

A High-Resolution (< 10 ps RMS) Multi-Channel Time-to-Digital Converter (TDC) Implemented in a Field Programmable Gate Array (FPGA)

Eugen Bayer and Michael Traxler

Abstract—A high-resolution 48-Channel Time-to-Digital Converter (TDC) implemented in a general purpose Field Programmable Gate Array (FPGA) is presented. Dedicated carry chains of the FPGA are utilized for time interpolation purposes inside a clock cycle. A counter running at the system clock frequency provides a global time stamp. These two values, along with the channel number, are stored for readout. An extra effort was made to improve the resolution beyond the intrinsic cell delay of the carry chain as well as to achieve the same resolution on all 48 channels. Due to large bin width variations a bin-by-bin calibration scheme was used. Time interval (TI) measurements between two channels were made to determine the RMS and the time resolution of a single channel. At least 6 ps single channel resolution was achieved for all channels. Additional measurements were performed to characterize the influence of the temperature and voltage variations on the RMS value and the mean as well as the sensitivity of the TDC to crosstalk. The results of these measurements are also presented in this paper.

Index Terms—FPGA, Time-to-Digital Converter, TDC, TDL, Virtex-4, picosecond resolution.

I. INTRODUCTION

TIME-TO-DIGITAL Converters are widely used in many scientific applications. High-resolution ASIC-TDCs and commercial modules are utilized in Time of Flight (TOF) detectors in virtually all nuclear physics experiments. The efficiency of the particle identification depends directly on the precision of the time measurements. For example, in the HADES (Hi-Acceptance-Di-Electron-Spectrometer) [1] a TDC resolution better than 30 ps is needed to discriminate electrons from protons. Two interpolation methods used in high-resolution applications – the Vernier and the tapped delay line (TDL) method – have been also successfully implemented in FPGAs. The advantage of a FPGA implementation is the less complex, less expensive and less time consuming design process as well as the flexibility and adaptability of the FPGA-TDC to special needs of the current application. Already in 1997 Kalisz, et al. achieved a resolution of 200 ps on a QuickLogic FPGA [2]. In 2002 Andaloussi, et al. implemented a TDC in a Virtex-4 FPGA with the 150 ps resolution [3]. Xie,

et al. extended in 2005 the previous approaches and achieved a resolution of 75 ps [4]. Wu, et al. implemented the TDL method in an Altera Cyclone II FPGA and proposed two methods to improve the resolution of the TDC beyond the intrinsic cell delay of the chain. The resolution was improved from 40 ps to 25 ps RMS with the first method and down to 10 ps RMS with the second method but on cost of a larger resource effort and an increased dead time [5]. A TDC with 40 ps resolution based on the Vernier method was implemented by Junnarkar et al. in 2008 [6]. In 2009 a pure TDL version was implemented by Favi, et al. on a Virtex-5 FPGA that is fabricated in the 65 nm process. On this chip a resolution of 17 ps (standard deviation) was achieved in some placement configurations [7]. We implemented the TDL method on a Virtex-4 FPGA that is fabricated in the 90 nm process and achieved 17 ps RMS in TI measurements. The design was then extended with a modified version of the first method (“Wave Union A” method) described by Wu [5]. The use of Place and Route (PAR) constraints was necessary to minimize the unpredictability of the routing algorithm that itself results in unpredictable signal delays. No manual routing was done. With regard to the need of many channels on a single FPGA and a small dead time a resource-efficient and fast thermobinary encoder was also developed.

In the following section the design of one TDC channel will be explained. Section III describes the calibration method that was used to derive the time interval in picoseconds. In the section IV the measurement results are presented. A conclusion and an outlook will be given in sections V and VI.

II. DESIGN

In common TDC-Designs two different methods are used for the coarse and the fine time measurement. Usually a counter running at the system clock frequency provides nanosecond resolution and the finer time resolution is achieved with a time interpolation technique. This approach offers a large dynamic range that is only limited by the number of the counter bits. A good overview on possible methods for the fine resolution can be found in [8]. Though, the time interpolation techniques used for implementation of multi-channel TDCs on FPGAs usually are based on the Vernier or the TDL method. The Vernier method uses two ring oscillators that run with a slightly different frequency. This difference equals the achievable incremental resolution of this method and has to be adjusted very carefully. However, due to the inhomogeneity

Manuscript received June 15, 2010. Work supported by EU FP6 grant, contract number 515876, EU FP6 grant RII-CT-2004-506078 and by the Hessian LOEWE initiative through the Helmholtz International Center for FAIR (HIC for FAIR).

E. Bayer is with the Department of Nuclear Physics, Johann Wolfgang Goethe - University, Frankfurt am Main, Germany, e-mail: eugen.bayer@gmx.de.

M. Traxler is with GSI Helmholtz Centre for Heavy Ion Research, Darmstadt, Germany.

of the signal run times in the FPGA fabric as well as the unpredictable routing inside the FPGA an additional effort for adjusting the needed difference on all channels is inevitable. Even more, the same effort could become necessary again, if the design is changed only slightly. Although, the TDL method also suffers from the inhomogeneous delays in terms of high non linearity, it is possible to overcome this problem with the proposed techniques.

A. General Considerations

The TDL method can be implemented in different schemes. The asynchronous input signal, the clock of the latching FlipFlops or both can be delayed. The latter structure is also called the Vernier delay line. However, in FPGAs it is not possible to delay the clock signal for each FlipFlop inside a logic block, but there are many dedicated carry lines inside the FPGA which connect adjacent slices to columns and can be used as delay elements for the asynchronous input signal. Fig. 2(b) shows the carry resources inside a Xilinx Complex Logic Bloc (CLB). When the carry logic is instantiated the cin input is delayed by the carry multiplexers and the register array is driven by a single clock. The cin to cout delay is, according to the data sheet, 90 ps (max.) and the calculative delay of one element is 45 ps (max.). As mentioned before the real delays show a significant variation and can be measured by performing a code density test. In our Test-FPGA the delays vary between 100 ps and 3 ps as is shown in (Fig. 1). The variation is caused by two factors: The smaller variations follow a periodic pattern that is caused by the CLB structure (Fig. 2(a)), whereas the two so called "ultra wide bins" (UWB, named according to [5]) mark the boundaries of the clock distribution tree. A descriptive explanation for this phenomenon is given in [9]. These bin width variations would cause high differential non-linearity (DNL) and integral non-linearity (INL).

A possible approach to flatten the DNL of a TDC channel is using several delay lines, which are fed with a delayed version of the input signal. Thus, if one line is in an UWB and is not sensitive to the propagation of the signal an other line will maintain the sensitivity. To get rid of the drawback of the larger resource usage a better technique was proposed in [5]. They store a pattern of 0-to-1 and 1-to-0 transitions in the carry chain which is released as soon as the rising edge of the asynchronous hit signal arrives. Analogue to the "pure" version, the propagation status of the pattern is stored at the rising edge of the system clock, performing multiple measurements with a single delay chain structure. According to the fact that the output is not a simple thermometer code any more, a more complex encoder structure is needed to convert the propagation status to a binary value.

B. Architecture

The pipelined design of one TDC channel, shown in Fig. 5, can be divided into four parts: the interpolator, the trigger, the sampling unit and the ancillary electronics. The interpolator is composed of the carry chain and the pattern launcher. The carry chain can be instantiated either by implementing

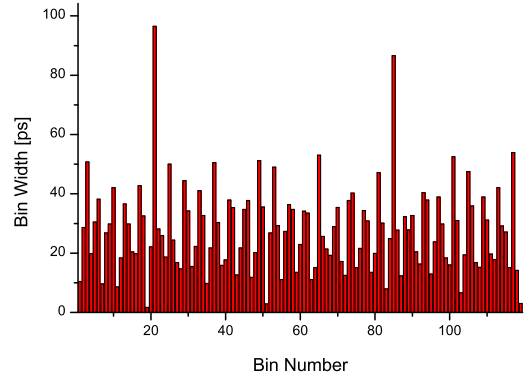


Fig. 1. Bin width plot of a region in the carry chain with two UWBs at positions 21 and 85.

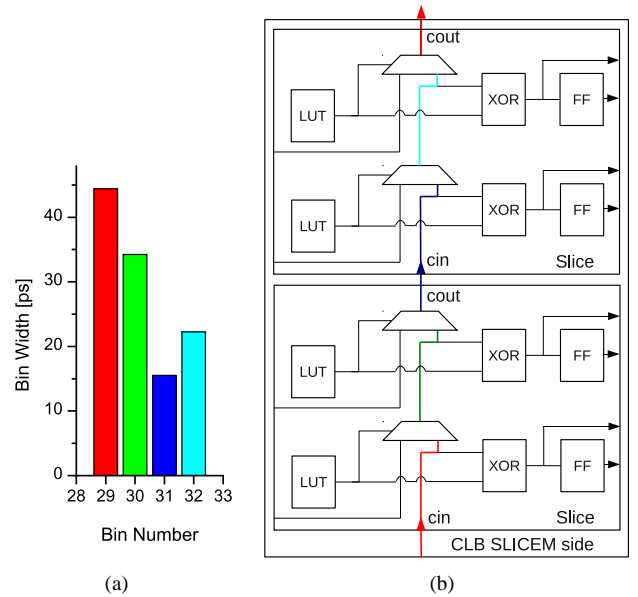


Fig. 2. The bin width pattern (a) and its origin in the CLB structure (b).

adder structures like it is specified in [9] or the carry chain components can be instantiated directly. In the pattern launcher a pattern of 0-to-1 and 1-to-0 transitions can be stored. The corresponding adder schema can be found in [10]. In our design the 1-to-0 transition is also used for interpolation and a minimalist Wave Union consists of one 0-to-1 and one 1-to-0 transition, forming a single "wave". All transitions are released by the trigger simultaneously if it detects a rising edge of the input signal. At each rising edge of the system clock the trigger signal and the propagation status of a possible hit event are sampled and the position of the 0-to-1 transition is encoded. If a hit occurs, the FlipFlop array clock is disabled for two clock cycles to enable the encoding of the 1-to-0 transition. During this time the channel is not sensitive to an other hit event. In the following clock cycles both positions are summed and stored along with a global time stamp in a channel-FIFO. This information can then be read out by the top-level design. On the basis of this information the time interval between different channels can be determined. The conceptual timing diagram is shown Fig. 4. The time interval (T_{DIFF}) can be explained

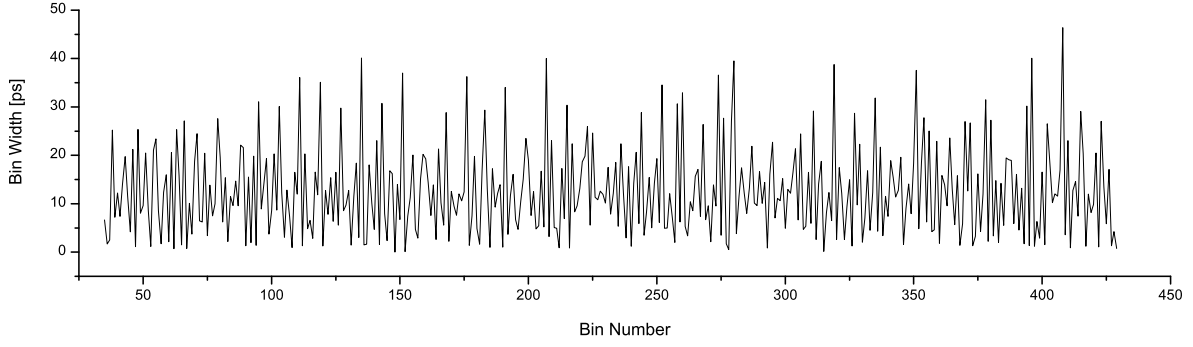


Fig. 3. The bin width plot of the 2-Transitions (minimalist) Wave Union.

as follows:

$$T_{DIFF} = T_A - T_B + |S_A - S_B|T_{SYS} \quad (1)$$

for $S_A < S_B$

where T_{SYS} = System clock period
 S_A, S_B = Timestamp (a number)

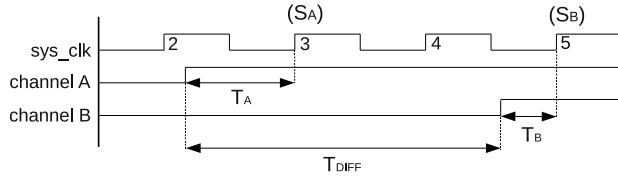


Fig. 4. The principle timing diagram of the measurement method.

This scheme can be easily modified for pulse width measurements. In this case two channels are utilized for one measurement, in which the channels are sensitive to the rising and the falling edge respectively.

The whole delay of the chain has to be larger than the period of the coarse counter clock. The variation of the signal run-time in the FPGA due to voltage and temperature variations has to be taken into account. At a lower environment temperature more delay elements could become necessary. In our Test-FPGA about 198 delay elements are passed at 30° C in 5 ns and the delay chain consists out of 256 delay elements to support the lower temperature region. Due to the inhomogeneity of the delays the transitions proceed to the next delay element at different points in time, thus the number of bins using two transitions is twice as large and the UWB are effectively subdivided. Their width never exceeds 50 ps (Fig. 3).

C. Asynchronous Design

Due to the asynchronous nature of the design special care has to be taken on the signal run times in the asynchronous part of the design. For instance the number of delay elements between the transitions, hence their distance on the FPGA die, has to be chosen carefully. The resulting delay should not be smaller than the largest UWB. Otherwise the wave or a part of a Wave Union may disappear if crossing a UWB

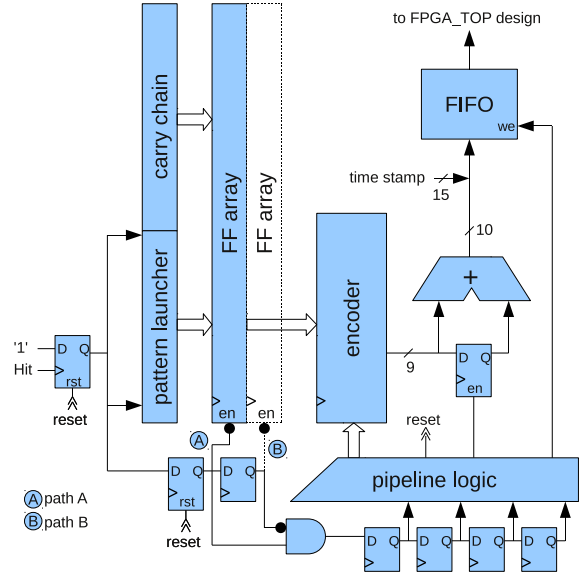


Fig. 5. Block diagram of a TDC channel.

while the delay chain is sampled. This delay is composed of two elements: the comprised delays of the chain between the transitions and the run time difference of the transition releasing signals. In our design six elements were sufficient to form a single wave if the unpredictability of the run times is minimized by placement constraints.

An other issue originating from the asynchronous design is the metastability of the sampling FlipFlops. If the rising edge of the trigger signal occurs in the metastability window of the sampling FlipFlop, it can enter a metastable state and resolve either to a one or a zero after an arbitrary time. If the metastable FlipFlop drives an other FlipFlop, it can also enter a metastable state if the metastability is not resolved until the next metastability window. However, the probability of a metastable event lasting longer than a certain time is decreasing exponentially and is considered to be very low for a 5 ns system clock period. A common method to make this probability negligible small is the use of a dual synchronizer. In our design both possibilities (two and a single synchronizing FlipFlop) were tested (path A and path B in Fig. 5). If a metastable signal state would arrive on the enable input of the FF-array, causing some FlipFlops in the array enter a metastable state and resolve to different logic levels,

the encoder output would be a random number. This would cause measurement values that exceed the usual deviations from the mean in a fixed time delay measurement. Since no measurement errors were found in measurements containing over 30 million hits, the first design (path A) was chosen for the final implementation. Metastability is also the origin of the so called "bubbles" problem. During the sampling of the propagation status there may exist some FlipFlops close to the 0/1 and 1/0 transitions in which the condition for a metastable state is fulfilled. These FlipFlops can again resolve to a one as well as to a zero and produce single inverted bits in the encoder input, e.g.: 111110100000 instead of 111111100000. Since this is also a common problem in the flash Analog-to-Digital Converter (ADC) designs, we used the methods from the relevant literature of this area [11]. In our folded thermo-to-binary encoder design we use the nand-gate based bubble suppression, to tackle this problem.

III. CALIBRATION

The initial point of the calibration is a code density test. If the hit distribution in the histogram, their total number and the period of the counter clock is known, the real bin width in picoseconds can be calculated. Assuming the total number is 50k, then the width of a bin W_i with N_i hits is $W_i = N_i \cdot 5000 \text{ ps} / 50000 = N_i \cdot 0.1 \text{ ps}$ for a 200 MHz system clock. With this information the lookup table (LUT) for bin number to picosecond conversion can be build. The corresponding time of a bin is then derived as follows:

$$T_i = \sum_{k=1}^{i-1} W_k + W_i/2 \quad (2)$$

In our measurements about 500k hits per channel were booked and used as input for the calibration software. Due to the influence of the environment conditions on the measurement setup, that is examined in the next chapter, always the newest data should be used for the code density test.

IV. MEASUREMENTS

In our measurements we determined the time interval between two rising edges which were fed into two different TDC channels. The pulses for the TDC inputs were generated outside the FPGA and were uncorrelated to the system clock. The time delay between the edges was varied by using different cable delays or with the Tektronix Data Timing Generator DTG5078. Several measurement series were performed to determine the achieved RMS value and the impact on it due to the temperature and supply voltage variations.

A. RMS

Several measurement series with increasing time interval were performed with both edges within a counter clock period, within a few periods and with a time interval of about 1 μs respectively. The results are shown in Table I - III. In the first two measurement series the RMS value was 9 ps. So the resolution of a single channel is $9/\sqrt{2} \approx 6 \text{ ps}$. Due to the limited accuracy of the system clock (SILABS-Si530)

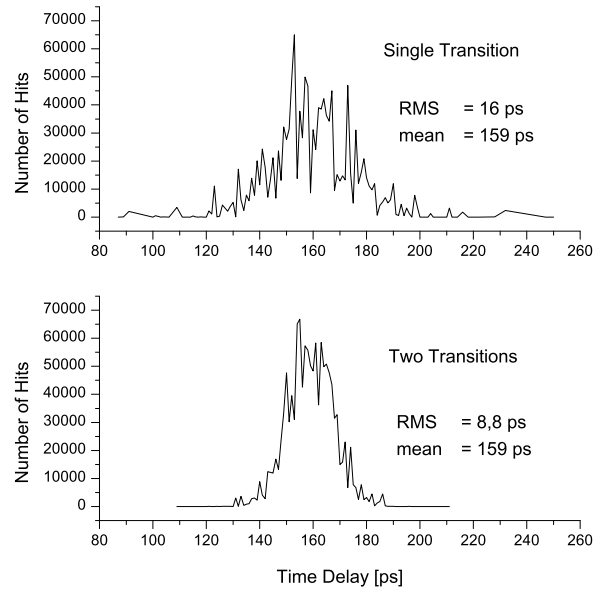


Fig. 6. The histograms of the measurements utilizing a single transition and two transitions.

TABLE I
MEASUREMENTS IN 1 CLOCK CYCLE RANGE

cable [cm]	mean [ps]	RMS [ps]
6	1697	9
7	1740	9
8	1781	9

TABLE II
MEASUREMENTS IN >1 CLOCK CYCLE RANGE

DTG delay [ns]	mean [ps]	RMS [ps]
42	45798	9
44	47798	9
46	49798	9

TABLE III
MEASUREMENTS IN μs RANGE

DTG delay [μs]	mean [ps]	RMS [ps]
1,004	1005795	11
1,006	1007797	11
1,008	1009798	11
1,010	1011800	11

additional 2 ps in the RMS value were measured in the long period measurement. This resolution was measured on all 32 channels implemented on the Virtex VC4VLX40 FPGA. The usage of two transitions improved the resolution by the factor 1.8. The histograms of two successive measurements utilizing a single and two transitions respectively is shown in Fig. 6 along with the measured mean and RMS value.

B. Temperature and Voltage Variations

The performance of the TDC was also examined under varying environment conditions. In the temperature test setup the temperature of the FPGA was increased from 30° C to

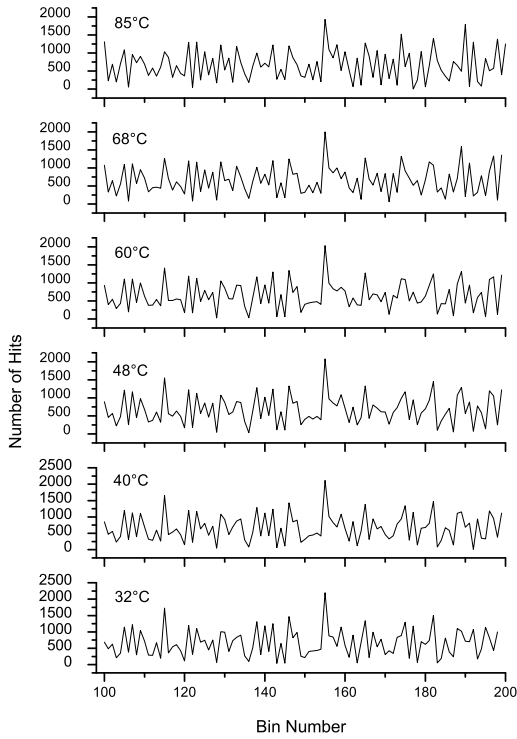


Fig. 7. The deformation of a region in a calibration look-up-table.

85° C in 5° C steps for a fixed time interval. If only data from the actual temperature is used for calibration, no significant effect on the RMS and the mean value could be observed. Though some characteristics of the raw data changed. A deformation of the calibration look-up-table (LUT) with the increasing temperature is shown in Fig. 7. This results in a worsening of the RMS value, if a calibration LUT beyond the 5° C temperature region is used (Fig. 8). However, the change of the temperature has no influence if the calibration table is updated often enough to follow the temperature rise.

In the voltage test setup the supply voltage was varied within the allowed voltage range of the FPGA core from 1.15 V to 1.25 V. We observed a shift of 40 ps in the measured mean and the RMS value was decreased about 2 ps in the whole voltage region (Fig. 9).

C. Crosstalk

The TDC's sensitivity to crosstalk was examined. For this purpose the time delay between channel 1 and 3 and between channel 1 and 4 was measured while the delay between channel 1 and 3 was increased. Under normal conditions no crosstalk could be measured. To be sure that the setup is sensitive to crosstalk we induced a crosstalk in the feeding of the asynchronous signals to the channels 3 and 4 by using a flat-band cable instead a shielded twisted pair cable. With this modification a shift of the mean value of about 170 ps was observed (Fig. 10).

V. CONCLUSION

A new FPGA-TDC based on the TDL method has been implemented. With the use of newly proposed methods the

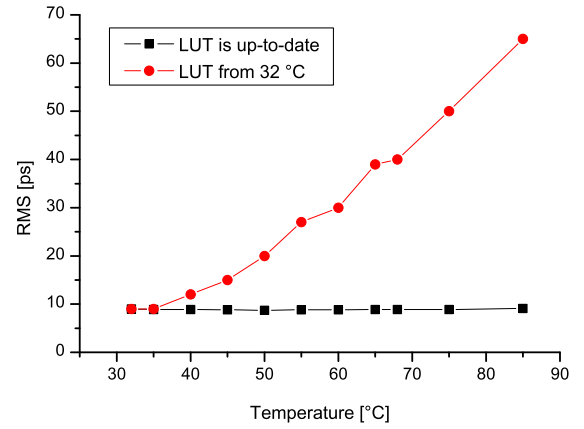


Fig. 8. The dependance of the RMS value on increasing temperature with continuous calibration and with the usage of the calibration table from a single temperature region.

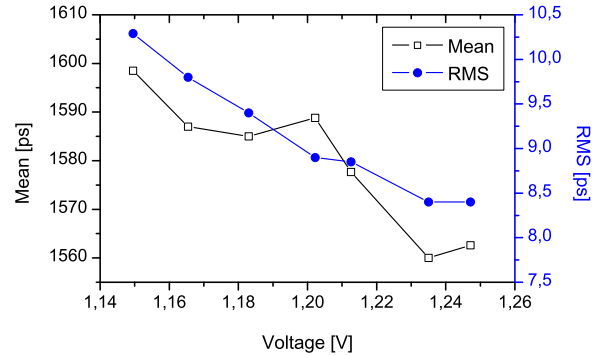


Fig. 9. The effect of voltage variation on the mean and the RMS value.

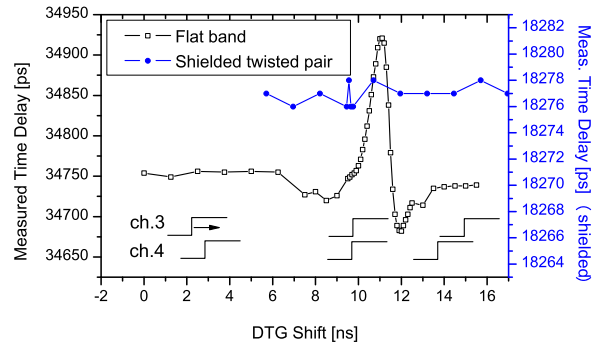


Fig. 10. The crosstalk measurement: The measurements of a fixed time interval between channel 1 and 4 with shielded twisted pair signal feeding and with the flat band cable feeding.

resolution of a TDC channel could be improved from 11 ps ($16 \text{ ps}/\sqrt{2}$) to 6 ps. In previous implementations the amount of used resources and/or the dead time increased, when methods for improving the resolution on a given FPGA were used – the use of several channels for one measurement or the Wave Union method. The usage of the latter method indeed allows to save the resources for an other delay chain but necessitates simultaneously a more complex encoder design. In our design this method worked fine with the same amount of resources by using only one rising and one falling edge for a minimalist Wave Union. With this scheme 48 channels could

be implemented on the target FPGA, each channel showing the same performance as presented above.

The experimental results on the influence of the voltage and temperature variations demonstrate that those can have a significant impact on the performance of the TDC and have to be taken into account during the design of the setup. However, these influences can be eliminated by using common calibration methods and technology. The supply voltage of the FPGA-TDC can be stabilized with commercial DC/DC converters over the full operating range in a ± 12 mV region without any special precautions on the PCB design. So, the maximum shift of the mean value can be limited to 7 ps in the worst case. On our hardware we didn't measure supply voltage variations above 2 mV. Temperature variations below 5 °C have no impact on the performance. For larger variations a calibration with the current measurement data is necessary.

VI. OUTLOOK

The use of the Wave-Union method is a trade-off between resource usage, dead time and resolution. It should be possible to increase the resolution of the TDC by using more than 2 transitions. So, the Wave Union is going to be extended in the next step. For this purpose a new encoder is needed. A new efficient encoder that allows to maintain the low dead time and the high channel number is currently under construction. Additional measurements will be done to characterize the power consumption of the TDC.

ACKNOWLEDGMENT

The authors would like to thank Karsten Koch, Jan Hoffmann and Nikolaus Kurz for their helpful inputs.

REFERENCES

- [1] G. Agakishiev *et al.*: *The High-Acceptance Dielectron Spectrometer HADES*, Eur. Phys. J. A, vol. 41, no. 2, 2009, pp 243-277.
- [2] J. Kalisz, R. Szplet, J. Pasierbinski and A. Poniecki: *Field-Programmable-Gate-Array-based time-to-digital-converter with 200-ps resolution*, IEEE Trans. Instrum. Meas., vol. 46, Feb. 1999, pp. 51-55.
- [3] M. S. Andaloussi, M. Boukodoum and E. M. Adoulhamid: *A novel time-to-digital converter with 150 ps time resolution and 2.5 ns pulse-pair resolution*, Proc. 14th Int. Conf. Microelectron., December 11-13, 2002, pp. 123-126.
- [4] D. Xie, Q. Zhang, G. Qi and D. Xu: *Cascading delay line time-to-digital converter with 75 ps resolution and a reduced number of delay cells*, Rev. Sci. Instrum., vol. 76, no. 014701, 2005.
- [5] J. Wu and Z. Shi: *The 10-ps wave union TDC: Improving FPGA TDC resolution beyond its cell delay*, IEEE Nuclear Science Symp. Conf. Rec., Oct. 19-25, 2008, pp. 3440-3446.
- [6] S. Junnarkar, P. O'Connor and R. Fontaine: *FPGA based self calibrating 40 picosecond resolution, wide range Time to Digital Converter*, IEEE Nuclear Science Symp. Conf. Rec., Oct. 19-25, 2008, pp. 3434-3439.
- [7] C. Favi and E. Charbon: *A 17 ps Time-to-digital Converter Implemented in 65nm FPGA Technology*, Proc. ACM/SIGDA Int. FPGA Symp., Feb. 22-24, 2009, pp. 113-120.
- [8] J. Kalisz: *Review of methods for time interval measurements with picosecond resolution*, Metrologia, vol. 41, 2004, pp. 1732.
- [9] J. Song, Q. An and S. Liu: *A High Resolution Time-to-Digital Converter Implemented in Field-Programmable-Gate-Arrays*, IEEE Trans. Nucl. Sci., vol. 53, no. 1, Feb. 2006, pp. 236241
- [10] J. Wu: *On-Chip processing for the wave union TDC implemented in FPGA*, Proc. IEEE-NPSS Real Time Conf. Rec., May 1015, 2009, pp. 279282.
- [11] E. Saell and M. Vesterbacka: *A Multiplexer Based Decoder for Flash Analog-to-Digital Converters*, TENCON, vol. 4, Nov. 21-24, 2005, pp. 250-253.