implement of statistic surveys. The format of transmitting message consists of: start bit, eight data bits, stop bit and parity check bit (Figure 2). Start and stop bits are used for synchronization between nodes, during the data transmission.

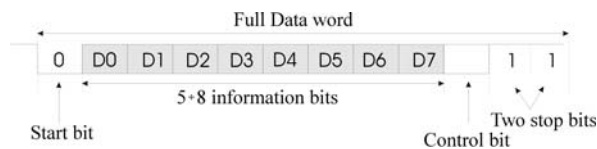


Fig. 2 Transmitter data format

Line driver to RS232 interface compatible bus connects the module. Its interface allows data transmission on condition that every node has own clock generator (9 600 KB/s).

Figure 3 gives the scheme of connections between nodes.

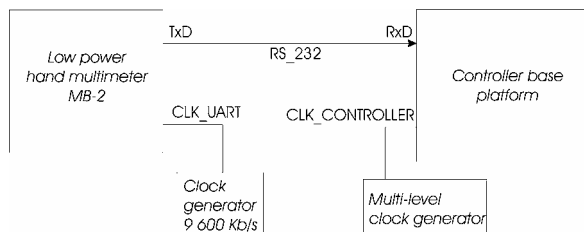


Fig 3 Example for connecting with other device

The main general functional properties of the module for serial communication are:

- Double buffering of interchanged data;
 - Auto adding to of the bit for parity check;
- The eight information bits are packed with control bits:
- Start bit, represented with '0' logical level;
 - One or two stop bits, represented with '1' logical level;
 - A bit for parity check.

The interval between symbols contains logic level '1' called marker. In such a way, each byte is transmitted independently from the others.

The data, which are sent through interface, are read from the data bus connected with the indication, sub-

ordinated to these signals: EN_DISP0, EN_DISP1, EN_DISP2, EN_DISP3.

The module is designed through high-level hardware description language. The design is subdivided into modules, which synchronization is achieved with one main finite state machine (FSM), shown in Figure 4. By means of describing the operating states of the design and conditions for transitions between them are determined by internal signals: MESURE_C and MESURE_CE. The top-down approach is used to decrease the time for designing. A programmable logic (XPLA - CoolRunner) from Xilinx is used for the implementation of the design.

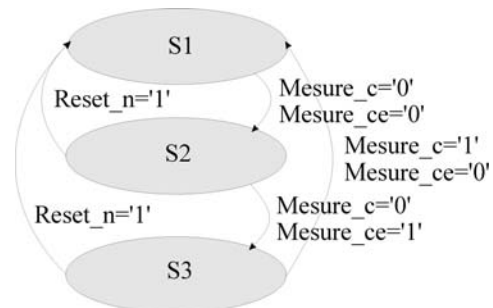


Fig. 4 Main Finite State Machine

Simulation results presenting design work are shown in Figure 5 and Figure 6.

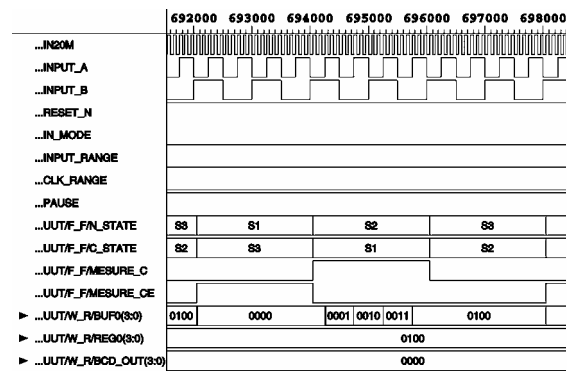


Fig. 5 Simulation results of the measurement process

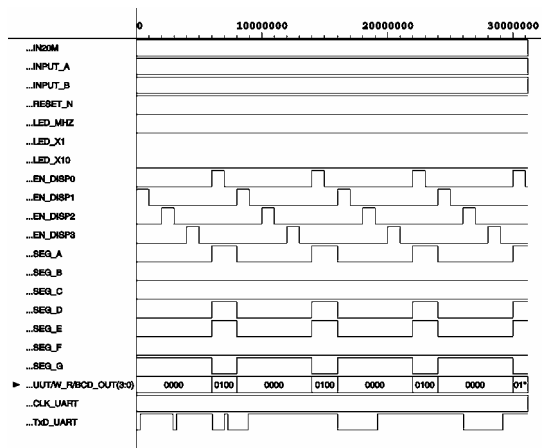


Fig. 6 Simulation results of the data output operation

MB-2 operates in five modes:

- Frequency meter – measures the frequency of signal on *input_a* and through control logic the signal is fed into input of block *work_reg*. There, for duration of time determined from *ref_reg*, the impulses entering on its input are counted. After the block *work_reg* is disabled, the result is sent to the code-converting block - *bcd7seg*.

Figure 7 gives the scheme of this mode of operation.

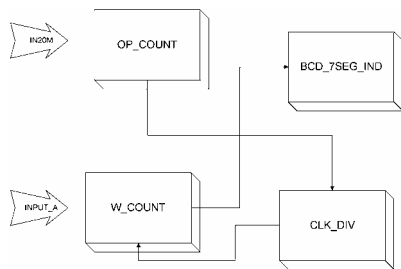


Fig. 7 Frequency mode

- Period meter – in that mode the signal, whose period will be measured, is fed into input divider *in_div* (it can make divisions by 1, 10, 100, 1000). After that, the output signal from *in_div* is used to enable the module *work_reg*. Impulses with frequency determined from block *ref_reg* are fed at *work_reg* input. This frequency is called frequency of digitalization. It defines the tolerance of period measurement. The final result is determined by formula (2):

$$T_{measure} = N_{scope} / N_{disp} \quad (2)$$

Where, $T_{measure}$ – measured value for the period; N_{scope} – defined range of input divider; N_{disp} – counted impulses

Figure 8 gives the scheme for period mode of operation

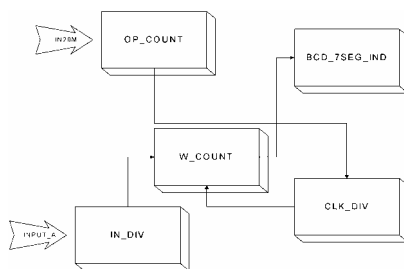


Fig. 8 Period mode

- Phase difference meter - in this mode of operation the phase difference between the signals fed into inputs *input_a* and *input_b* is measured. The signal through *input_a* is directly connected with the input of *work_reg*. The other signal (fed to *input_b*) that passes through the input divider activates *work_reg*. To achieve precise results in this mode it is necessary for the signal fed into the input of block *work_reg* to have higher frequency.

The final result is determined by formula (3):

$$N = N_{disp} * IN_MUX_DIV \quad (3)$$

Where, N – result from measurement; N_{disp} – counted impulses; IN_MUX_DIV – defined range of input divider.

Figure 9 gives the scheme for phase differences mode of operation

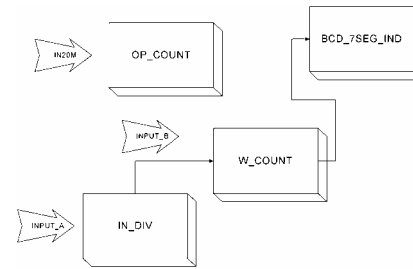


Fig. 9 Phase difference mode

- Calibrator – the mode by which the operator can check and respectively calibrate the frequency from external generator. Principle block scheme is shown in Figure 7.

- Chronometer – by this function we can measure time intervals. In that case at the input of module *work_reg* impulses are fed with frequency determined by sub module *ref_reg*. The sub module *work_reg* is active for the time determined by an operator through button *int*.

Figure 10 gives the scheme for Chronometer mode of operation.

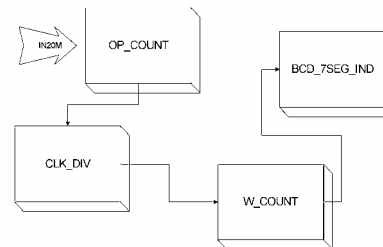


Fig. 10 Chronometer mode

3. SYNTHESIS RESULTS

Some synthesis results from main Finite State Machine, Bin to BCD converter, Clock divider and Tranmitter module are shown below:

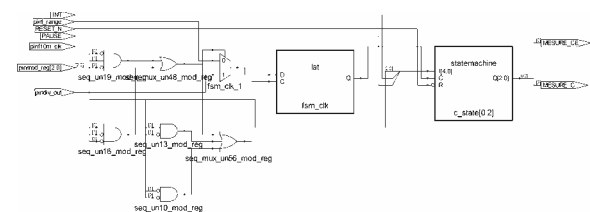


Fig.11 Synthesis results of Finite State Machine

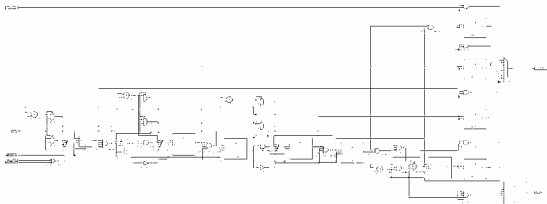


Fig.12 Synthesis results of Bin to BCD converter

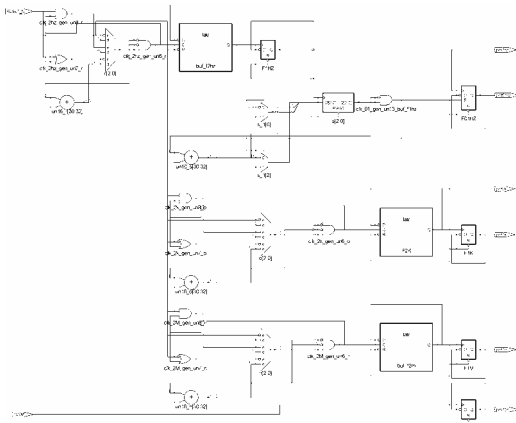


Fig.13 Synthesis results of Internal Clock divider

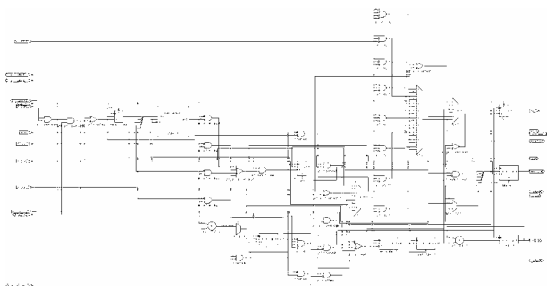


Fig.14 Synthesis results of Transmitter module

REFERENCES

- [1] B. Donchev, Dr. M. Hristov, V. Manoev, " Programmable Structures Application in the Electronic Design", Electronics '00, Sozopol – Bulgaria, 2000, pp 81-86.
- [2] AppLinX, Xilinx Data Base CD ROM, 2001, www.xilinx.com .
- [3] S. Mazor, P. Langstraat, A guide to VHDL, 1993,
- [4] T. Vassileva, Automation of specialized ICs designing, 1997, Bulgaria.
- [5] Z. Navabi, VHDL Analysis and Modeling of Digital Systems, 1993, McGraw-Hill Book Co, USA

3. CONCLUSION

This project demonstrates top-down approach for CPLD designing, using VHDL and post following simulations.

In an ideal design environment, the high level description of the system is understandable to the managers and to the designers, and it uniquely and unambiguously defines the hardware. This high level description can serve as the documentation for the part as well as an entry point into the design process.

Designing with programmable arrays decreases to minimum subjective faults.

In conclusion, two aspects mentioned above lead to short time-to-market of developed device.

This project is made by ECAD Lab (<http://ecad.vmei.acad.bg>) in the Technical University - Sofia, Bulgaria.

