

---

This is an electronic reprint of the original article.  
This reprint may differ from the original in pagination and typographic detail.

Antonov, Yury; Stadius, Kari; Kosunen, Marko; Ryyänen, Jussi

## Open-loop all-digital delay line with on-chip calibration via self-equalizing delays

*Published in:*

23rd European Conference on Circuit Theory and Design (ECCTD 2017)

*DOI:*

[10.1109/ECCTD.2017.8093344](https://doi.org/10.1109/ECCTD.2017.8093344)

Published: 01/01/2017

*Document Version*

Peer-reviewed accepted author manuscript, also known as Final accepted manuscript or Post-print

*Please cite the original version:*

Antonov, Y., Stadius, K., Kosunen, M., & Ryyänen, J. (2017). Open-loop all-digital delay line with on-chip calibration via self-equalizing delays. In *23rd European Conference on Circuit Theory and Design (ECCTD 2017)* (pp. 1-4). (European Conference on Circuit Theory and Design). IEEE.  
<https://doi.org/10.1109/ECCTD.2017.8093344>

---

This material is protected by copyright and other intellectual property rights, and duplication or sale of all or part of any of the repository collections is not permitted, except that material may be duplicated by you for your research use or educational purposes in electronic or print form. You must obtain permission for any other use. Electronic or print copies may not be offered, whether for sale or otherwise to anyone who is not an authorised user.

# Open-loop all-digital delay line with on-chip calibration via self-equalizing delays

Yury Antonov, Kari Stadius, Marko Kosunen and Jussi Ryyänen  
 Department of Electronics and Nanoengineering, School of Electrical Engineering  
 Aalto University, Espoo, Finland, P.O. Box 15500 Aalto  
 Email: yury.antonov@aalto.fi

**Abstract**—A novel calibration technique and its all-digital implementation for the open-loop delay line is presented. Fully autonomous approach iteratively compares each digitally-controlled delay stage of the line with an on-chip reference delay, correspondingly tuning selected stage and memorizing associated settings. After correcting all individual stages, the total delay of the line is compared with the external period and reference delay is then readjusted. When on-board settling monitor observes repetition of reference delay settings, it locks delay line by applying previously collected settings. The delay line is shown to lock in presence of 30% static offset of delays from their designed values. Furthermore, random spread of delays worsened by 3 times (5%<sub>PP</sub> to 15%<sub>PP</sub>) results in only 2% decline (3%<sub>PP</sub> to 5%<sub>PP</sub>) after applying the proposed calibration to 16-delays line.

## I. INTRODUCTION

Delay-locked loops (DLLs) are widely employed in modern applications for the main purposes of data synchronization (serial-to-parallel converters), clock multiplication (frequency synthesizers) and multi-phase distribution (phase modulators). Above applications leverage multi-output nature of DLL and arbitrary switching between outputs, while other inherent advantages over PLL counterpart include unconditional stability, faster settling and smaller dynamic power consumption when fed from low-frequency reference.

Typical DLL outlined in Fig. 1a contains a series of voltage-tuned delays excited from external stable frequency and a negative feedback loop, aligning total chain delay with the external cycle. With ongoing CMOS scaling, intrinsic delay of basic inverter is steadily reduced, requiring more delay stages to accommodate low reference frequencies. More stages provide more outputs, however additional space is consumed with stretched layout and mismatch errors are induced between the stages. The latter is especially detrimental in the clock multiplication use case (Fig. 1b), causing jitter patterns [1] and harmonic spurs [2] at the output.

In this paper we propose a new calibration technique and an all-digital implementation thereof, demonstrating several advantages over known art. Recent digitally assisted DLLs [3], [4] still utilize analog tuning voltage and rely on the retention time of the integration capacitor ( $C_{LF}$  in Fig. 1a). Digitalization offered in [5], [6] replaces voltage-controlled tuning with the code-controlled one but is not able to correct individual delays, limiting phase-to-phase accuracy. Calibration technique proposed in this paper achieves both *delay equalization* and

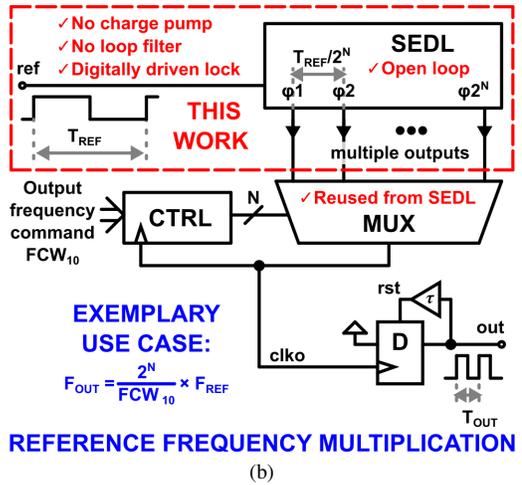
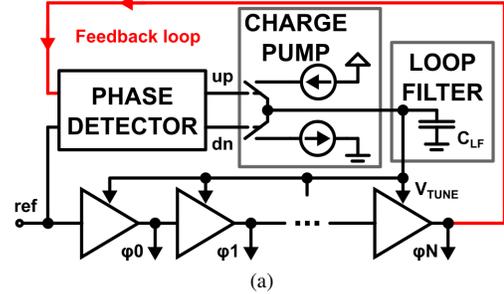


Fig. 1. (a) Conventional DLL and (b) described SEDL with a use case.

*delay line locking*. Different from [7], our approach excludes both bulky integration capacitor and analog charge-pump from the circuit, completely eliminating any analog tuning voltage controls. Contrasted to [8] our concept of *reference delay* and an *on-chip comparison* task does not require feeding delay line with a large number of input transitions for statistical tests. Furthermore, suggested architecture already includes multiplexers, as per use case in Fig. 1b, thus calibrating out also delays of the cells and wiring for the two channels of clock multiplication. Simulated with 16-delays line, this design achieves lock in presence of 30% static offset of delays from their intended values. Tested with random 5%<sub>PP</sub>, 10%<sub>PP</sub>, and 15%<sub>PP</sub> deterioration of delays spread, proposed calibration diminishes the spread to as little as 3%<sub>PP</sub>, 4%<sub>PP</sub> and 5%<sub>PP</sub>.

Section II of this paper describes overall architecture, with subsections detailing various building blocks. Simulation results are given in section III. Section IV concludes the paper.

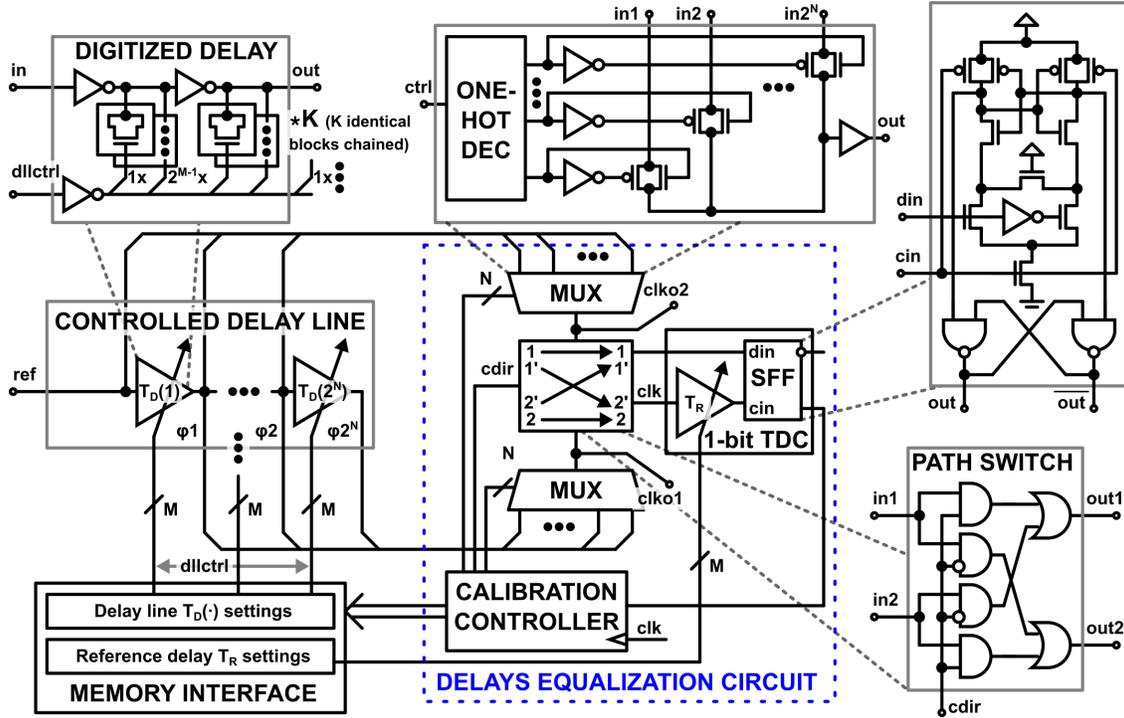


Fig. 2. Block diagram of the proposed SEDL - delay line architecture based on self-equalizing delays.

## II. PROPOSED DLL ARCHITECTURE

In conventional DLL designs, integration capacitor  $C_{LF}$  would hold the tuning voltage  $V_{TUNE}$  and should be refreshed by the feedback loop to compensate for charge leakage.

In the proposed SEDL architecture (Fig. 2), digitization of delay line enables memory interface to store settings for the individual stages, thus eliminating  $C_{LF}$  and analog voltage tuning altogether. Storing previous settings allows on-chip algorithm to detect repetitions and eventually lock digitized delay line through the three main phases described in subsection B.

The other enabling concept in this architecture is propagation delay comparison. On-chip comparing circuit serves two purposes: first, it contrasts arrival delays of the reference rising edge and its replica, and second, it provides clock  $clk$  for the calibration controller to switch between states. By using single edge and its replica, such calibration is theoretically insensitive to duty-cycle or frequency variations of the external reference.

### A. Controlled digitized delays line

In all-digital implementation of the delay line, analog delaying stages (e.g. based on voltage-controlled transconductance in the differential pair [7]) are replaced with identical buffers, loaded with digital varactors - NMOS transistors with source and drain tied together as one terminal, and gate serving as the other terminal. These transistors are paralleled in the common-centroid layout to achieve binary weighting up to size  $M$  ( $1x, 2x, \dots, 2^{M-1}x$ , where  $x = W/L$ , Fig. 2, top

left). Individual inverter's output then sees total capacitance:

$$C_D = \sum_{i=0}^{2^M-1} C_{LOW,i} + \sum_{i=0}^{M-1} dllctrl(i) \cdot \Delta C_0 \cdot 2^i, \quad (1)$$

where  $\Delta C_0 = C_{HIGH,0} - C_{LOW,0}$  is the difference between capacitance states switchable by the least significant bit (LSB) of digital bus  $dllctrl$ .

In this design  $N = 4$  and the target reference frequency  $ref = 200\text{MHz}$  which makes required delay per stage  $T_D = T_{REF}/2^N = 312.5\text{ps}$ . Fig. 3a sketches digitized delay of the paired capacitively loaded inverters, giving for  $K$  pairs:

$$T_D = K \cdot (t_{PLH} + t_{PHL}) = 0.69K \cdot (R_P C_D + R_N C_D), \quad (2)$$

where Elmore model is used to estimate loaded inverter propagation delay. For  $M = 4$ , channel resistances of  $R_P = R_N = R = 0.33\text{K}\Omega$  and ballpark values  $C_{LOW} = 20\text{fF}$  and  $\Delta C_0 = 5\text{fF}$ , one can estimate  $K$  by setting  $dllctrl$  content to  $2^{M-1} - 1$  ( $\approx$  middle of the code range):

$$K = \left\lfloor \frac{T_D}{1.38R \cdot ((2^M - 1)C_{LOW} + (2^{M-1} - 1)\Delta C_0)} \right\rfloor, \quad (3)$$

$$K = \lfloor 2.05 \rfloor = 2.$$

From the denominator of (3), code-controlled delay of the stage is 278...341ps with LSB step of 4.5ps, while the total chain delay of  $2^4$  stages covers the range 4.45...5.46ns.

Due to inherently stretched layout of the delay line, process variations affect digital varactors and buffer transistors, shifting individual delays from their designed values. However after the manufacturing, mismatch-induced errors are fixed and can be calibrated out.

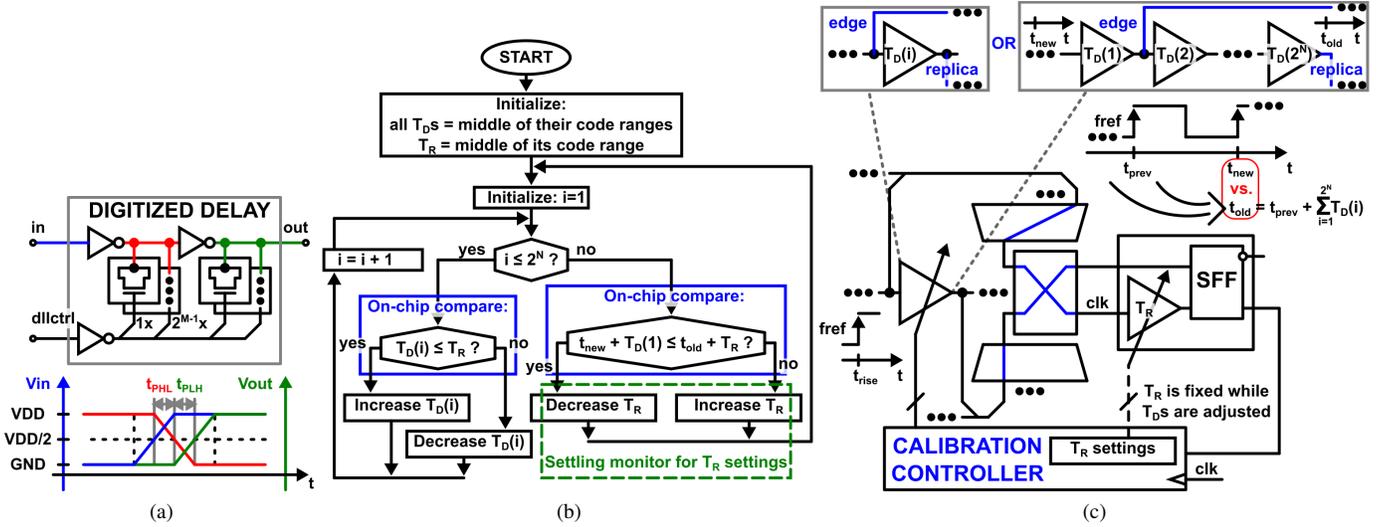


Fig. 3. (a) Loaded inverter propagation delay, (b) calibration algorithm flow-chart and (c) delays comparison implementation.

### B. Calibration algorithm

Calibration approach developed in this paper corrects mismatch errors between delaying stages and at the same time locks delay line to the external reference period. This is possible in the case of digitized delay line, since bulky integration capacitor  $C_{LF}$  is now “broken” into  $2^N \times (2^M - 1)$  adjustable parts and equally spread across the entire line. The goal of calibration controller therefore is to equalize physical delays and establish lock without iterating through all  $(2^N)^{(2^M)}$  combinations of the delay line settings available on this chip.

In order to calibrate individual delaying stages, any stage must be independently accessible and for this purpose calibration algorithm governs two multiplexers as shown in Fig. 2. We note here that this arrangement fits perfectly to Multiplying Delay-Locked Loop (MDLL) concept [9] and will calibrate out also propagation delays of the cells and wiring for the two channels of clock multiplication:  $clk01$  and  $clk02$ . These multiplexers are connected to delay line outputs with a mutual shift by 1 delaying stage, which allows to reach both input and output of a chosen stage.

Calibration algorithm flow-chart in Fig. 3b is realized in a form of Mealy Finite-State Machine type controller. In the first phase of the algorithm, after initializing all the delays to the middle of the available code range, the controller iteratively compares individual delays  $T_D$  with the reference delay  $T_R$  as in Fig. 3c, top left. At this phase,  $T_R$  is presumed to be correct and after comparison with  $T_D$  is moved one step towards  $T_R$ .

In the second phase, after calibrating all individual delays, the controller proceeds to comparison of total line delay with the external reference period as in Fig. 3c, top right. At this phase, all delay line settings are presumed to be correct and propagation time of the *new* reference rising edge is contrasted to that of the *old* one (i.e. previous rising edge that has passed through the entire delay line). Depending on comparison result  $T_R$  is adjusted in the correct direction: if total delay of the line

is less than reference period,  $T_R$  delay setting is incremented.

Before returning to individual delays readjustments, in the third phase the controller checks short history of  $T_R$  settings to identify repetitions and lock the state with previous settings.

In summary, presented algorithm aligns delay stages with the reference delay which settles to the desired  $T_{REF}/2^N$ .

### C. Phase selection multiplexer

A rising edge propagating along the delay line needs to be multiplexed at a desired point to be compared with its replica in TDC block. Multiplexing functionality, opened up in Fig. 2, intentionally separates control block from the available propagation paths. This enables custom design of transmission gates and combining node (wired-OR) to minimize their delay contribution, while one-hot decoder can be placed and routed with known automated tools. Phase selection proceeds in two steps: first, one-hot decoder performs N-to- $2^N$  code transformation and activates particular propagation path, then, rising edge travels down the path to the output.

### D. Signal path switch

Signal path switch in the presented architecture enables a novel on-the-fly-calibration mode for the single channel clock multiplication. Such mode allows to stabilize frequency multiplying operation against voltage and temperature variations. In this regime, one of the multiplexers performs iterative edge combining as in MDLL, while the other multiplexer simultaneously connects to the previous output of the delay line, effectively selecting delay stage for the measurement (Fig. 3c, top left. For example, if one multiplexer switched from output 3 to  $2^N$ , the other multiplexer would switch from output 2 to  $2^N - 1$ ). Flexibility attained by applying the second multiplexer and the signal path switch allows to “decouple” frequency synthesis from the on-chip measurements. To be able to equally utilize both multiplexers, criss-cross type switch was designed for this application.

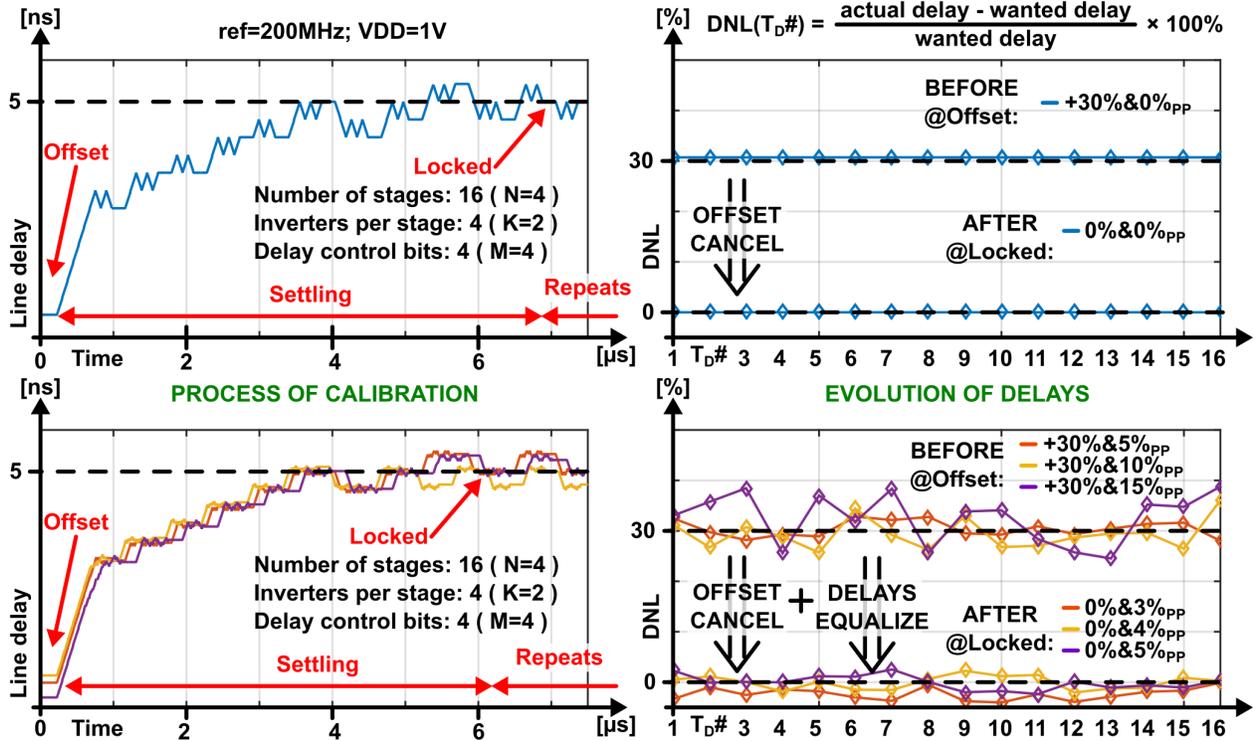


Fig. 4. Calibration of the SEDL with 30% static shift and 0%<sub>pp</sub>, up to 5%<sub>pp</sub>, 10%<sub>pp</sub>, 15%<sub>pp</sub> random shifts of individual delays from their ideal initial values.

### E. 1-bit Time-to-Digital Converter

TDC encompasses differential edge-triggered latch and a single delaying stage (identical to  $T_D$ ) serving as a time-reference. Comparison result between two propagation delays of the same rise transition is held in this latch until the next readout by calibration controller. Depending on the chosen direction for the path switch, either half of the delay line can produce clock  $clk$  for the latch pin  $cin$ . Calibration controller operates from the same clock  $clk$  and there is no need in special reset/clear signals for this latch between readouts.

### III. SIMULATION RESULTS

For behavioral simulations of presented design a scalable randomly-initialized VHDL/VHDL-AMS simulation framework was developed. Simulation results gathered in Fig. 4 confirm that 30% offset from intended delays due to process spread can be reliably canceled with the proposed circuitry. Furthermore, open-loop digitized delay line can be consistently locked to 200MHz reference frequency via proposed algorithm even in presence of up to 15%<sub>pp</sub> random mismatch between individual delays.

### IV. CONCLUSION

This paper suggests a calibration approach of delay line locking through individual delays comparison and equalization. Applied to open-loop delay line, presented algorithm implementation effectively locks 16-delays chain to an external 200MHz reference period under 30% static and up to 15%<sub>pp</sub> random offsets of individual delays.

### ACKNOWLEDGMENT

The authors would like to thank Lyubov Milakhina for her valuable ideas during early development of the algorithm.

This research received funding from Academy of Finland.

### REFERENCES

- [1] R. van de Beek, E. Klumperink, C. Vaucher, and B. Nauta, "On jitter due to delay cell mismatch in DLL-based clock multipliers," in *Proc. IEEE Int. Symp. Circuits Syst.*, vol. 2, May 2002, pp. 396–399.
- [2] A. Ojani, B. Mesgarzadeh, and A. Alvandpour, "Modeling and analysis of harmonic spurs in DLL-based frequency synthesizers," *IEEE Trans. Circuits Syst. I: Regular Papers*, vol. 61, no. 11, pp. 3075–3084, Nov. 2014.
- [3] A. Hejazi, S. Kazemini, and R. Abdollahi, "A digitally assisted 20MHz–600MHz 16-phase DLL enhanced with dynamic gain control loop," in *Proc. European Conf. Circuit Theory Design*, Aug. 2015, pp. 15B.1–4.
- [4] Y. Antonov, K. Stadius, and J. Rynänen, "A charge limiting and redistribution method for delay line locking in multi-output clock generation," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2017, pp. 1806–1809.
- [5] M. T. Pasha, Y. A. Shah, and J. Wikner, "A wide range all-digital delay locked loop for video applications," in *Proc. European Conf. Circuit Theory Design*, Aug. 2015, pp. 5.1–4.
- [6] B. Mesgarzadeh and A. Alvandpour, "A low-power digital DLL-based clock generator in open-loop mode," *IEEE J. Solid-State Circuits*, vol. 44, no. 7, pp. 1907–1913, Jul. 2009.
- [7] H.-H. Chang, J.-Y. Chang, C.-Y. Kuo, and S.-I. Liu, "A 0.7–2-GHz self-calibrated multiphase delay-locked loop," *IEEE J. Solid-State Circuits*, vol. 41, no. 5, pp. 1051–1061, May 2006.
- [8] F. Baronti, L. Fanucci, D. Lunardini *et al.*, "A technique for nonlinearity self-calibration of DLLs," *IEEE Trans. Instrum. Meas.*, vol. 52, no. 4, pp. 1255–1260, Aug. 2003.
- [9] T. Rapinjoja, Y. Antonov, K. Stadius, and J. Rynänen, "Fractional-N open-loop digital frequency synthesizer with a post-modulator for jitter reduction," in *IEEE Radio Freq. Integr. Circuits Symp. Dig. Papers*, May 2016, pp. 130–133.