FH University of Applied Sciences TECHNIKUM WIEN

MASTER THESIS

Thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Engineering at the University of Applied Sciences Technikum Wien - Degree Program Mechatronics/Robotics

Characterization and calibration of rad-hard bandgap voltage reference designed for high-energy physics application

By: Yannick Konstandin, B.Eng.

Student Number: 2010331012

Begutachter: Dipl.-Ing., Christoph Mittermayer
Begutachter: MSc. Wilfried Wöber

Bergamo, September 2, 2022

Declaration

Ich erkläre hiermit an Eides statt, dass ich die vorliegende Arbeit selbständig angefertigt habe. Die aus fremden Quellen direkt oder indirekt übernommenen Gedanken sind als solche kenntlich gemacht. Die Arbeit wurde bisher weder in gleicher noch in ähnlicher Form einer anderen Prüfungsbehörde vorgelegt und auch nicht veröffentlicht. Ich habe zur Kenntnis genommen, dass die Arbeit noch auf Plagiate geprüft wird. Hinweis: Bei mehrfachem Upload besteht die Möglichkeit, dass es zu einem Plagiat kommt, da die Arbeit im System ist.

Ich räume der Fachhochschule Technikum Wien das Recht ein, das/die übermittelte/n Dokumente elektronisch zu speichern und - im Falle von Masterarbeiten - in Datennetzen öffentlich zugänglich zu machen. Ich räume der Fachhochschule Technikum Wien ferner das Recht zur Konvertierung zum Zwecke der Langzeitarchivierung unter Beachtung der Bewahrung des Inhalts ein.

Ich erkläre außerdem, dass von mir die urheber- und lizenzrechtliche Seite (Copyright) geklärt wurde und Rechte Dritter der Publikation nicht entgegenstehen.

Ich bestätige hiermit, dass ich die eidesstattliche Erklärung mittels digitaler Signatur unterzeichnet habe.

Bergamo, September 2, 2022

Kurzfassung

Bandgap-Referenzen sind sehr verbreitet in integrierten Schaltungen, bei denen eine konstante Spannung oder ein konstanter Strom als Bezugswert benötigt wird. Beispiele dafür sind LDO, ADC/DAC, PLL oder DRAM. Diese Geräte benötigen eine Referenzspannung oder Strom, welcher in DC Form bereitgestellt wird. Diese Referenz soll möglichst unabhängig von der Temperatur, dem Prozess und der Eingangsspannung sein. Für hochleistungs Physik Anwendungen, wie es im arcadia projekt der Fall ist, muss zudem die Resistenz gegen Strahlungsschäden sichergestellt werden. Gängige Werte für die Referenzspannung liegen im Bereich von 1.2 V bis 1.3 V. Da sich das arcadia das Ziel gesetzt hat den Leistungsverbrauch des Testboards so gering wie möglich zu halten, wurde die Referenzspannung auf 600 mV gesenkt. Dies führte zu einem neuen Ansatz des Layouts.
Trägt man die Referenzspannung über der Temperatur auf, so ergibt im ideal Fall eine Linie bei 600 mV mit einer Steigung von 0. Da in der Realität vorallem Störungen durch Produktionsabweichungen auftreten kann dies nur schwer erreicht werden. Im design der arcadia bandgap wurden deshalb zwei programmierbare 4-bit-Widerstände zur Feinjustage der Steigung und des Absolutwertes der U_{BGR}/T Kurve eingebaut, um die zuvor beschriebenen Abweichungen zu kompensieren. Das arcadia Testboard verfügt über insgesamt 16 solcher Bandgap-Referenzen, von denen jede über die erwähnten 4-bit-Widerstände verfügt. In dieser Arbeit wurden zwei wichtige Eigenschaften der Bandabstandsreferenzen untersucht.
Der erste Teil der Charakterisierung bestand darin, die bestmögliche Konfiguration der 4-bit-Widerstände für die jeweilige Referenzspannung zu finden. Bestmöglich heißt in diesem Zusammenhang, dass die Temperaturabweichung, sowie die Abweichung zum konzipierten Betriebspunkt so gering wie möglich werden soll. Um dies zu realisieren wurden alle Referenzspanungen für alle möglichen Konfigurationen in einem Bereich von –40 ℃ to 70 ℃ gemessen. Die Ergenisse wurden danach in einer detaillierten Analyse mit Matlab© ausgewertet. Das Resultat für die bestmögliche Konfiguration der 16 Referenzspannungen bei einer Eingangsspannung von 1.2 V weist einen Mittelwert von ~600 mV und eine Standardabweichung von ~3 mV auf. Die zweite Untersuchung geht der Frage nach, wie sich die Referenzspannungen auf Änderungen in der Eingangsspannung verhalten. Im folgenden werden die vorgehensweise zu dem Messungen, sowie zur deren Auswertung genaustens beschrieben.

Schlagworte: Kleinspannungs Bandgapreferenz, niedrige Leistung, Strahlungshärte, Strombetrieb CMOS, Temperaturkompensation, arcadia

Abstract

Bandgap references are very common in integrated circuits where a constant voltage or current is required as a reference. Examples of this are LDO, ADC/DAC, PLL or DRAM. These devices require a reference voltage or current, which is provided in a DC quantity. This reference should be as independent as possible from temperature, the process and the input voltage. For high-energy physics applications, as is the case in the arcadia project, radiation-harness must also be ensured. In addition, the magnitude of the common reference voltage is in the range of 1.2 V to 1.3 V. Since the arcadia aims to keep power consumption as low as possible, the reference voltage was also lowered to 600 mV. This led to a new approach to the layout.
If the reference voltage is plotted against temperature, the ideal case is a line at 600 mV with a slope of 0. In reality, this is difficult to achieve due to production variations. In the design of the arcadia bandgaps, two programmable 4-bit resistors were included to fine-tune the slope and absolute value of the U_{BGR}/T curve to compensate for the deviations described above. The arcadia testboard has a total of 16 bandgap voltage references, each of which has two of the 4-bit resistors mentioned above. In this work, two important properties of the bandgap references were investigated.
The first part of the characterization was to find the "best possible" configuration of the 4-bit resistors for each reference voltage. In this context, "best possible" means that the temperature deviation, as well as the deviation from the designed operating point, should become as small as possible. To realize this, all reference voltages of all possible configurations were measured in a range from –40 °C to 70 °C. The results were then evaluated in a comprehensive analysis with Matlab©. The result for the best possible configuration of the 16 reference voltages at an input voltage of 1.2 V has a mean value of ~600 mV and a standard deviation of ~3 mV.
The second investigation addresses the question of how the bandgap voltages respond to changes of the power supply. In this work, the procedure of the measurements as well as the evaluation of the results are described in detail.

Keywords: low-voltage bandgap reference, low power, radiation-hardness, current mode CMOS, temperature compensation, arcadia

Acknowledgements

This master thesis was written in cooperation with the University of Bergamo and represents a milestone in my studies at the University of Applied Sciences Technikum Vienna in the Mechatronics/Robotics program. At this point I would like to take the opportunity to thank everyone who supported me in the preparation of this master thesis.

A special thanks goes to my supervisor in Bergamo Prof. Luigi Gaioni, who allowed me the chance to make my contribution to the arcadia project, for his support and constant willingness to discuss. I would also like to thank Prof. Gianluca Traversi for his great help.
Furthermore, I would like to thank MSc. Wilfired Wöber for taking over the second correction. I would like to thank all employees of the electronics laboratory for their help and advice.

Special thanks are also due to Dipl.-Ing. Christoph Mittermayer, who supported me as a supervising professor at the University of Applied Sciences Technikum Vienna in the preparation of this thesis.

Contents

1 Introduction

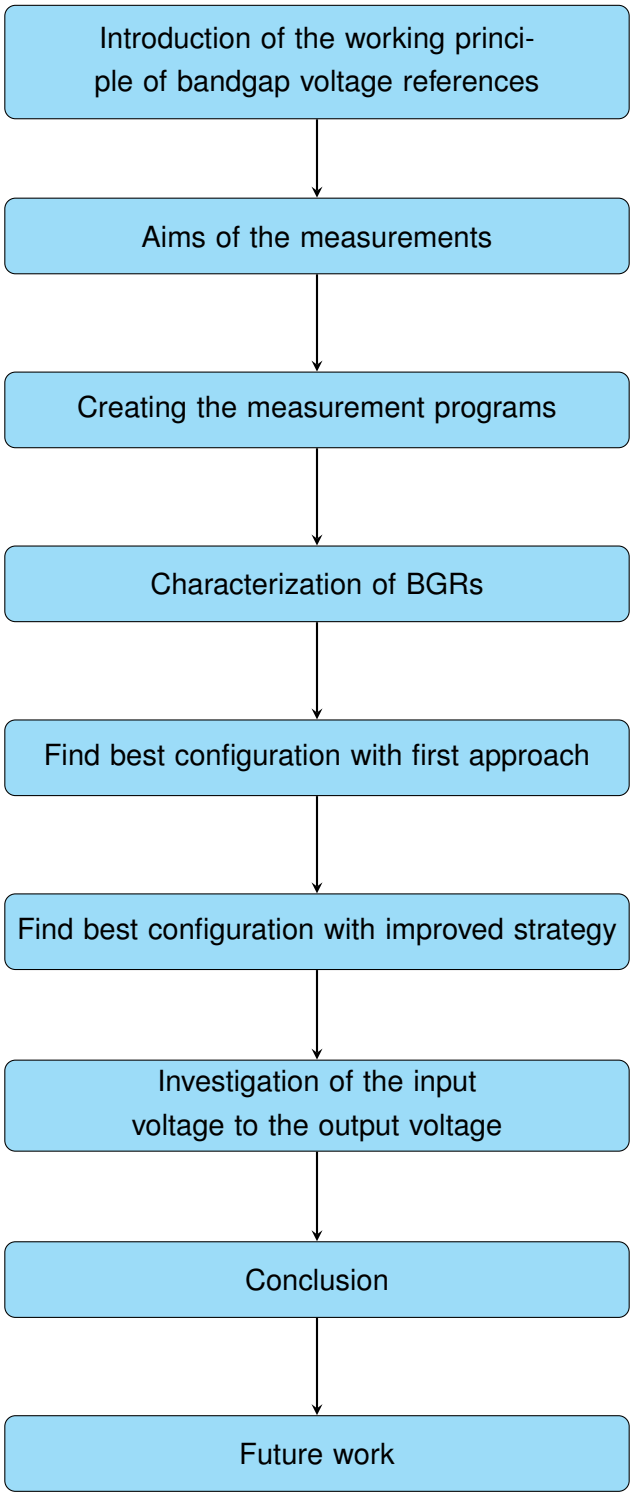
The project ***Advanced Readout CMOS Architectures with Depleted Integrated sensor Arrays***, shortly arcadia, has been set up by a subgroup of the *Istituto Nazionale di Fisica Nucleare* (INFN) with sites in Frascati, Padova, Perugia, Trento, and Turin. In this project, monolithic fully depleted pixel sensors are being developed in collaboration with the *Conseil européen pour la recherche nucléaire* (CERN). Compared to hybrid pixel sensors, these sensors have the advantage that they can be manufactured at lower cost while consuming less material. They were developed in a special 110 nm CMOS technology. Since this idea is still relatively new, the arcadia scientists have set themselves the goal of building a large-scale demonstrator.
The sensors consist of various ADCs, DACs and LDOs. For all these components it is important to provide a constant reference voltage. The less this reference depends on the input voltage, the process and the temperature, the more accurately the quantities recorded by the sensor can be digitized and converted. Since these sensors are used in high-energy physics applications, where the radiation reaches up to 1 Grad and 10^{16} neq/cm^2 with the phase 2 upgrade at CERN [16], they must also have radiation-hardness.
These references are realized with so-called bandgap voltage references (BGR). However, since these references are based on the pn-junction, they exhibit a certain dependence on the temperature even after temperature compensation. In addition, the manufacturing process and fluctuations in the power supply cause irregularities in the U_{BGR}/T curve. To analyze these deviations, a total of 16 such bandgap voltage references have been implemented in the arcadia testboard, each of which has two programmable 4-bit resistors. With these resistors the temperature coefficient of the reference voltage, which in this work will be called **slope** and the absolute value, which will be called **mean**, of the U_{BGR}/T curve can be fine-tuned.
In most cases one of the following phenomenons occurs:

- very low temperature dependence
- very small deviation from the operating point of 600 mV

A test point (TP) is a tap point on the testboard, which allows to measure one of the 16 bandgap voltages. Although the 4-bit resistors have been implemented in the design, they have hardly been used for compensation, since their behavior has so far only been investigated in simulations. In this work, the behavior of the bandgap voltages for different resistor values is to be investigated in detail with measurements and an optimal configuration for each TP is to be made available especially for the testboard. In order to find the optimal configuration of the two 4-bit resistors for each TP, the bandgap voltages were measured for all possible combinations in a temperature range of –40 °C to 70 °C. In addition, the dependence of the bandgap voltage

on the input voltage was investigated. Again, voltage measurements were made at the same temperature range for input voltages of 0 V to 1.32 V and vice versa. In the next step, the data collected here was evaluated in detail using Matlab©. In order to find an optimal configuration for the two resistors, a good compromise between low temperature dependence and low deviation from the design point had to be found. An initial heuristic method for finding the optimum configuration was improved using the insights gained from the detailed analysis of the characteristics of the bandgap references. In this work, the functioning of the bandgap voltage reference, the execution of the measurements and, above all, the evaluation of the data are thoroughly described.

1.1 Structure of the work



2 State of the art

This chapter gives a short overview of the principle of bandgap references in general and of the special functions realized in the arcadia project.

2.1 Electronic band structure

The Bohr atomic model is based on the fact that atoms consist of a positively charged atomic nucleus surrounded by negatively charged electrons. These electrons can move on individual orbits, whereby each orbit can be assigned a certain energy value, which increases with increasing distance from the atomic nucleus. By plotting the possible energy levels in a diagram, one obtains a so-called line diagram. The distance W between the two energy levels corresponds to the energy that must be applied to move an electron from an inner orbit to an outer orbit. This energy can be applied by increasing the temperature or by irradiation with light. At rest, electrons always strive for the lowest-energy state.

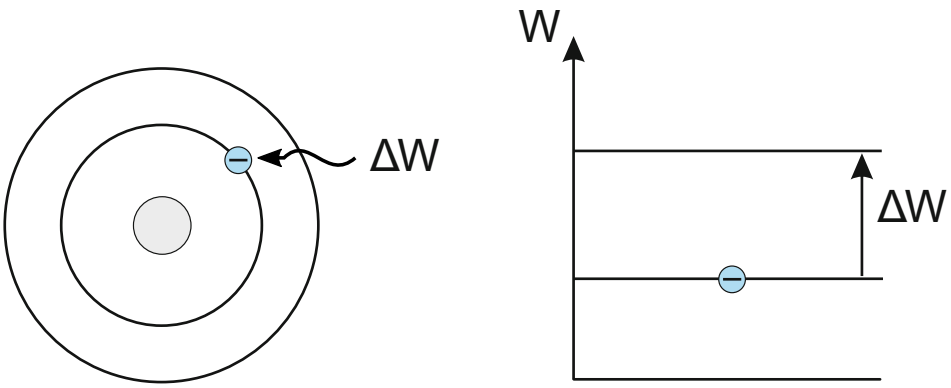


Figure 1: Model of an atom with two energy states and corresponding line diagram [3].

Since many atoms interact with each other in solids, the splitting into many single states takes place. Therefore, in this case, one no longer speaks of individual energy states, but of energy bands. The lower energy level is called valence band, the level at higher energy is called conduction band. In this region there are no energy states which can be occupied by electrons. The distance between the upper valence band edge and the lower conduction band edge is called bandgap. [3]

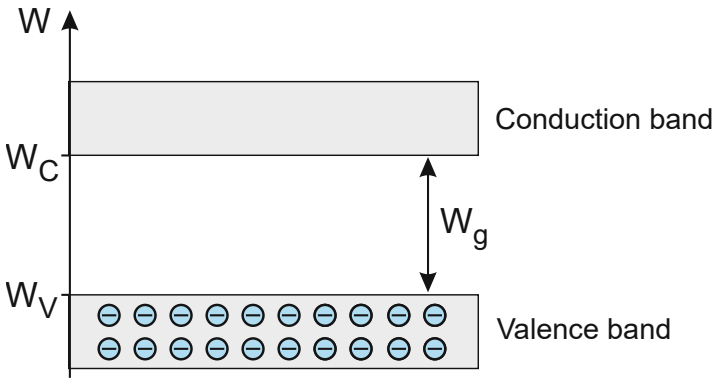


Figure 2: Band diagram of a semiconductor. At T = 0 K, the valence band is fully occupied by electrons, while there are no electrons in the conduction band [3].

Silicon forms the basis of almost all conventional semiconductors. Since it also forms the basis for the transistors in this experiment, its structure will be explained in more detail below. If we look at silicon on the atomic level, we see that there are four electrons on the outer shell, which can form a bond with neighboring electrons. These electrons are called valence electrons because of their energy level. Since the inner electrons are irrelevant for the bonding properties, only the valence electrons are considered in the following. The silicon atoms in a crystal lattice are arranged in a periodic, spatial structure. In this structure, each of the four valence electrons forms a bond with an adjacent atom. At absolute zero, all electrons are bound and no free electrons are available. In this state, no charge transport can take place. In Figure 2, this state is illustrated by the fact that all electrons are in the valence band and the conduction band is unoccupied.

If the silicon crystal is now supplied with energy in the form of heat, this enables individual electrons to break up the crystal lattice. This process corresponds to the electron transfer from the valence band to the conduction band. This electron is now no longer part of a bond and is thus available for charge transport, since it can move freely. Thus, the higher the temperature, the more electrons overcome the band gap. Therefore, the number of free electrons in the semiconductor increases with increasing temperature.

The number of free electrons is also dependent on the distance between valence and conduction band. For a very small to no distance W_g, a large number of electrons can pass from the valence band to the conduction band and the conductivity of the material increases. These materials belong to the category of **metals**. If the distance W_g is larger, however, fewer electrons succeed in this transition. These materials are called **semiconductors**. If the band gap is sufficiently large that only very few electrons succeed in *ascending* into the valence band, we speak of **insulators**. Silicon has a band gap of ~1 eV at 302 K, which is equivalent to about 1.602×10^{-19} J, and thus belongs to the category of semiconductors. [3]

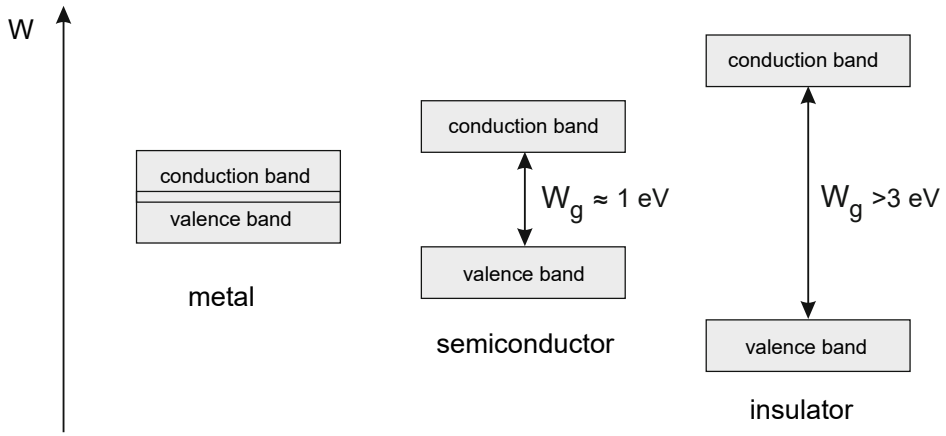
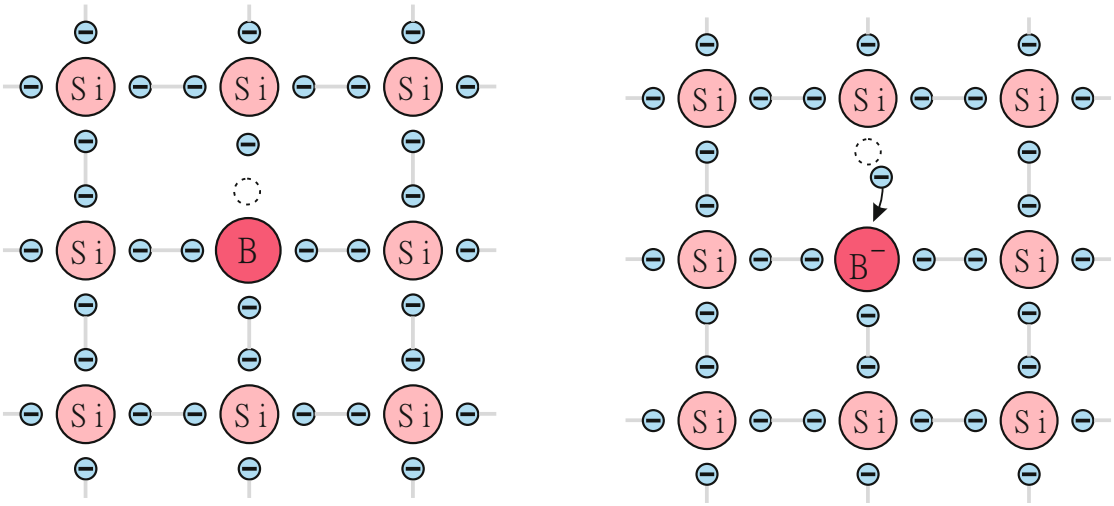


Figure 3: Comparison of band diagrams of metals, semiconductors and insulators [3].

2.2 Doping of semiconductors

Since the silicon semiconductor material is a quadrivalent element, it has hardly any free charge carriers in pure form. For the effective transport of charge, however, a charge carrier must be present. This is made possible by the incorporation of foreign atoms, known as doping. If a silicon crystal is doped with a pentavalent element, such as phosphorus, it is referred to as an *n-type* semiconductor, since one electron remains unsaturated when the two atoms bond. The bonding energy between the unsaturated electron and the phosphorus atom is very low, so that the thermal energy at room temperature is sufficient to release it from the atomic body. The phosphorus atom is also called donor (lat. *donare* "to give"), because it donates an electron. At thermodynamic equilibrium, the electron density of the n-doped semiconductor is significantly higher than that of the hole density, which is why the electrons are called majority charge carriers.

In the p-doping of semiconductors, exactly the opposite happens. Some atoms of the silicon crystal are substituted by a trivalent element, such as boron (fig. 4a). So there is a free space (hole) on the outer shell of the atoms, that can easily accept electrons (fig. 4b). Because the boron atom can accept an electron it is also called acceptor (lat. *receiver*). If an electron takes this place, it leaves a hole at another atomic bond. This movement of electrons can be seen the other way around as a migration of positively charged holes. In *p-type* semiconductors, the hole density dominates over the electron density, which is why the holes in this case are called majority charge carriers. In summary, electrons are responsible for charge transport in n-type semiconductors and holes in p-type semiconductors. [3]



(a) Doping the silicon crystal lattice with trivalent boron atoms creates free spaces in the crystal lattice

(b) P-doped lattice can now accept an electron from another bond for T>0 K

Figure 4: Through p-doping, the crystal lattice is now able to accept electrons. However, since the hole density outweighs the electron density, in p-type semiconductors the hole migration is considered as charge carrier [3].

2.3 The pn-junction of semiconductors

The pn-junction is formed by bringing together p-type and n-type semiconductors. It can be found in almost all semiconductor devices, especially in diodes and in npn, or pnp version also in transistors. Since a pn-junction is sufficient to generate bandgap voltage references, only this junction will be explained in this chapter. The n-doped region of the component is called cathode and the p-doped region is called anode. If a p-element and a n-element are brought together, a diffusion movement of holes in the direction of the n-region and of electrons in the direction of the p-region occurs due to the different charge carrier concentration. The diffusion of the electrons creates a region of low electron concentration near the junction in the n-region. The donor atoms, which are positively ionized at the beginning, are no longer neutralized in this region, which is why the semiconductor is positively charged in this region. In the p-region, the migration of the holes due to the negatively ionized acceptor atoms creates a region with a low hole concentration, which has a negative charge. This region, which opens up over both semiconductors, is called the space charge zone. Since dopant atoms in this region each have a different charge, an electric field that points from the n-region to the p-region is created. This in turn leads to a drift movement of the charge carriers, which behaves in the opposite direction to the diffusion movement. Figure 5 shows a pn-junction with the resulting diffusion and drift currents. The positive holes are drawn in red, electrons on the other hand in light blue. [3]

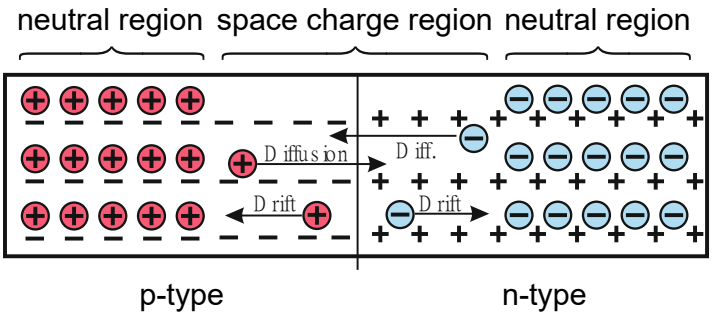


Figure 5: One-dimensional representation of the pn-junction with the directions for the diffusion and drift currents [3].

In thermodynamic equilibrium, drift and diffusion currents compensate each other, which is why no charge transport takes place. This assumption is also made to calculate the voltage across the pn-junction. In the neutral region, the charges of the dopant atoms and the free charge carriers compensate each other. Therefore, only the charge of the dopant atoms remains in the space charge region. These interrelationships give rise to the processes shown in figure 6. The indices x_p and x_n show how far the space charge region extends into the respective p- or n-region.

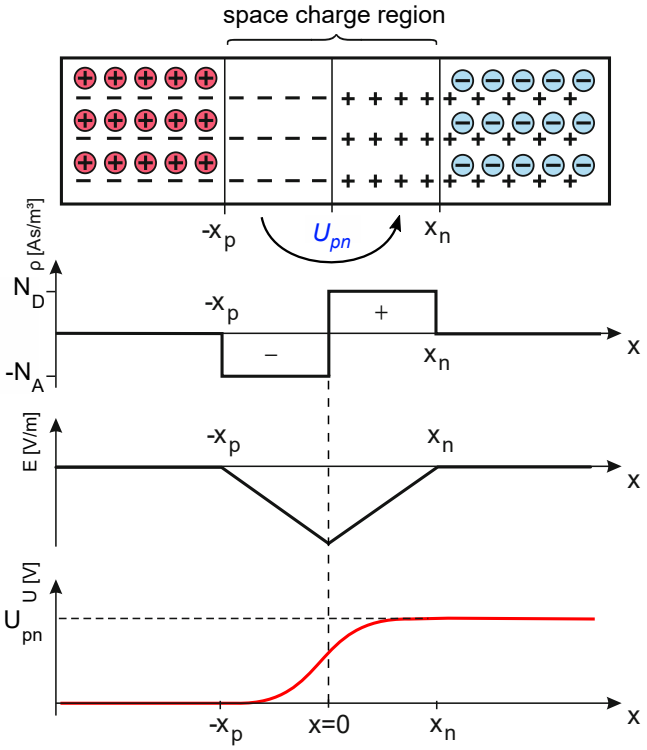


Figure 6: Space charge region and the corresponding diagrams of space charge density, electric field and junction voltage U_{pn} as a function of junction distance x [3].

For the electric field in the space charge region, the following term is obtained:

$$Edx = -\frac{D_n}{\mu_n}\frac{d_n}{n}, \text{ with } D_n = \frac{kT}{q} \cdot \mu_n \tag{1}$$

To obtain the electric field to the voltage U_{pn} across the space charge region from the electric field, the expression must be integrated.

$$U_{pn} = \frac{kT}{q}\int_{n_{p0}}^{n_{n0}} \frac{d_n}{n} \tag{2}$$

$$U_{pn} = \frac{kT}{q} ln\left(\frac{n_{n0}}{n_{p0}}\right) \tag{3}$$

For the n-region, due to the low ionization energy at room temperature, it can be assumed that the density of the free electrons n_{n0} is equal to the density of the doping atoms N_D. In the p-region, the same can be assumed for the hole density, which is why the electron density is $n_{p0} = n_i^2/N_A$. If we now substitute the charge carrier densities with the relations shown, we obtain for voltage U_{pn}:

$$\boxed{U_{pn} = \frac{kT}{q} ln\left(\frac{N_A N_D}{n_i^2}\right)} \tag{4}$$

The voltage applied to the pn-junction is also called diffusion potential and is in the range of 600 mV to 700 mV. As can be seen from equation 4, this voltage shows a dependence on temperature. How this dependence can be used specifically to generate a temperature-independent reference voltage will be explained in the next chapter. [3]

2.4 Working principle of a bandgap voltage reference

The basic principle of a bandgap voltage reference is based on the assumption of summing two voltages with opposite temperature coefficients (TC). The voltage that shows a rising behavior with temperature is called Proportional to Absolute Temperature component (PTAT), the voltage component with a negative behavior for rising temperatures on the other hand is called Complementary to Absolute Temperature component (CTAT). Since the positive component is usually much smaller than the negative one, it has to be amplified. Theoretically, the resulting reference voltage has a TC = 0. Figure 7 shows a depiction of how the two voltages can be added.

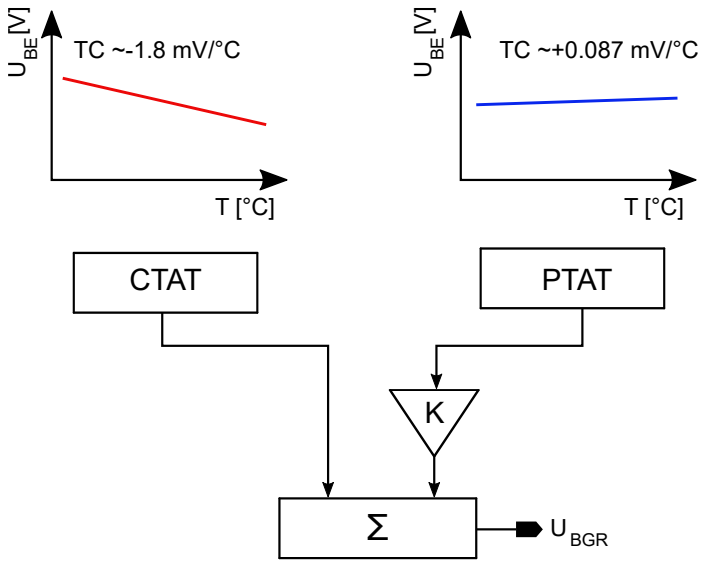


Figure 7: Principle of a conventional bandgap voltage reference with quantitative values of temperature coefficients [8].

Of all existing semiconductor devices, bipolar junction transistors (BJT) have the best characteristics in terms of reproducibility and well-defined quantities, providing both positive and negative TCs. However, under strong irradiation, some negative aspects show up when using BJTs, which is explained in detail in section 2.5. The core of reference generation is mostly formed by BJTs even if many tasks are also implemented with metal-oxide-semiconductor (MOS) devices, as we will see in the following chapters. [12, 1, 15]

If the gain for PTAT is chosen properly and we assume that the TCs of CTAT and PTAT do not vary over temperature, we get the following expressing for the bandgap voltage U_{BGR}:

$$\frac{\partial U_{CTAT}}{\partial T} + K \cdot \frac{\partial U_{PTAT}}{\partial T} = 0 \tag{5}$$

$$U_{BGR} = U_{CTAT} + K \cdot U_{PTAT} \tag{6}$$

2.4.1 Complementary to Absolute Temperature component

In order to generate a voltage with CTAT characteristics, a forward biased pn-junction is required, as found base-emitter junctions for example in bipolar devices. The voltage U_{BE} at this junction is of CTAT type, where the TC of the voltage depends on the temperature and its magnitude. Again, in this case we will find a negative TC. The collector current of a bipolar device can be expressed [12]:

$$I_C = I_S \cdot e^{\frac{U_{BE}}{U_T}} \tag{7}$$

with:

$$U_T = kT/q \tag{8}$$

The relationship between the saturation current and the temperature can described as:

$$I_S \propto \mu k T N^2 \tag{9}$$

where:

μ	mobility of minority carriers	k	Boltzmann constant
T	Temperature	N	intrinsic carrier concentration of silicon

The components of equation 9 have the following relationships:

$$\mu \propto \mu_0 T^{-\frac{3}{2}} \tag{10}$$

$$N \propto T^3 \cdot e^{-\frac{W_g}{kT}} \tag{11}$$

where:

μ_0	mobility of minority carriers
W_g	Bandgap energy of silicon $\simeq$ 1.12 eV (at 300 K)

It follows

$$I_S = bT^{4+m} e^{\frac{-W_g}{kT}} \tag{12}$$

The variable b is here defined as a proportionality factor. The base-emitter voltage can now be written as $U_{BE} = U_T \cdot ln(I_C/I_S)$, which leads to the TC of U_{BE}. assuming that I_C is held constant, one can now derive U_{BE} from the temperature T.

$$\frac{\partial U_{BE}}{\partial T} = \frac{\partial U_T}{\partial T} ln\left(\frac{I_C}{I_S}\right) - \frac{U_T}{I_S}\frac{\partial I_S}{\partial T} \tag{13}$$

If you now insert equation 12 and shorten the final expression is as follows:

$$\frac{\partial U_{BE}}{\partial T} = \frac{U_{BE} - \frac{5}{2}U_T - \frac{W_g}{q}}{T} \tag{14}$$

From this equation we can see that TC is negative as long as U_{BE} is chosen small enough. In approximation, a linear dependency can be assumed especially in the area of interest (–40 °C to 70 °C).

Starting from this voltage, which is of the CTAT type, a current can now be generated which flows through the resistor *R*, which also has a CTAT characteristic. With sufficiently high amplifier gain, the amplifier inputs are kept at approximately the same level (fig. 8). The inverted

input of the amplifier is connected to the U_{BE} voltage of Q3, which has a negative TC. The current flowing through the resistor R also shows a negative dependence on the temperature. Figure 8 shows the described circuit, where the U_{BE} is shown in green and the CTAT current in red.

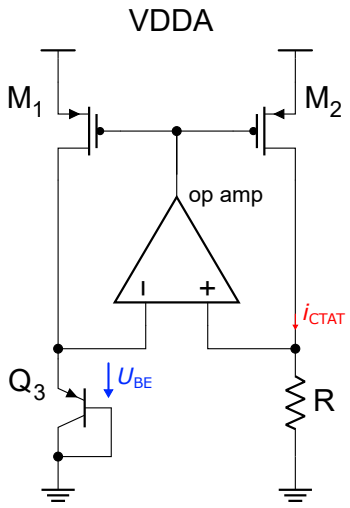


Figure 8: Simplified schematic of a CTAT circuit [11].

The circuit shown here seems simple at first sight, but this is due to the fact that some assumptions were made in the derivation. This was mainly to simplify simulations as already done by Pezzoli [11]. In the real circuit, the amplification and the offset of the operational amplifier (op amp), as well as the parasitic capacities introduced by the manufacturing process, affect the temperature dependence.

2.4.2 Proportional to Absolute Temperature component

The second component of to realize a bandgap reference voltage is the PTAT. To obtain a PTAT voltage, one can operate bipolar transistors with unequal current densities with a ratio $R_j = \frac{j_C}{j_{C0}}$. The difference in current densities is usually done by changing the emitter size of the transistor. The base-emitter voltage of PTAT can be described as [12]:

$$U_{BE} = U_T \cdot ln(R_j) \tag{15}$$

Now we have to introduce a second voltage to get a voltage difference $_{\Delta}U_{BE}$ by changing the current density to N times the density of the first base-emitter voltage .

$$_{\Delta}U_{BE} = U_T \cdot ln(N \cdot R_j) - U_T \cdot ln(R_j) \tag{16}$$

$$_{\Delta}U_{BE} = U_T \cdot ln(N) \tag{17}$$

The dependency on temperature again can be described by deriving $_{\Delta}U_{BE}$ by the temperature and substituting U_T by k/q.

$$\frac{\partial_{\Delta} U_{BE}}{\partial T} = \frac{k}{q} \cdot ln(N) \tag{18}$$

where:

U_T	thermal voltage	N	emitter area ratio
k	Boltzmann constant	T	temperature
q	magnitude of the electrical charge of an electron (~1.602×10^{-19} C)		

As it was proven in equation 18 the differential voltage $_{\Delta}U_{BE}$ between the base-emitter voltages of the two components has a positive TC and is therefore of the PTAT type. From equation 18 we can also see, that the temperature dependency can be set by the ratio of the emitter area N which also includes the current densities.
As mentioned before, in order to generate a PTAT voltage one needs two bipolar devices. This is done by adding a second BJT to the schematic shown in figure 8. The two currents *I* and *I'* have to kept equal, which is done by M_1-M_4 shown in figure 9. As a result the two nodes *A* and *A'* are equal. The U_{BE} voltage of *Q1* equals node *A* and node *B* is equal to U_{BE} of *Q2*. Hence, the voltage difference between the two transistors $_{\Delta}U_{BE}$ equals the voltage drop on the resistor *R*. Since both U_{BE} voltage of *Q1* and *Q2* are of type PTAT, the output voltage $_{\Delta}U_{BE}$ at the resistor is also of type PTAT. [11]

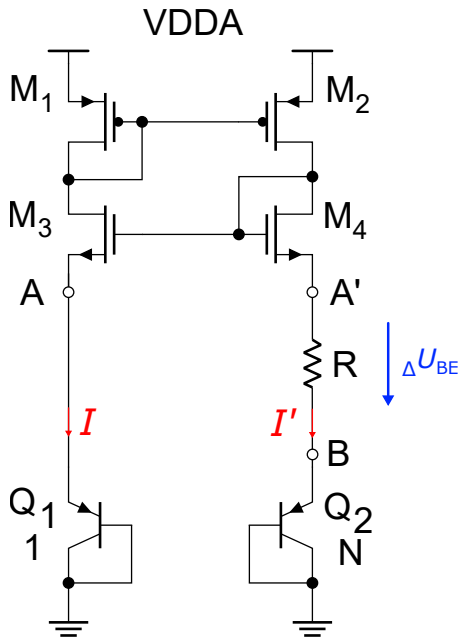


Figure 9: Simplified schematic of a PTAT circuit [11].

2.4.3 Bandgap in current mode

The bandgap reference voltage can be realized in two different designs. On the one hand in voltage mode and on the other hand in current mode. Designing the circuit in voltage mode has the disadvantage that the output voltages of the reference is in the range between 1.2 V to 1.3 V, which in turn leads to a relatively high board power dissipation. Therefore the arcadia committee decided to implement the bandgap voltage in current mode, which leads to the fact that the circuit can provide output voltages in the range of 400 mV to 600 mV. This primarily leads to the fact that the power consumption is drastically reduced. Furthermore, thanks to this circuit, parasitic influences and noise are also minimized. The basic principle of this circuit is still based on the two previously explained voltage types PTAT and CTAT. But instead of using the voltages, as the name suggests, the current flowing through these elements is considered [9].

To realize this circuit we have to merge the two circuits described in section 2.4.1 & 2.4.2 and add few components. The op amp shown in figure 10 again will keep the the two nodes *A* and *A'* roughly at the same potential. Knowing this, the voltage drop on the resistor R_0 is equal to the $_{\Delta}U_{BE}$ of Q_1 and Q_2. Which leads to the fact, that the voltage drop on R_0 is of PTAT-type and so is the current I'_1 flowing through it. Therefore the current I'_1 can be seen as the voltage drop on R_0 divided by its resistance.

Since the amplifier keeps the two nodes *A* and *A'* at the same voltage level and ideally no current flows into it, the voltage at R_1 is equal to the U_{BE} voltage of *Q1* and is therefore of type CTAT. The current I'_2 can therefore be calculated by $\frac{U_{A'}}{R_1}$. Given these relationships *I'* can be written as [11, 12]:

$$I' = \frac{\Delta U_{BE}}{R_0} + \frac{U_{BE,Q1}}{R1} \tag{19}$$

By substituting $_{\Delta}U_{BE}$ from equation 17 we get:

$$I' = \underbrace{\frac{U_T ln(N)}{R_0}}_{PTAT} + \underbrace{\frac{U_{BE,Q1}}{R_1}}_{CTAT} \tag{20}$$

Since U_T and U_{BE} depend on the respective semiconductor material, the temperature dependence of the current can be tuned with the parameters R_0, R_1, and *N*. The two transistors M_1 and M_2 are used to provide the same current for *I* and *I'*. M_3 mirrors the *I'* current which then can be converted into a voltage by adding a resistor R_2. Finally, the bandgap voltage U_{BGR} can be expressed as:

$$U_{BGR} = \left[\frac{U_T ln(N)}{R_0} + \frac{U_{BE,Q1}}{R_1}\right] \cdot R_2 \tag{21}$$

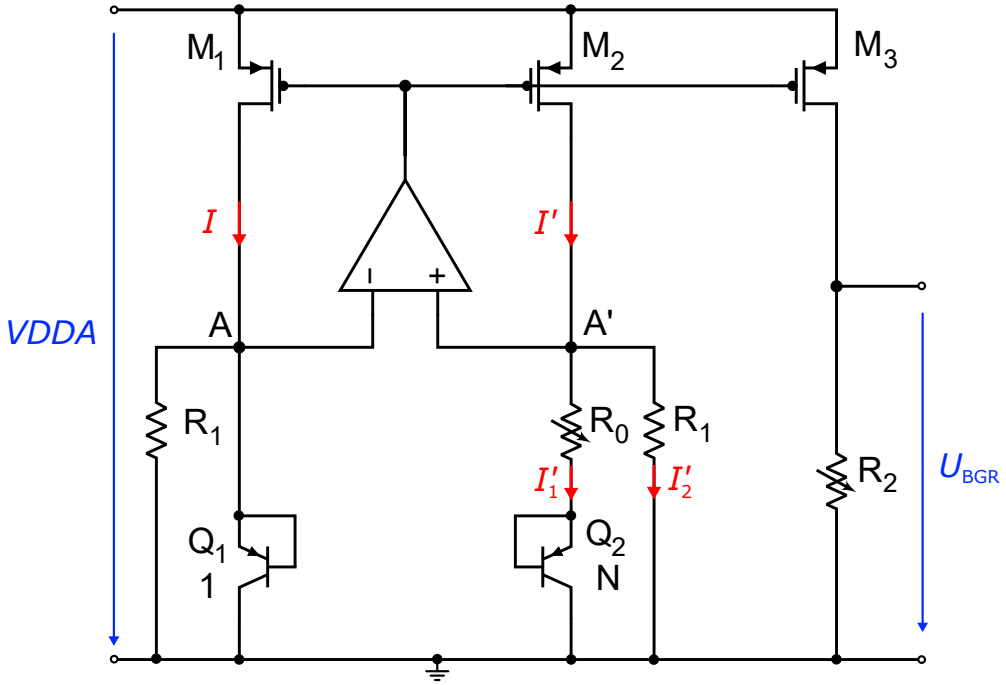


Figure 10: Schematic of the bandgap reference voltage in current mode [11].

For given $U_T ln(N)$ the bandgap voltage U_{BGR} can also be expressed as:

$$U_{BGR} = \frac{W_g}{q} + (4 + m)U_T \tag{22}$$

where:

$$U_T = \frac{kT}{q} \tag{23}$$

From the two equations 22 & 23, it can now be concluded that for T→0 the bandgap voltage U_{BGR} depends only on the bandgap voltage, which is silicon in most of the cases. This voltage is described by the bandgap energy W_g, which was already introduced in chapter 2.1, divided by q, the magnitude of the electric voltage of an electron. From this context and the fact that the *simple* generated bandgap voltage is in the range between 1.2 V to 1.3 V, which is similar to the band energy of silicon which is around 1.22 eV at 0 K, have led to the generated bandgap voltage called **bandgap voltage**. [11, 12]

2.4.4 Radiation damage of semiconductors

When sensors are used in high-energy environments where high levels of radiation are present, this causes damage to the semiconductor which alters the electrical properties. This damage can be attributed to two causes.

The first one is called ***ionization damage***. In order for significant damage to occur in semiconductors, a lot of energy must be supplied. This is achieved in high-energy physic applications

by X-rays, gamma rays and charged particles in the silicon oxide. The supplied energy can release electron-hole pairs in the semiconductor, which can deposit in the lattice defects. Since lattice defects usually migrate to the boundary layer between two materials, the electron-hole pairs also deposit there. Some can recombine, which changes the electrical properties only for a very short time. Others, however, lead to a relatively high charge concentration in this area of the component. The concentration differences create electric fields at the edges, which affect the electrical properties of the component in the long term. This is especially true for surface conduction based devices such as MOSFETs.
The second form is called ***deposition damage***. The penetration of heavy particles into the semiconductor can displace atoms from their bond at the lattice level, resulting in damage to the lattice structure. If this happens often enough, the electrical properties of the semiconductor are altered. Accordingly, these effects are also dependent on irradiation time. These damages are mainly caused by neutrons or protons. They mainly affect BJTs, whose technology is based on volume conduction. [11]

2.5 Bandgap scheme used in arcadia

As already mentioned, the base-emitter voltage of the semiconductor elements plays a decisive role in determining the quality and properties of the bandgap voltage U_{BGR}. Since the environmental conditions in which the arcadia sensor is used are very specific (very strong radiation), the three different semiconductor elements have already been thoroughly studied by Prof. Gianluca Traversi and his team [14]. The tested devices were a pn-diode, a pnp-BJT and a N-MOSFET transistor biased in weak inversion region. They were each used with the aim of voltage generation. To determine which device was best suited for use under very strong irradiation, they fabricated a circuit for each of the three semiconductors using 65-nm technology, then irradiated it up to 250 Mrad and tested it afterwards. In their paper, they concluded that MOSFET devices are best suited for use under very strong irradiation. The maximum deviation of the bandgap voltage after irradiation was just −8.3 mV in a temperature range of −50 °C to 150 °C. The deviations of BJTs and diodes were >40 mV after irradiation.

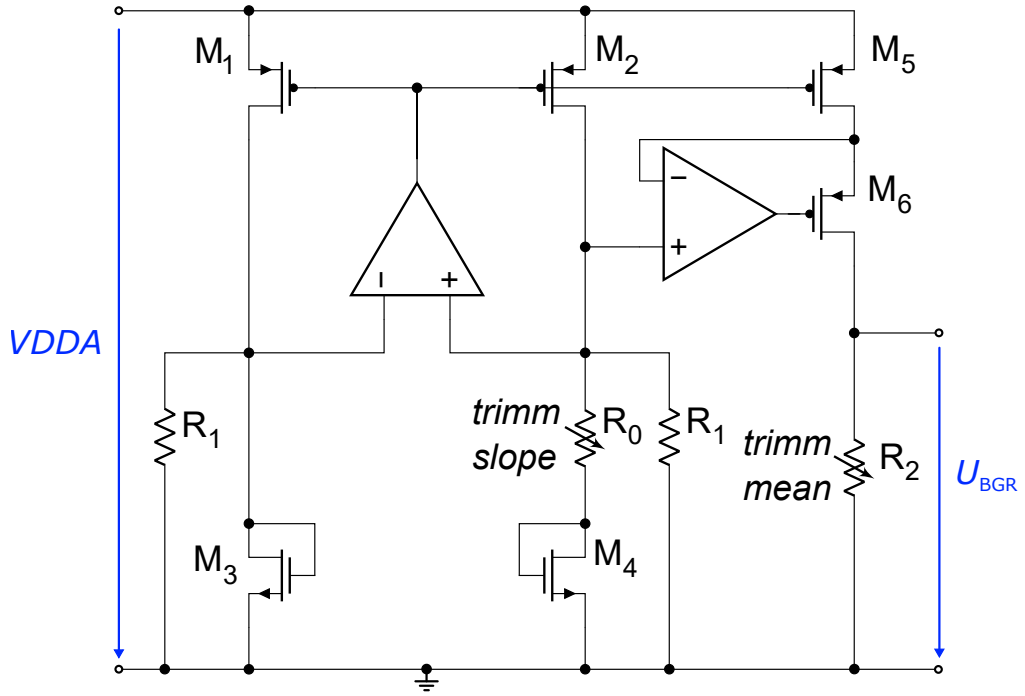


Figure 11: Schematic of the current mode, rad-hard bandgap voltage reference developed for the arcadia project [11].

Compared to the previous circuit from figure 10 only three things have changed to get to schematic designed for the arcadia project, which is shown in figure 11. The first thing that changes are the two MOSFETs M_3 and M_4. As already discussed in the beginning of this chapter they will be used instead of the more commonly used BJTs. The second change is the additional amplifier between M_2 and M_5. It was included to ensure that M_2 and M_5 have the same drain-source voltage, which leads to an increased performance of the current mirror. In addition, the two resistors R_0 and R_2 are programmable in this circuit. From equation 21 it can be seen that by changing R_0, the slope of the bandgap voltage $\partial U_{BGR}/\partial T$ is changed. This therefore affects the **slope** of the CTAT component. Resistor R_2, in turn, can be used to set the absolute value of the bandgap voltage, which will be simply called **mean** in the following. As can be seen in equation 21, R_2 relates to both parts of the equation (CTAT and PTAT) and thus provides a vertical shift of the bandgap voltage across the temperature. The variation of the two 4-bit resistors is realized with binary weighted resistors connected in series. Each of these resistors has a parallel pass-gate through which the resistors can be individually short-circuited, as shown in graph 12. A similar approach has already been studied by Kondo and Tanno [7]. In their paper, they explored an approach with 3-bit resistors and a bandgap voltage reference of 500 mV designed in 600 nm standard CMOS process. They could successfully trim the BGR, but concluded that the bit-width must be expanded for more accurate results. In case of arcadia the circuit was designed for a BGR voltage of 600 mV and it was decided to use 4-bit resistors to extend fine adjustment. By using 4-bit resistor [0:15] the operating point must be placed in

the middle of the range. This means that the digital value of R_2, which is responsible for the mean value should show an output of around 600 mV by setting it to the decimal value of 7 [0100]. By setting R_0 to 7 the U_{BGR}/T in theory the curve should have a slope of almost 0.

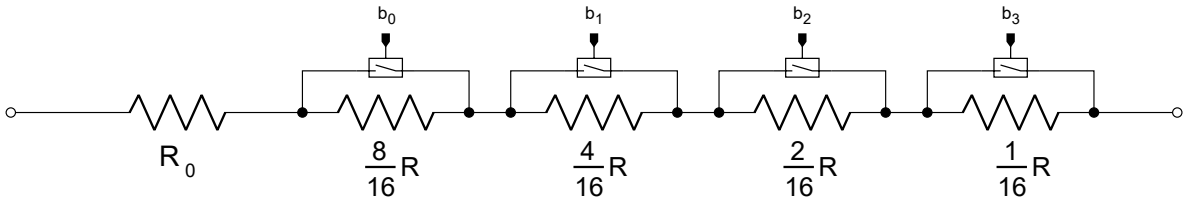


Figure 12: Schematic of the 4-bit resistors [11].

The circuit shown in figure 12 works as follows in the case of R_0, i.e. the slope change. If the resistor is set to decimal 0 [0000], all 4-bit resistors are bridged and only R_0 contributes to the voltage change. For this value the slope reaches the highest positive value. If the resistor R_0 is set to decimal 15 [1111], all 4 pass-gates are open and the current must pass through all resistors. This leads to the fact that the voltage drop at R_0 becomes maximum and therefore the slope reaches the maximum negative value.
The same applies to the setting of the resistor R_2. As can be seen from equation 21, by increasing the resistor, the output voltage U_{BGR} can be increased. If R_2 is set to decimal 15 [1111], the maximum output voltage U_{BGR} is obtained and vice versa the minimum output voltage for decimal 0 [0000]. Here it should be mentioned that the exact resistor values of R_0 and R_2 are unknown. This circuit is implemented in each of the 16 test points of the board. [11, 8]

2.5.1 Startup circuit

Another problem with the bandgap reference circuit is that it has two stable operating points. When the input voltage VDDA is switched from 0 V to 1.2 V, the circuit stabilizes in *zero current mode* and therefore is not able to provide the output voltage U_{BGR}. This is due to the fact that both inputs of the op amp are grounded, as well as the output of the amplifier and also U_{BGR}. To escape this obviously unfavorable operating point a **startup circuit** is used. The idea behind the startup circuit is to reduce the output voltage at the amplifier, which causes a current to flow through the two transistors M_1 and M_2, forcing the circuit into the second stable operating point. When the reference circuit is in this second state, the startup circuit should be switched off. In the arcadia project, a dynamic startup circuit was used, which is shown in figure 13. The requirements for the startup circuit can be summarized as:

- must be able to quickly generate a bias current to start the bandgap core circuit
- after the core circuit of the bandgap enters the normal operating state, the start-up circuit can be automatically turned off

- should not affect the normal operation of the circuit
- provide bias current for the op amp

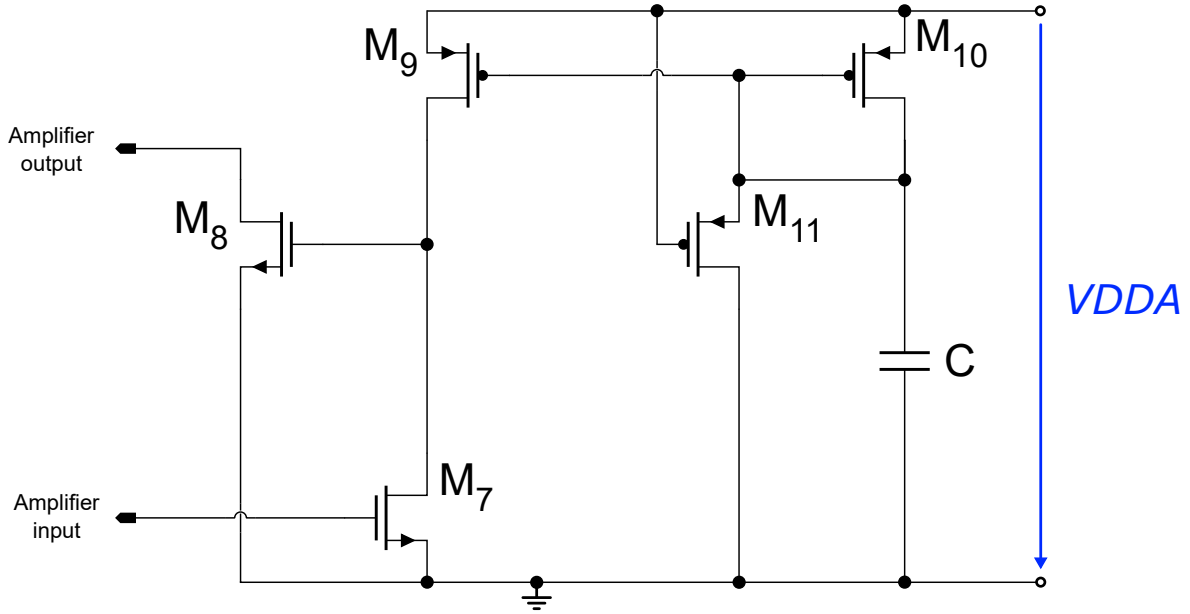


Figure 13: Schematic of the dynamic startup circuit [11].

The capacitor C is used to generate the time constant for switching off the startup circuit. It can be derived by $\tau = R \cdot C$. Initially the capacitor is discharged and the input voltage VDDA rises from 0 V to the nominal voltage 1.2 V, causing a current to flow into the capacitor. The two transistors M_9 and M_{10} serve to mirror the charging current of the capacitor. At the beginning the bandgap core is in *zero current mode*, which leads to the gate of M_7 being at ground. The drain current of M_9 causes the gate voltage at M_8 to increase, which decreases the output voltage of the amplifier. The increasing currents in the bandgap core lead to increasing voltages at the amplifier inputs. This causes the gate source voltage at M_7 to increase. As soon as the drain-source transition (DS) of M_7 becomes conductive, the gate of M_8 is at ground and the amplifier output is no longer affected. When the capacitor is fully charged, no more current flows into the current mirror and the power consumption of the circuit approaches zero. As soon as VDDA is switched off, the capacitor is discharged by M_{11} in order to bring it back in its initial state when switched on again. [6, 11, 10]

3 Aims of the measurements

To achieve automated calibration of the BGRs, their behavior must be known as precisely as possible. To do so, different goals are pursued. The goals can be described and calculated mathematically. Often irregularities occur in the process. In this chapter I want to explain how the behavior can be described mathematically with the help of the recorded data and which limitations and problems occur and their statistical significance. I would like to achieve this with the points listed as follows:

- **Quality of trimming**
- **Linearity of the system**
- **Cross dependencies in the system**
- **Dependence between temperature and BGR voltages**
- **Balance between mean and TC**
- **Specimen scattering**
- **Circuit improvement**

3.1 Quality of trimming

The mode of action of the two resistors has already been described in detail in theory. Now, however, it must be demonstrated on the basis of the data how the change of the resistance affects the change of the output voltage (slope & mean). A comparison can be made with simulations already performed by Pezzoli [11]. The criteria to be evaluated are the change of the slope per bit and the resulting slope change of the BGR voltage per bit. The same can be determined for the mean settings. The criterion here is how much the mean voltage changes per bit. This analysis is intended to explore the step sizes in which the output voltages can be trimmed.
Furthermore, the output voltage at a bit configuration of 7 [0111] is to be considered, since this value must already be taken into account in the board design. It should be about 600 mV. In addition, the range is to be examined, i.e. how far the voltage U_{BGR} can be varied upwards and downwards from the design point. It can also be investigated if there are any temperature dependencies introduced by the resistors.

3.2 Linearity of the BGR voltages

There are several ways to find and classify linearities of a system. In this chapter I would like to introduce the two methods that are relevant for the problem at hand. These are **linear regression** and the **temperature coefficient**.

3.2.1 Linear regression

The dependence of the output voltage with temperature can be expressed with the help of the temperature coefficient TC. However, this is based on the existence of a linear model. To confirm this, it must first be investigated whether and how the measurement data can be linearized. One way, to fit linear model parameters, is to form a linear single regression (see e.g. [13]). If the error between the measurement and the approximated straight line is small enough, the system can be described linearly. By approximating the measured values with a regression straight line, a very small mean error is also obtained. The linearity of the system forms the basis of all subsequent statistical evaluations and is therefore of particular importance. The regression line is calculated as follows [13]:

$$y = a + b \cdot x \tag{24}$$

with:

$$b = \frac{\sum_{i=1}^{n}(x_i - \overline{x}) \cdot (y_i - \overline{y})}{\sum_{i=1}^{n}(x_i - \overline{x})^2} \tag{25}$$

and

$$a = \overline{y} - b \cdot \overline{x} \tag{26}$$

Where $\overline{x}$ and $\overline{y}$ correspond to the mean values of the measurement dates x and y. How well the system can be linearized can be described by the *y-error*. The *y-error* results from the difference between the measured value and the value of the regression line for the same temperature. Thus, for small *y-errors*, the system can be considered linear. However, if the *y-error* increases, the system can no longer be described sufficiently accurately by a regression model. This leads to nonlinearities, which have to be described by nonlinear methods. These nonlinearities mainly affect the calculation of the temperature coefficient, as described in detail in section 5.3.1. How high the y-error may be in general is not fixed and must be determined with the help of the recorded data.

3.2.2 The temperature coefficient

The relative change of a physical property can be expressed in parts-per-million. In case of change over temperature this can expressed as temperature coefficient. To calculate the TC, the total change of the output voltage (U_{BGR}*(max)*-U_{BGR}*(min)*), the output voltage U_{BGR} at app. 30 °C room temperature and the total temperature range are needed. The unit of the relative

change over temperature can be expressed as:

$$TC = \left(\frac{U_{BGR,max} - U_{BGR,min}}{\frac{U_{BGR,30°C}}{110°C}} \right) \cdot 10^6 \qquad [\mathrm{ppm/°C}] \tag{27}$$

As can be seen from the equation for TC, the values for $U_{BGR,max}$ and $U_{BGR,min}$ are decisive for the result. For linear systems the temperature coefficient can be a good description. For nonlinear ones, however, there are some difficulties that should be taken into account when calculating the TC. [5]

I would like to illustrate this by means of a calculation example. For this purpose, the TC was determined for TP8, for a configuration where the BGR voltage behaves nonlinear over temperature. First on the basis of the measured data and secondly on the basis of the linearized data. For TP8, the configuration **slope 14 mean 15** resulted in a relatively large percentage *y-error* of approx. 1.2 %. In the following equations, the TC for TP8 was determined for the same configuration. I want to show why it is important to choose the right $U_{BGR,min}$ and $U_{BGR,max}$. TC_{meas} was determined here with the measured values where $U_{BGR,min}$ and $U_{BGR,max}$ were chosen by checking for maximum and minimum values. For calculating $TC_{meas,40\text{-}70}$ only the BGR values for –40 °C and 70 °C were taken into account. TC_{lin} on the other hand was determined with the help of the regression line. The following calculations show the different results a nonlinear system can lead to:

$$TC_{meas} = \left(\frac{666.23\,\mathrm{mV} - 648.25\,\mathrm{mV}}{\frac{665.03\,\mathrm{mV}}{110°C}} \right) \cdot 10^6 = 245.8\,\mathrm{ppm/°C} \tag{28}$$

$$TC_{meas,-4070} = \left(\frac{661.48\,\mathrm{mV} - 648.49\,\mathrm{mV}}{\frac{665.03\,\mathrm{mV}}{110°C}} \right) \cdot 10^6 = 180.8\,\mathrm{ppm/°C} \tag{29}$$

$$TC_{lin} = \left(\frac{666.77\,\mathrm{mV} - 662.75\,\mathrm{mV}}{\frac{655.90\,\mathrm{mV}}{110°C}} \right) \cdot 10^6 = 55.8\,\mathrm{ppm/°C} \tag{30}$$

As one can see here, the three results are very far apart, although they describe the same curve. Whereby one can note here that the calculation of TC_{meas} describes the actual behavior probably in the worst way, since here the wrong maximum value is used. For the calculation of TC, therefore, the boundary values should always be preferred to the extreme values.

In addition, it can be concluded that for this state, no meaningful statement can be made about the TC, since the system can no longer be sufficiently linearly approximated. Thus, an upper limit for linearity must be set. This is defined as already mentioned with the help of *y-error*.

For the evaluation in this work, however, these nonlinearities can be neglected, since they only occur for very high mean values, which are exclusively located sufficiently far away from the operating point. This can also be seen in figure 14 by looking at the minimum BGR voltages.

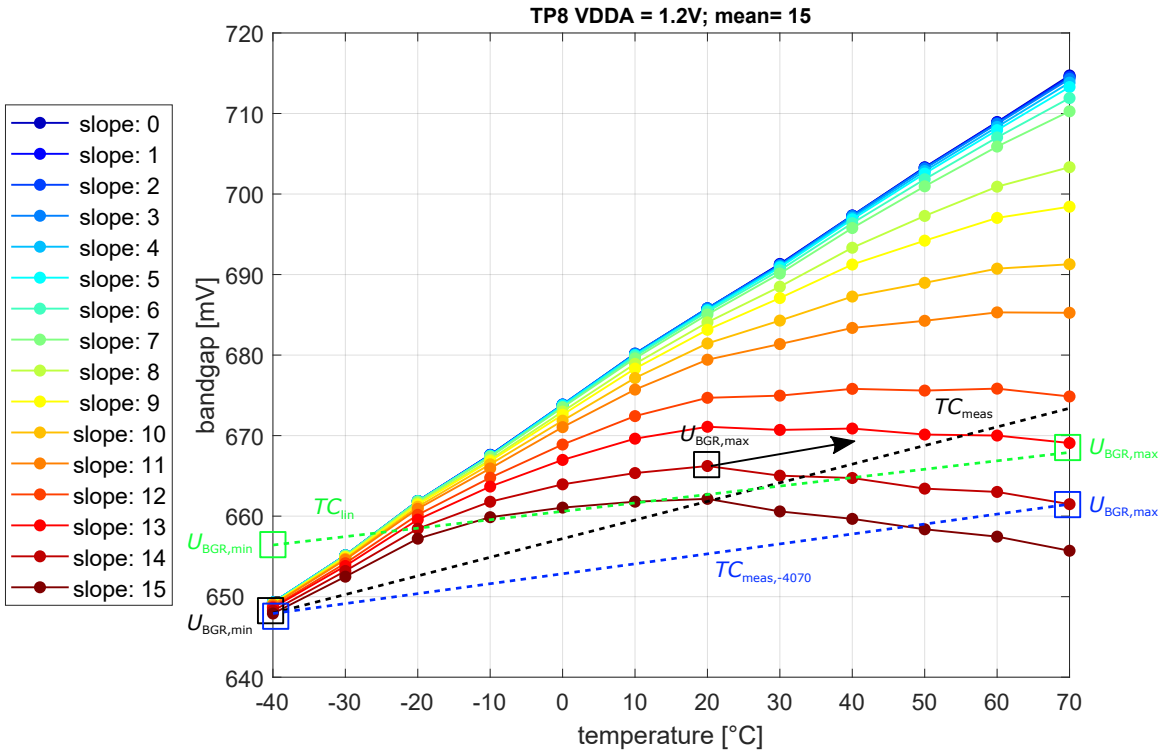


Figure 14: Measured data points of TP8 at mean setting 15 and slope setting of 14.
The dashed black line shows the quantitative results of TC calculated by TC_{meas}.
The dashed blue line shows the quantitative results of TC calculated by $TC_{meas,-4070}$.
The dashed green line shows the quantitative results of TC calculated by TC_{lin}.

3.3 Dependency between mean and slope setting

As already mentioned in section 3.1, the BGR voltage can be tuned by the two different resistors mean & slope. In a perfect system, the two parameters can be set completely independently of each other. If this was the case, it would greatly simplify the configuration search. A possible interdependence can be concluded from diagrams of the measurements. But a mathematical criterion can help in order to give an objective assessment.
There are several ways to answer this question. A relatively simple mathematical analysis can be performed using the *correlation coefficient* between the parameters (see e.g. [13]). This is a good choice if the are not too many parameters to take into account. However, the basis for this is that the correlation between the parameters is approximately linear. Another criterion for the *correlation coefficient* is that data are interval scaled. This also applies to these measurements, since the data were recorded in increments of the resistors. The *correlation coefficient* r can be mathematically described with the following equation [13]:

$$\boxed{r = \frac{\sigma_{m,s}}{\sigma_m \cdot \sigma_s}} \tag{31}$$

with:

$$\sigma_m = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n} \cdot (m_i - \bar{m})^2} \tag{32}$$

$$\sigma_s = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n} \cdot (s_i - \bar{s})^2} \tag{33}$$

$$\sigma_{m,s} = \frac{1}{n-1}\sum_{i=1}^{n} (m_i - \bar{m}) \cdot (s_i - \bar{s}) \tag{34}$$

Where $\sigma_{m,s}$ is the covariance of mean and slope divided by the product of the variance of both parameters. The coefficient r can take values $-1 \leq r \leq 1$.

- For $r = -1$ there is a maximum reciprocal correlation between the two characteristics. This means that for increasing x-values the y-value decreases.
- For $r = 0$ there is no linear relationship between the two characteristics, but this does not rule out the existence of nonlinear relationships.
- For $r = 1$ both characteristics tend to increase. This is called absolute positive correlation

Whether and to what extent the two characteristics are related to each other can thus be expressed using a single parameter.

3.4 Balance between mean and TC

If we analyze the data by first determining the configuration for the lowest TC and then looking at the output voltages, we will find that they scatter relatively randomly around the design point. The goal, however, is to find a configuration with both a reasonable TC and a small deviation from the output voltage of 600 mV. This can be ensured e.g. by a predefined weighting of the two parameters. In addition, tolerances can be defined for the two parameters, based on the extreme values, which must not be exceeded.
In addition, there are various methods for finding this configuration. One method, for example, is to first find the configuration with the lowest temperature coefficient. This evaluation provides **one** slope and **one** mean value per TP. Now one can fix the mean value and you get **16** corresponding slope values **per TP**. The one with the smallest deviation from the design point can then be determined, which leads to a new configuration.
The same can be done the other way round. If one fixes the slope value after the TC analysis and then look at all the associated mean values, you can also determine those that have the smallest deviation from the design point. The two methods must then be compared and weighed against each other.

3.5 Dependency of the output voltage to the input voltage

In the testboard measured here, there is an LDO that stabilizes the input voltage. However, this was switched off for measurement purposes. In the later demonstrator, there will no longer be a LDO that stabilizes the voltage. Deviations in the input voltage may therefore occur. In this experiment, measurements of ±10 % of the input voltage were performed. With the help of this data, limit values of the input voltage are to be determined in order to guarantee that the bandgap voltages provide stable output values.
Since all measurements for the characterization were also made for ±10 % of the input voltage, the trim quality can be investigated as a function of the input voltage. Line regulation is used to study the dependence on the input voltage from the power supply. The line regulation is the ability of a power supply to keep an output voltage constant despite changes of the input voltage. In this case the BGR circuit will be considered as the power supply, the output voltage will be U_{BGR} and the input voltage VDDA. The mathematically definition of the line regulation is [11]:

$$LR = \frac{{}_{\Delta}U_{BGR}}{U_{BGR,30°C} \cdot {}_{\Delta}VDDA} \cdot 100\% \qquad [\%/\mathrm{V}] \tag{35}$$

where:

${}_{\Delta}U_{BGR}$	difference between maximum and minimum U_{BGR}
$U_{BGR,30°C}$	U_{BGR} at 30 °C
${}_{\Delta}VDDA$	difference between maximum and minimum input voltage

Line regulation - once more - is implying a linear model. It will be necessary to investigate what limits can be defined for the linear model.

3.6 Specimen scattering

Due to production deviations, there are differences in the individual samples. The aim here is to investigate whether there are trends that can be attributed to production. An important parameter which is already determined during the design is the ratio R_j, which determines the ratio of the emitter sizes of the transistors. This parameter determines the PTAT ratio and thus influences the temperature behavior of the BGR voltage. Furthermore, it has to be investigated to what extent the respective test points differ in their temperature and trimming behavior. And whether statements about production influences and expected deviations can be made on the basis of these measurement data.

3.7 Circuit improvement

In the last step, I would like to use the recorded data to make suggestions for improving the circuit. The aim is to investigate where the trimming reaches its limit. An important factor of the circuit is whether the output voltage for a set decimal resistor value of 7 really lies in the middle. Can the voltage be trimmed equally well upwards and downwards? The same applies to the slope of the voltage. If the middle value was set incorrectly, the outer values of the 4-bit resistors may have to be used to find an acceptable configuration when trimming. With the help of this analysis, an evaluation of the two resistors R_0 and R_2 should take place.

4 Measurement setup

In this chapter I will describe in detail how the measurements were performed. But before I go into more detail about the measurements themselves, I will talk about the layout of the arcadia. Figure 15 shows a section of the power part of the testboard, whose complete schematic can be found in appendix E. From this figure one can see that the analog supply voltage **VDDA** is fed directly to a low-dropout regulator, short LDO, which provides 1.2 V at the output. However, since in these measurements also fluctuations of the input voltages are examined, the voltage provided by the external power supply is fed directly via TP19 and the LDO is disabled via R55. The two digital input voltages **VDD_IN** and **VDD1_IN** are supplied via the two jumpers J13 and J15 as shown in the diagram in appendix E. In order to characterize the bandgap voltages of the testboard, there are two extensive measurements that need to be performed. In the first measurement the input voltage VDDA will be varied from 0 V to 1.32 V and contradictory from 1.32 V to 0 V in steps of 10 mV. This measurement will be called V_{in}/V_{out} in the following. With the help of these measurements, information about the behavior of the start up circuit can be obtained.
In the second measurement the bandgap voltage for every TP will be measured for every possible configuration of the two 4-bit resistors in a temperature range between –40 °C to 70 °C. These measurements will be called *characterization measurements* and will help to find the best configuration for each TP.

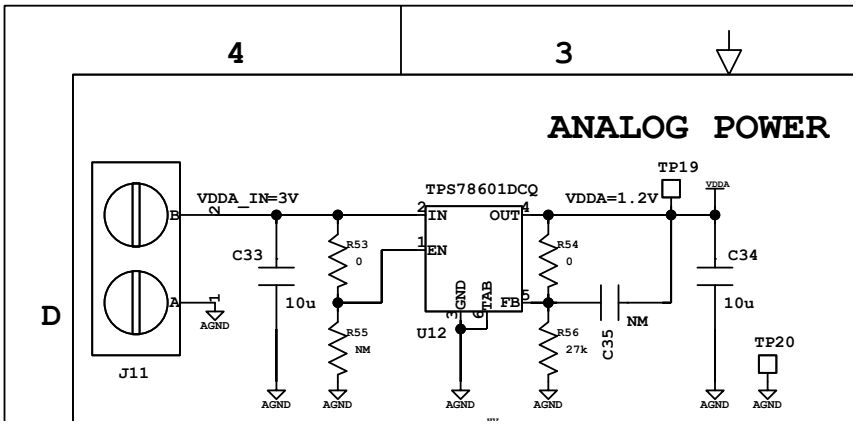


Figure 15: Extraction of the schematic of the arcadia testboard.

The measurement setup is kept the same for both measurements. The only thing that changes is the program, that perform all the measurements. It consists of two DC power supplies and a multimeter which are connected to the PC via USB cables. The first power supply, Agilent E3631A, is used to provide the three digital input voltages *VCC_IN*, *VDD_IN* and *VDD1_IN*.

It is not programmable and the voltage and compliance have to be set manually before taking measurements. The second power supply, Agilent B2961A, is programmable and provides the analog input voltage of *VDDA*. It can be set and read from the measurement program. The bandgap voltage references were measured by the digital multimeter Keysight 34461A illustrated in figure 16 in green. Due to the fact that the testboard had to stay inside the climate chamber during the measurements all cables that were connected to it had to be longer than 1.5 m.

Since the operating voltage of *VCC_IN* at 4 V causes a relatively large current and the board is thus heated, the minimum input voltage of the LDO was determined in a test run. Therefore the input of *VCC_IN* is operated with 2.5 V instead of 4 V. The Field-programmable gate array (FPGA) was used to control the board. The firmware was uploaded to the board with the software vivado using the FireFly™ cables between the breakout board and the arcadia testboard. Those have also determined the upper limit for the temperature value of 70 °C, as they are not suitable for operation above this temperature.

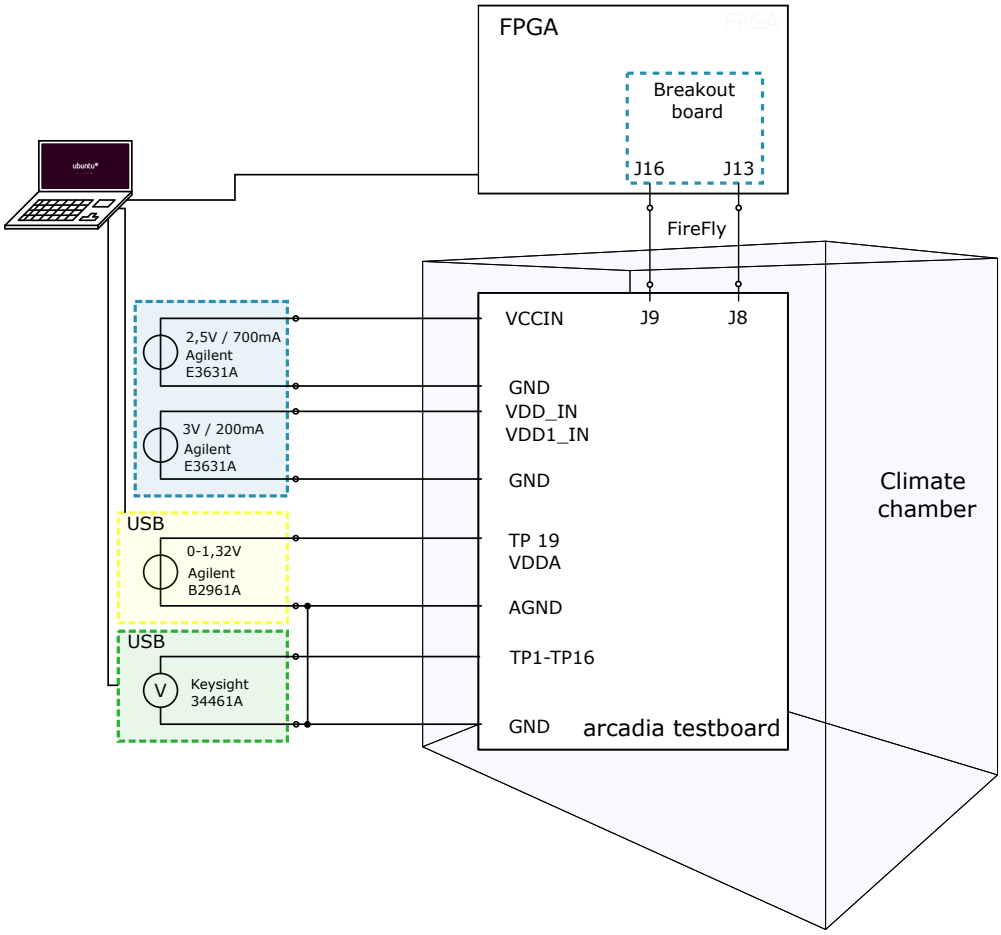


Figure 16: Schematic of the measurement setup [self made drawing].

4.1 Measurement programs

To perform the V_{in}/V_{out} measurements the input voltage has to be in a range comprised between 0 V and 1.32 V. This is due to the fact that the circuit should be able to perform properly in a range of ±10 % of the nominal input voltage which is 1.2 V. The step size was chosen of 10 mV which is sufficiently small for this sort of measurement. With the help of these measurements it was possible to investigate as of the BGR no longer shows any changes with respect to the changes in the input voltage. These measurements were performed for every TP and for the temperatures –40 °C, 30 °C and 70 °C. Because of the startup circuit different results were expected from ranging the input voltage from 0 V up or from 1.32 V down to 0 V. An extract of this measurement program is shown in code 1. In lines 2-3 the parameters for TP are defined, while the temperature had to be set manually. Since the multimeter has only one input for voltage measurement, every TP has to be connected manually after concluding the measurements on the one before it. Lines 8-10 are used pass IP addresses of the respective devices to the variables. In lines 19 and 20 the output voltage and compliance of the power supply will be set. After this step a sleep time of 0.5 s was included before reading the BGR output voltages. To make sure that the input voltage was set to correctly its value was read back from the power supply. In lines 27-32 a matrix was created to write to print and save the results in a csv-file. The full program can be seen in appendix C.

```
1 ##########################################
2 TP = 1      # testpoint
3 tmp = 30   # temperature
4 ##########################################
5
6 plt.close("all")
7 rm = pyvisa.ResourceManager('@py')
8 ps_agilent = rm.open_resource('USB0::2391::36632::MY52350174::0::INSTR')
9 agilent = Agilent34461A(rm)
10
11 header = ['set voltage', 'bandgap voltage','Temperature']
12
13 # define voltage range and steps
14 volt_in = np.arange(0,1.33,0.01)
15 #volt_in = np.arange(1.32,-0.01, -0.01)
16 results = pd.DataFrame()
17
18 for voltage in volt_in:
19   ps_agilent.write('VOLT %1.2f' % voltage)
20   ps_agilent.write('SENS:CURR:PROT 2') #compliance in A
21   time.sleep(0.5)
22   print('Vin: %1.3f' % voltage)
23   vin_real = float(ps_agilent.query('MEAS:VOLT:DC?'))
24   print('Vin real: %1.3f' % vin_real)
25   voltage_dec = decimal.Decimal(voltage)
```

```
26
27   temp = {}
28   temp['Vin'] = float(voltage)
29   temp['Vin_real'] = vin_real
30   temp['Vout'] = agilent.get_voltage_rang_1()
31   temp['temp'] = tmp
32   temp['VCC']  = VCC
33   results = results.append(temp, ignore_index=True)
34
35 if tmp < 0:
36     temper = tmp*-1
37     results.to_csv('/home/microlab/Documenti/Python_scripts/Band_gap/VIN_VOUT/
38 ␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣Results_TP'+str(TP)+'_TEMP_m'+str(temper)+'_0_UP.csv')
39 else:
40     results.to_csv('/home/microlab/Documenti/Python_scripts/Band_gap/VIN_VOUT/
41 ␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣Results_TP'+str(TP)+'_TEMP_'+str(tmp)+'_0_UP.csv')
```

Code 1: Extract of the measurement program Vin/Vout

The second program which had to be developed was a bit more complex. As a requirement for the script, the following parameters had to be varied:

- input voltage for ±10 % range (1.08 V; 1.2 V; 1.32 V)
- mean-register (0-15 in decimal numbers)
- slope-register (0-15 in decimal numbers)

To realize this, three *for-loops* were used for each variation. The first one changes the input voltage, the second one the slope-register and the third one the mean-register. This leads to 769 measure points per TP and per temperature. During this measurements a total amount of 147.648 measure points were collected. In code 2 an extract of the program is shown. In lines 5 and 9 the aforementioned ranges are defined. Once the input voltage is set, the second loop starts from slope-register 0 and enters the third loop, where it will change the mean-registers from 0-15. After accomplishing this the slope-register will increase and so on. The results of these measurements are written to a csv-file, where to columns hold the following information:

- test point TP
- temperature
- input voltage set
- input voltage measured by the power supply
- slope value
- mean value
- bandgap voltage

```
1
2 ####################################
3 TP = 1      # set testpoint
4 tmp = 70    # set temperature
5 ####################################
6
7 # define voltage range and steps
8 word_bits = np.arange(0,16,1)
9 mean = np.arange(0,16,1)
10 #slope = np.arange(15,-1,-1)
11 slope = np.arange(0,16,1)
12 ps_volts = np.array([1.08, 1.2, 1.32])
13 i = np.arange(0, 3, 1)
14
15 results = pd.DataFrame()
16
17 register = TP -1
18 register_mean = 'BIAS'+str(register)+'_BGR_MEAN'
19 register_slope = 'BIAS'+str(register)+'_BGR_SLOPE'
20
21 if tmp < 0:
22     temper = tmp*-1
23     filename =
          '/home/microlab/Documenti/Python_scripts/Band_gap/final_characterization/
24 ␣␣␣␣Results_TP'+str(TP)+'_REG_m'+str(temper)+'.csv'
25 else:
26     filename =
          '/home/microlab/Documenti/Python_scripts/Band_gap/final_characterization/
27 ␣␣␣␣Results_TP'+str(TP)+'_REG_'+str(tmp)+'.csv'
28
29 for n in i:
30     PS_VOLT = ps_volts[n]
31     ps_agilent.write('VOLT␣%1.2f' % PS_VOLT)
32     for slopes in slope:
33         # set slope value
34         x.chip.write_gcrpar(register_slope, slopes)
35         slope0 = x.chip.read_gcrpar(register_slope)
36         print('slope:␣'+str(slope0)+'')
37         for word in mean:
38             # set mean value
39             x.chip.write_gcrpar(register_mean, word)
40             mean0 = x.chip.read_gcrpar(register_mean)
41             print('slope:', slope0,'mean:', mean0)
42             time.sleep(0.01)
43             #readback voltage of PS
44             vin_real = float(ps_agilent.query('MEAS:VOLT:DC?'))
45             Iin_real = float(ps_agilent.query('MEAS:CURR:DC?'))
46             print('PS␣Iout␣measure:␣%1.4f' %Iin_real)
47             print('PS␣Vout␣set␣to:␣%1.4f' %PS_VOLT)
```

```
48              print('Vin real: %1.4f' % vin_real)
49              #read voltage of TP
50              volt = agilent.get_voltage_rang_1()
51              temp = {}
52              temp['TP'] = TP
53              temp['reg'] = register
54              temp['temp'] = tmp
55              temp['Vin_real'] = vin_real
56              temp['Vin'] = PS_VOLT
57              temp['SLOPE'] = slope0
58              temp['MEAN'] = mean0
59              temp['Volt'] = volt
60              results = results.append(temp, ignore_index=True)
61
62 results.to_csv(filename)
63 plt.plot(results['SLOPE'],results['Volt'], 'b.')
64 plt.xlabel('Slope')
65 plt.ylabel('bandgap [V]')
66 plt.title('Bandgap measurement of TP'+str(TP)+' at '+str(tmp)+'C')
67 plt.grid(visible=True, which='major', color='grey', linestyle='--')
68 ps_agilent.write('SYST:BEEP 1000,0.7')
69 plt.show()
```

Code 2: Extract of the measurement program characterization

4.2 Temperature measurement

As already mentioned, the temperature was varied using a climatic chamber. This temperature was increased from –40 °C to 70 °C in steps of 10 °C. For each temperature, all configurations for each TP were measured and then the temperature was increased. When changing from one temperature to the next higher, I first waited until the target temperature was reached. A timer was then set to 10 min to ensure that all elements of the board were reasonably at that temperature. The ambient temperature inside the climatic chamber was measured using the temperature sensor installed in the chamber. This provided sufficiently accurate results. The fluctuations during the measurements were at most in the range of ±0.1 °C. The temperature was then entered manually in the measuring program as one can see in code 2 at line 4.
A measurement of the board temperature was carried out in advance. The chip temperature was measured with a thermal imaging camera. This measurement resulted in a temperature of 53 °C at 26 °C ambient temperature with a *VCC_IN* voltage of 4 V. Because of this high board temperature the *VCC_IN* voltage was reduced to 2.5 V. This was done in a test by reducing the supply voltage as much as possible, so the LDO would still work properly. All of the components affected by this change were part of the digital power supply (fig. 57) and therefore do not affect the BGR voltages. This improvement brought the chip temperature below 40 °C for its maximum at 26 °C ambient temperature, because the current flow was drastically reduced.

5 Results

In order to evaluate the collected data, the numeric computing software Matlab© was used, since this software allows the user to perform calculations as well as to represent data in a very customizable way. As mentioned earlier, approximately 150,000 data points were collected during the measurements. This data also includes different input voltages (1.08 V, 1.2 V and 1.32 V). For the evaluation of the *characterization measurements*, only the data for 1.2 V are considered, since this is the design point. In order to determine the relationship between input and output voltage, the data obtained from the V_{in}/V_{out} *measurements* are used. In this part, I would like to present the results of the data analysis with reference to the problems mentioned in chapter 3. I will also demonstrate why each method was applied and what conclusions can be drawn from the results.

5.1 Raw data

In the first step I would like to present the raw data, because some effects are already recognizable. In figure 17 all measurement points for TP1 for an input voltage of 1.2 V are presented in a 3D plot. However, there are actually four dimensions shown in this plot. The 1st one shows the slope values, the 2nd the mean values, the 3rd the BGR voltages and the 4th the temperatures - represented by the height of the „bars“. The light blue layer represents the operating point.
From this plot one can already see, that the span of output change over temperature is highest for the configuration slope 0 and mean 15. One can also see that there are nonlinearities in the represented system, since the data propagate in form of curved traces. This graph also provides a good description of the main goal, which is to find a configuration in which the bar height is the smallest and the distribution of the bar is as close as possible to the blue plane.
Furthermore, one can already see from this graph that the values are well distributed around the operating point at first glance. Qualitatively, no precise statement can be made yet, but one can see that the plane intersects the measured values approximately in the middle.
Patterns can also be seen when looking at the TC values in figure 18. The amplitudes of the TC extreme values in the positive range are significantly higher than in the negative range. In addition, the dependence of the TC on the mean settings increases for high mean and low slope values. This is also the range in which the linearity increases. Looking at the two diagrams, the difficulty of the task also becomes clear. A configuration must be found whose properties are as close as possible to the ideal ones presented by the planes.

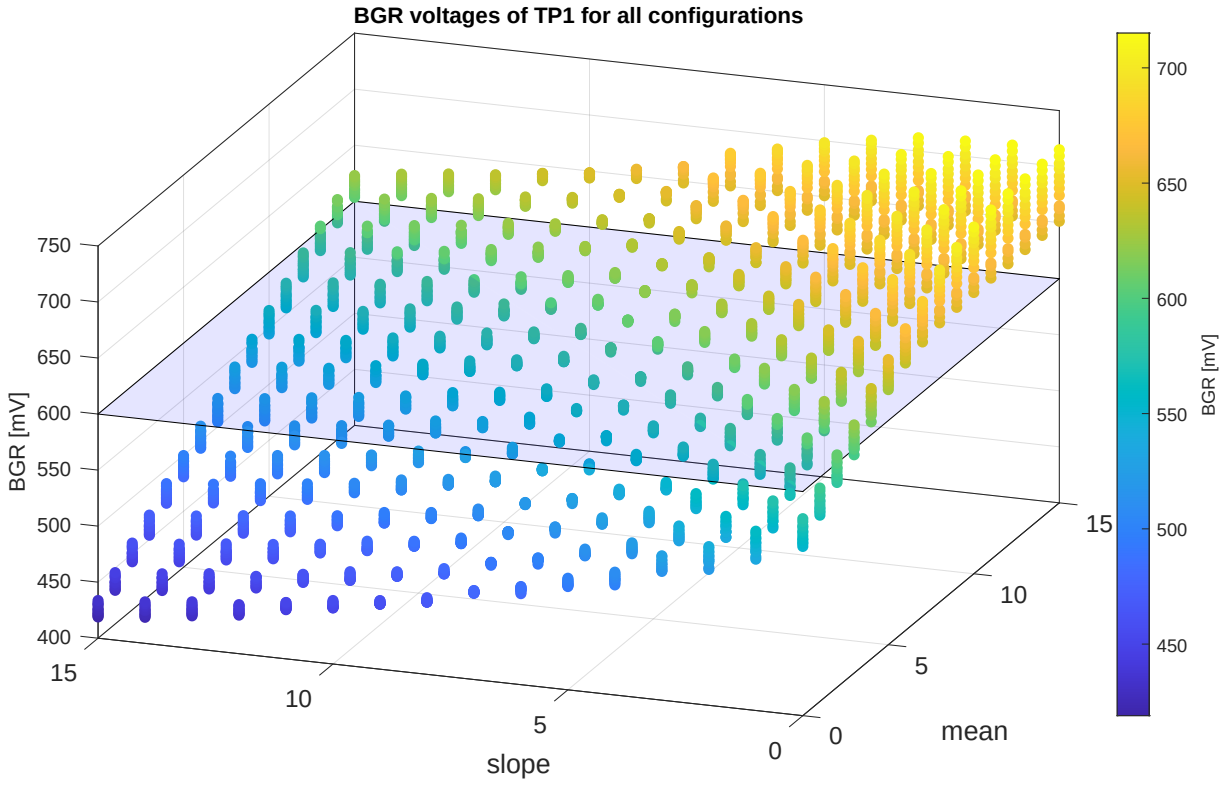


Figure 17: 3D plot of TP1 for all configurations and BGR voltages. The blue plane represents the design point of 600 mV.

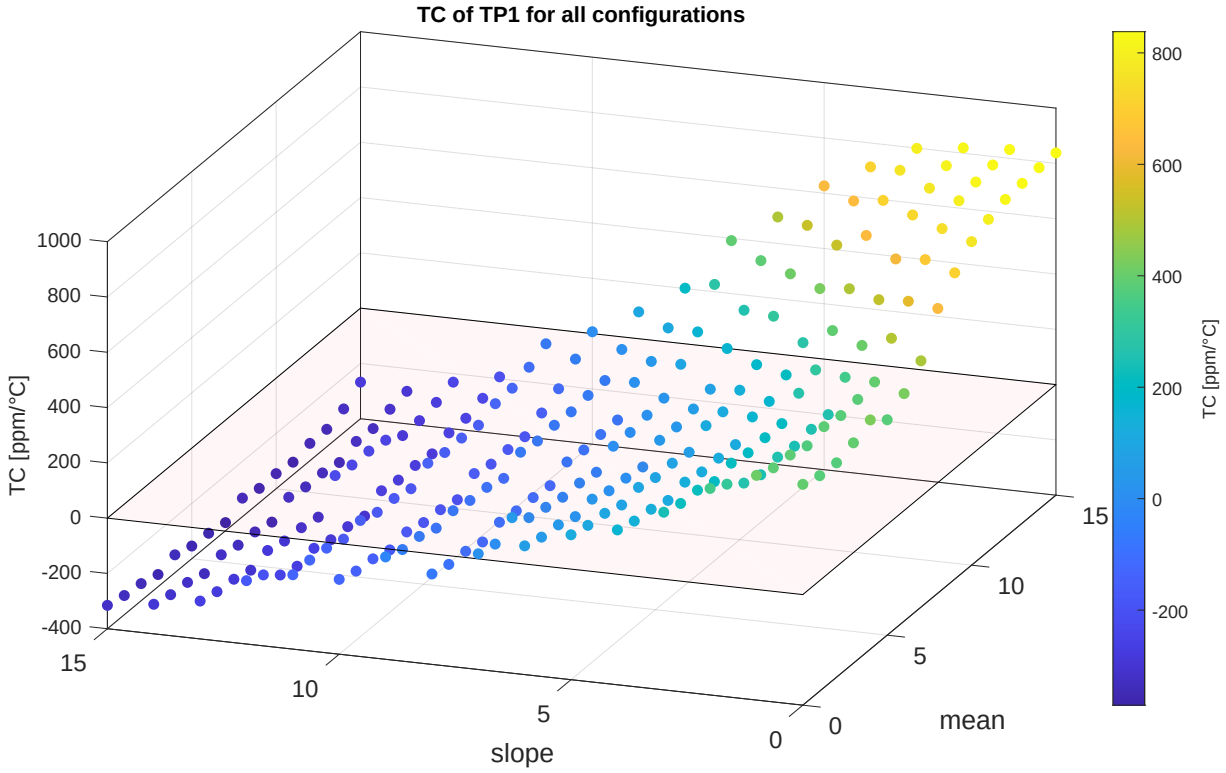


Figure 18: 3D plot of TP1 for all configurations and TCs. The red plane represents the ideal TC equal to 0.

In the following two figures 19 & 20 the data for TP1 are shown 2-dimensional once for –40 °C and 70 °C. On the x-axis the settings for the mean resistance R_2 are plotted. On the y-axis the corresponding bandgap voltages are plotted and every color represents a specific setting for the slope resistance R_0. In fact the diagrams are the projection of the lowermost points –40 °C and the uppermost points 70 °C from figure 17 to the mean/BGR-voltage plane. For the measurements at an almost linear behavior can be observed for mean settings up to 10. After that an effect of saturation occurs and the output voltage cannot be increased further. The same effect is more pronounced for the same TP at –40 °C. Here, saturation effects can already be seen from a mean setting of 8. These effects occur for all TPs especially for **high mean settings** and **low slope settings** and intensify for negative temperatures which can be seen in appendix F. The exact impact of this effect will be investigated in more detail in section 5.2. The data shown here are not related to the temperature, but are only plotted for one temperature point. It can be seen clearly that the trim settings both influence the mean value of the output voltage. So the interdependence will have to be considered in the calibration process.

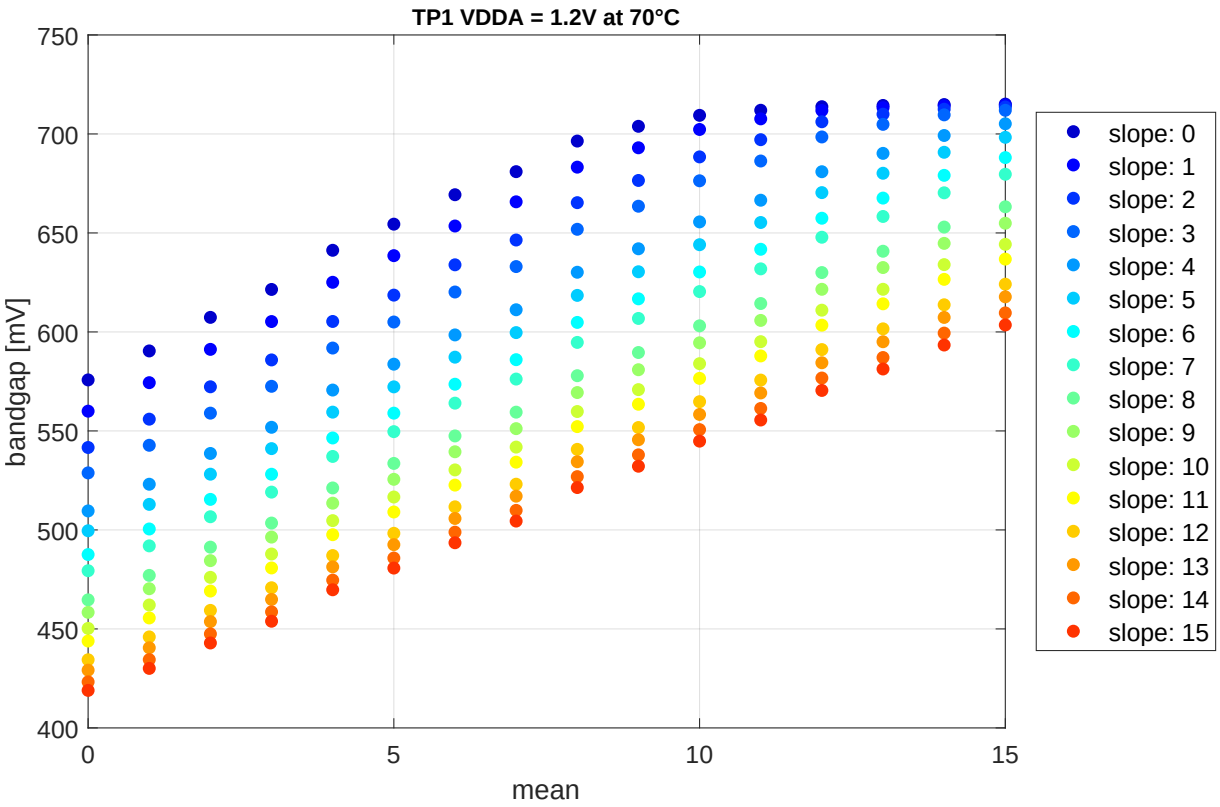


Figure 19: BGR over mean-settings of TP1 at 70 °C

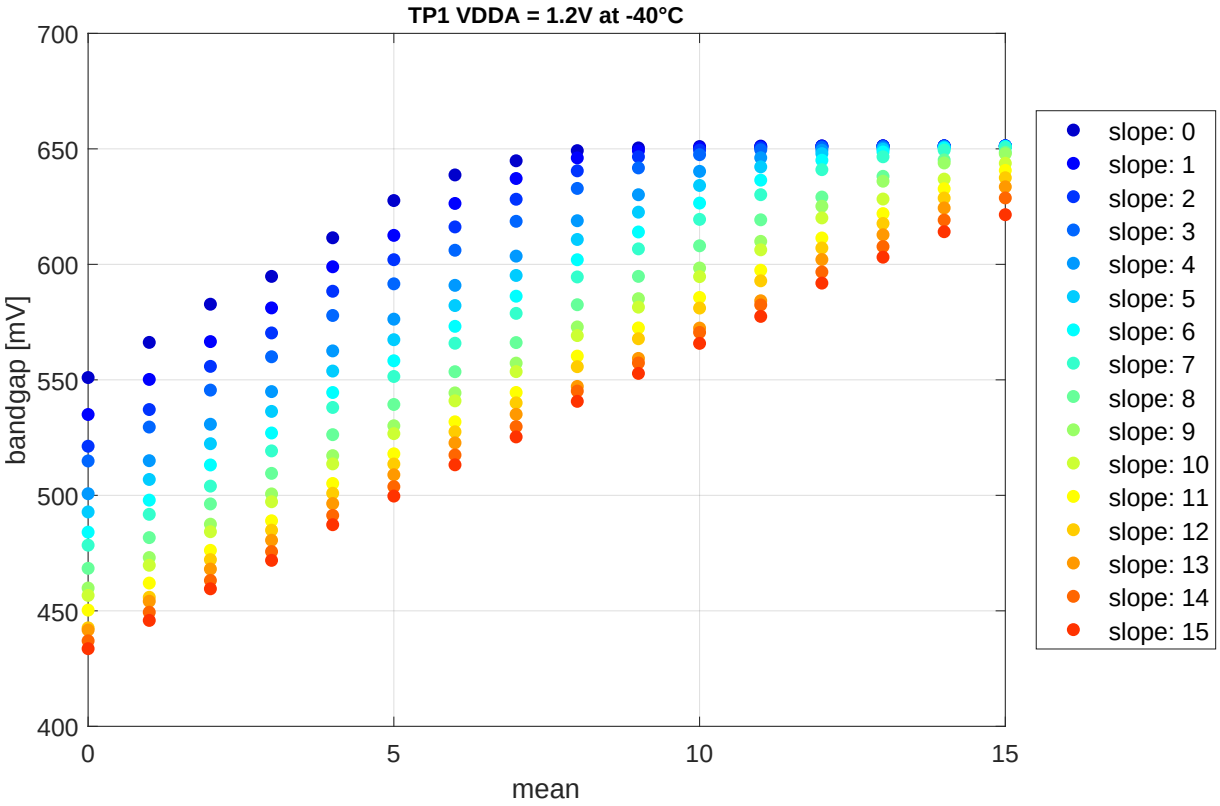


Figure 20: BGR over mean-settings of TP1 at –40 ℃

5.2 The quality of trimming

A correlation between mean and slope settings has already been shown in section 5.1. In this section it will be evaluated based on the data, how a change of the resistance affects the BGR voltage in detail. In order to make an objective statement, the following four properties were examined:

- **range and location of BGR voltages**
- **step width of mean**
- **step width of slope**
- **dependency between mean and slope settings**

5.2.1 Range of trimming

To check whether the values are distributed around the operating point at all, the range of the BGR voltages and their position were evaluated. Ideally, the ranges are equally distributed above and below the operating point. Table 1 shows that the minimum always occur at the configuration slope 15 mean 0. This is due to the fact that the slope for 15 becomes maximally

negative and, in addition, the mean takes very low values. The maximum values occur with almost all TPs for the configuration slope 0 mean 15. Due to the fact that the minima scatter much more strongly (σ = 17.1) than the maxima (σ = 0.4), one could also determine an effect of the saturation here. However, this cannot be said unambiguously on the basis of this diagram. The fact that the extreme points occur at 70 is due to the fact that the "pivot point" of the slope setting is in the negative range of the temperature, which can be seen in figure 21.

TP	BGR_{min} [mV]	temp. [℃]	slope	mean	BGR_{max} [mV]	temp [℃]	slope	mean
1	419.0	70	15	0	715.1	70	0	15
2	446.9	70	15	0	715.6	70	0	15
3	457.6	70	15	0	715.0	70	0	15
4	439.9	70	15	0	715.2	70	0	15
5	413.7	-40	15	0	714.3	70	0	15
6	441.7	70	15	0	714.9	70	0	15
7	426.9	70	15	0	714.9	70	0	15
8	460.4	70	15	0	714.7	70	0	15
9	457.7	70	15	0	715.0	70	1	15
10	438.3	70	15	0	715.2	70	0	15
11	451.2	70	15	0	715.2	70	0	15
12	434.9	70	15	0	714.8	70	0	15
13	437.4	70	15	0	714.3	70	0	15
14	405.4	-40	15	0	714.6	70	0	15
15	417.3	70	15	0	715.0	70	0	15
16	417.6	70	15	0	715.8	70	0	15

Table 1: Max and min values for each TP including accordingly temperature and configuration.

In table 2, the location of the range was compared to the mean of the distributions. The mean was taken for each TP for all configurations and all temperatures, as well as the associated standard deviation σ. Then the total range was calculated from $BGR_{min} - BGR_{min}$. The theoretical location ($location_{theo}$) of the range center is formed by $BGR_{min} + (range_{total}/2)$. This can then be compared to the mean μ ($location_{diff}$). This value should give an indication of how evenly the range is distributed. If $location_{diff}$ is positive, the mean value μ is above the range middle and for negative values the opposite occurs. To get a better overview of the distribution of the data, the median was also included in the table. It indicates at which value 50 % of the values lie above and below. Particularly TP5 and TP14 are conspicuous here, which are clearly below the operating point in the μ and median. TP8 stands out because it is clearly above the operating point in both, the μ and the median.

TP	μ	σ	$range_{total}$	$location_{theo}$	$location_{diff}$	median
1	578.8	70.7	296.1	567.0	11.8	583.3
2	604.9	65.1	268.8	581.3	23.6	615.7
3	614.2	63.1	257.4	586.3	27.9	629.1
4	599.9	66.5	275.3	577.5	22.3	609.6
5	567.5	71.9	300.5	564.0	3.5	570.0
6	597.9	67.2	273.2	578.3	19.6	606.7
7	584.2	69.9	287.9	570.9	13.4	589.8
8	617.3	61.3	254.3	587.6	29.7	633.5
9	615.4	62.3	257.4	586.4	29.0	630.7
10	600.1	66.8	276.9	576.8	23.4	610.6
11	609.6	63.7	264.0	583.2	26.4	622.3
12	592.3	67.8	279.9	574.8	17.5	599.8
13	593.4	67.4	277.0	575.9	17.6	601.3
14	560.5	73.8	309.3	560.0	0.5	561.7
15	574.8	70.9	297.6	566.1	8.7	578.8
16	574.8	71.8	298.2	566.7	8.1	578.2

Table 2: Mean-values and location of mean-values within range for each TP shown in mV.

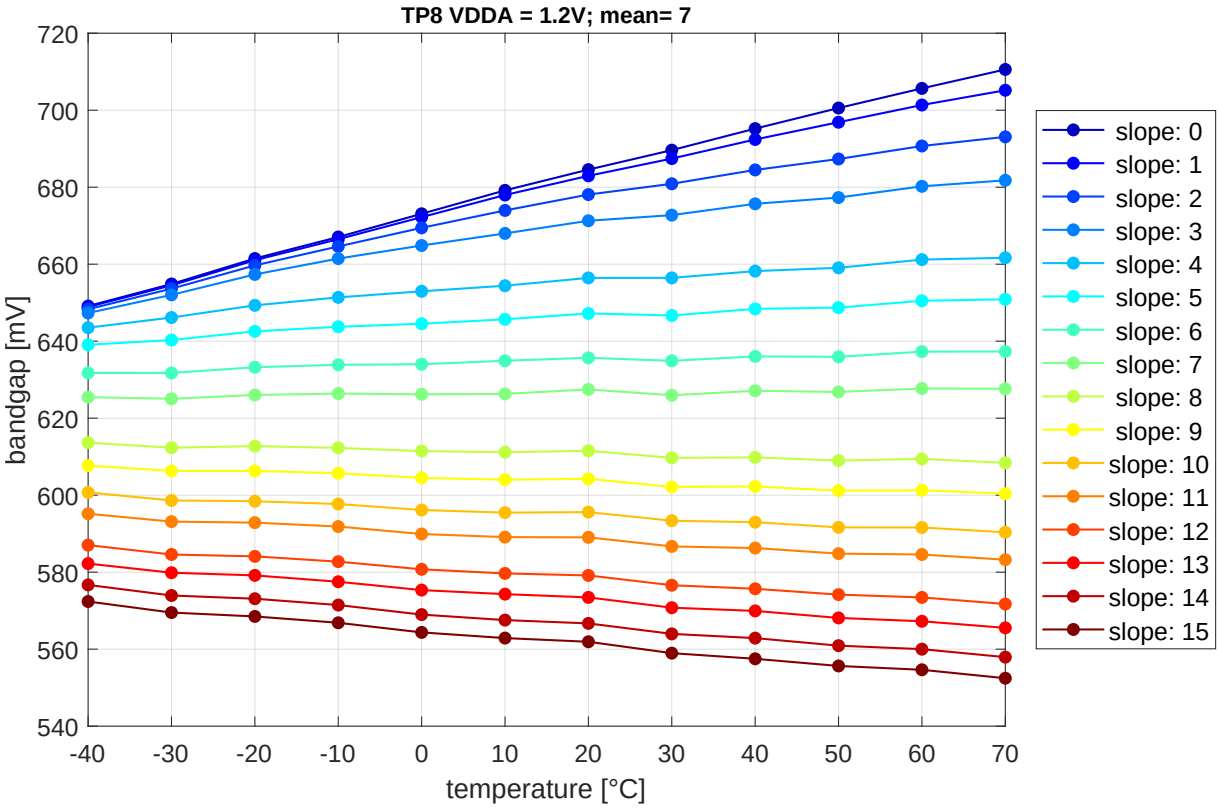


Figure 21: BGR voltages over temperature of TP8 at configuration: mean = 7. Different colors represent the slope values. Extreme values appear at 70 °C.

5.2.2 Step width of mean setting

To explain which step width is meant here I would like to refer to figure 22. In this plot the BGR voltage for TP1 is shown over the temperature. The slope setting is fixed at 10 and the colors show the different mean settings. The step width should ideally be the same for one temperature between two mean settings in distance of one. This approach results in a total of 15 step widths per temperature and twelve temperatures per step width.

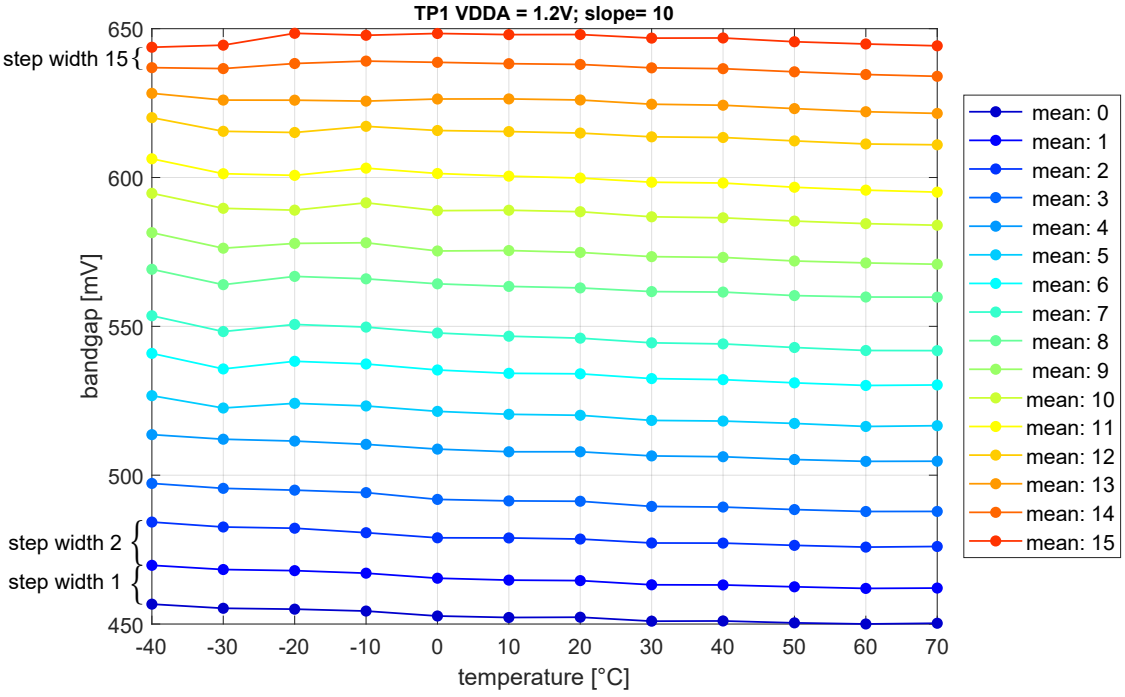


Figure 22: BGR voltages over temperature of TP1 at configuration: slope = 10. For illustration the nomenclature of the step width is shown graphically on the left. Different colors represent mean settings.

To illustrate the step width once again, figure 23 shows the output voltage over the mean settings. In each case for all twelve temperatures. Here one can also see that there is a dependence of the step width on the temperature. For high mean settings and low temperatures the BGR voltage can hardly be changed.

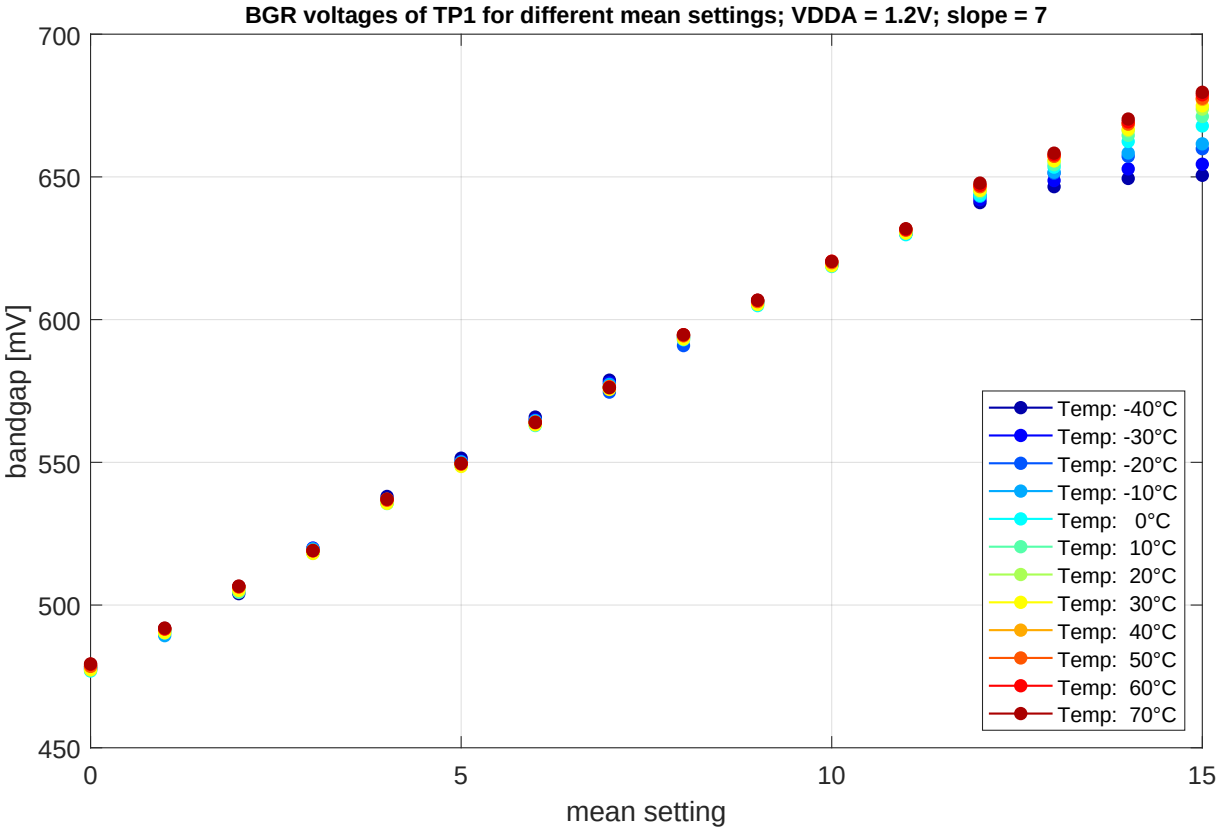


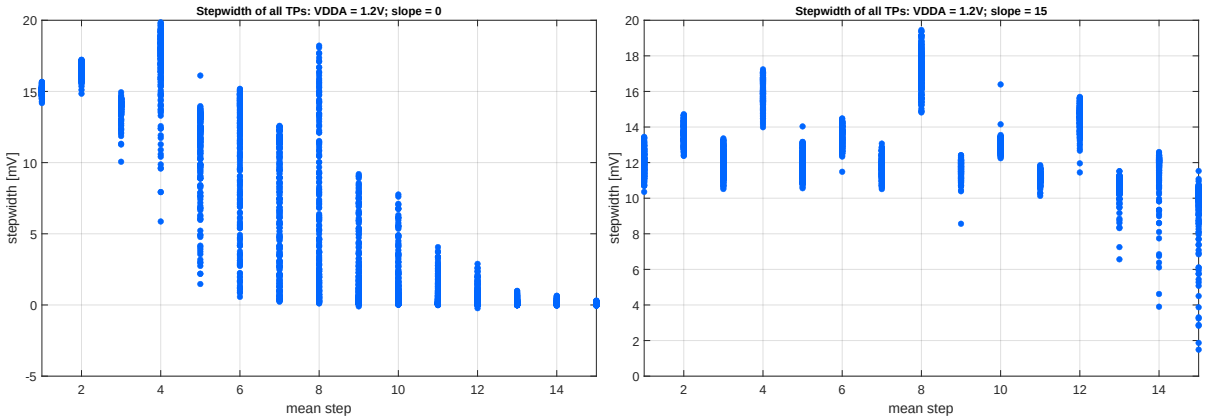
Figure 23: BGR voltages over mean settings of TP1 at configuration: slope = 7.

To get a better overview of the results they were summarized in a table 3. Here the mean value μ and the standard deviation σ for the slope settings 0, 7 & and 15 for the different step widths are shown. It can be seen that the μ values for slope setting 15 vary a lot but they don't take μ values smaller than 9. For slope setting 0, the step width varies for μ from about 15 mV to 0.05 mV. For this configuration, there is no longer any change worth mentioning for mean settings from 10 upwards, since it is in the range <1 mV.

	slope 0		slope 7		slope 15	
step nr.	μ [mV]	σ [mV]	μ [mV]	σ [mV]	μ [mV]	σ [mV]
1	15.040	0.275	13.495	0.630	12.035	0.656
2	16.550	0.395	15.257	0.575	13.671	0.556
3	13.830	0.664	13.344	0.569	11.971	0.651
4	16.794	2.538	17.662	0.736	15.999	0.719
5	10.656	3.129	13.080	0.403	11.952	0.610
6	9.677	4.540	14.618	0.272	13.494	0.534
7	6.305	4.166	12.575	0.316	11.795	0.559
8	6.097	5.536	17.818	1.386	17.227	1.041
9	2.375	2.693	11.560	0.954	11.606	0.493
10	1.489	1.929	12.162	2.100	13.060	0.362
11	0.653	0.886	9.496	2.609	11.236	0.300
12	0.418	0.583	10.737	4.440	14.531	0.681
13	0.155	0.208	6.559	3.599	10.605	0.628
14	0.098	0.133	5.915	4.156	11.443	1.404
15	0.050	0.066	3.861	3.335	9.360	1.854

Table 3: Step width of TP1 between bit-wise mean settings with nomenclature from figure 22. The table shows the μ values for extreme configurations slope: 0 & 15 and slope: 7 to compare.

This effect can be observed for all TPs, sometimes more and sometimes less strong. To illustrate this, the step widths for all TPs for the two extreme configurations slope setting 0 & 15 are shown quantitatively in figure 24. Here it can be seen that the step widths for high slope settings rarely take small values. The effect of saturation for high mean settings and low slope settings already mentioned in section 5.1 can also be observed here for the step widths. The corresponding BGR/temperature curves for the three slope settings 0, 7 and 15 can be found in appendix G.



(a) Step width for all TPs at configuration: slope = 0. (b) Step width for all TPs at configuration: slope = 15.

Figure 24: The two graphs show the step width between different mean values at fixed slope values for all 16 TPs.

5.2.3 Step width of slope setting

Likewise, the step width of the slope setting can be examined. For the three mean settings 0, 7 and 15 for all slope settings [0:15] a regression line was determined. The slope of the estimated linear model was then compared with the slope of the next higher setting which resulted when the slope settings was increased by one bit. This resulted in 15 differences between the respective steps for each mean setting. This was again considered for all 16 TPs. Table 4 shows the μ and σ values of the distributions of the slope differences.

	mean 0		mean 7		mean 15	
step nr.	μ	σ	μ	σ	μ	σ
	$\times 10^{-5}$ mV					
1	3.01	0.76	7.10	2.52	0.20	0.16
2	4.13	0.22	9.50	3.19	0.51	0.44
3	2.69	0.59	6.04	2.60	0.80	0.75
4	4.95	0.42	8.64	2.99	3.15	2.95
5	2.13	0.37	3.26	1.05	3.13	2.57
6	2.62	0.33	3.54	1.21	5.34	3.48
7	1.59	0.74	1.58	1.67	4.90	2.44
8	2.27	3.33	0.54	4.49	11.53	3.53
9	0.14	1.37	-0.80	1.40	5.74	1.54
10	-0.67	1.89	-1.66	1.69	6.71	2.29
11	-0.58	0.94	-1.13	1.06	3.85	3.67
12	-1.57	1.74	-2.18	1.84	4.98	6.40
13	-0.82	0.77	-1.06	0.89	0.56	4.34
14	-1.06	0.86	-1.43	0.81	-0.07	4.70
15	-0.65	0.67	-0.98	0.50	-0.82	3.19

Table 4: Average step width between slope steps for all TPs.

5.2.4 Dependency between mean and slope settings

In this section, I would like to examine the relationship between changes in slope settings on changes in mean voltage. To check if there is a dependency between the two parameters, the correlation of the two parameters can be determined. This is done by fixing the mean setting at 7 and looking at the BGR voltages for changes in the slope settings and vice versa. However, when looking at the plot 21, one can see that for the temperature range shown here, the slope can not be set without changing the mean value.
Since data are available in scaled intervals for the parameters slope and mean [0:15] and the temperature deviations are considered negligible, the **correlation coefficient** between mean and slope can be calculated using the data of the parameters. However, the basis for this is that the correlation between the parameters is approximately linear. Figure 25 shows the two characteristics and their linearized straight lines. Since the maximum deviation from the measured value to the linearized straight line for both parameters is below 5 %, the distributions can be considered sufficiently linear. For TP1, the calculation yields:

$$r = -0.991$$

Which means that there is a strong reciprocal relationship between the two parameters. Same can be seen for all other TPs. As can also be seen in figure 63 (appendix H), the mean value

always changes with different slope settings. In order to achieve independence between the two parameters for the temperature range given here, the point of intersection of the lines shown in figure 63 (appendix H) would have to be at 15 °C.

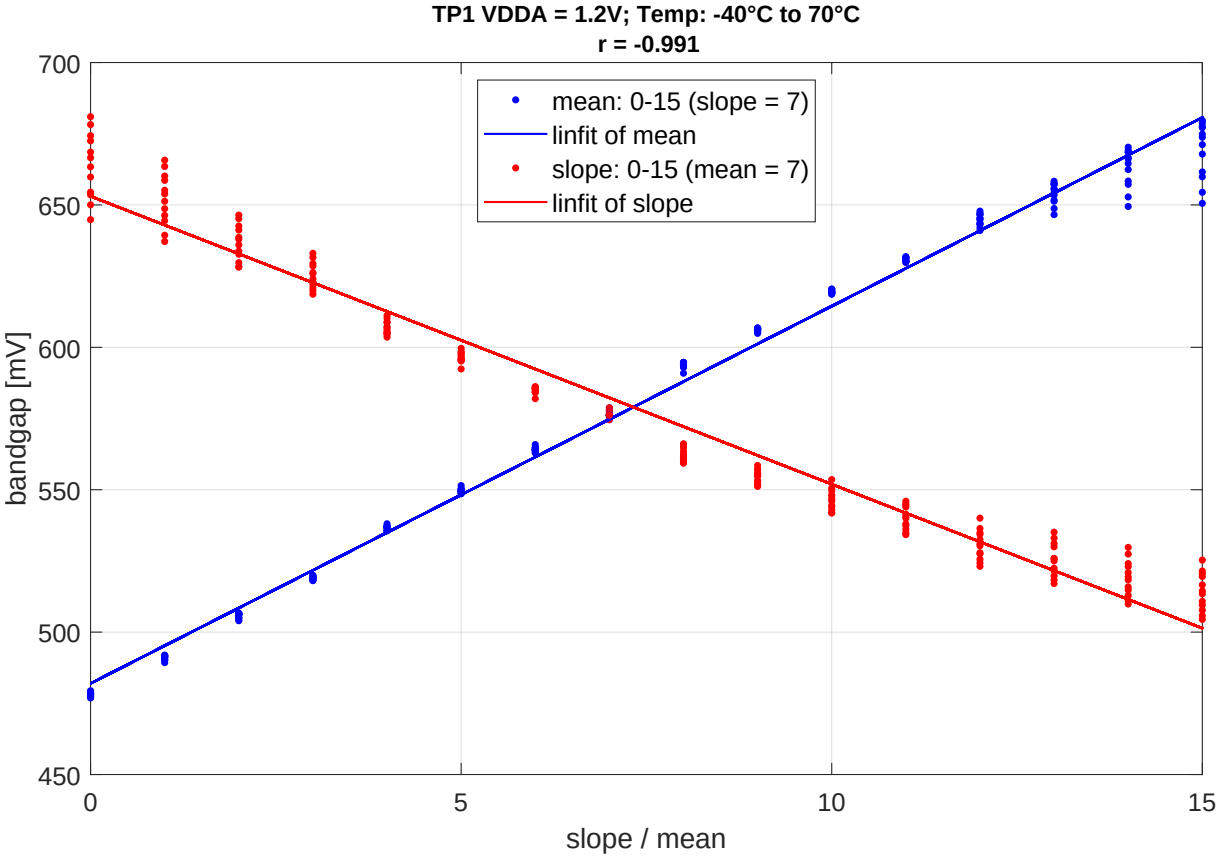


Figure 25: BGR of TP1, where red represents BGR voltage over slope with fixed mean at 7 and blue BGR voltage over mean with fixed slope at 7. For each configuration there are twelve values for the respective temperatures.

5.3 Linearity of the system

One of the trim parameters is the temperature coefficient. The notion "temperature coefficient" is based on a linear model for the output voltage change depending on the temperature. Therefore it is advisable to investigate validity and limitations of this linear model. To investigate this, a linear regression was performed for the configuration mean setting 7 and slope [0:15]. The deviations from the measured values were then evaluated. The linear regression can be calculated using formula 24. This formula provides a straight line that is as close as possible to all measured values. The deviation of the temperature is again assumed to be negligible and thus the deviation from measured value to linearized system can be considered to be the *y-error*. For this purpose, the slope was varied for the fixed mean setting of 7 for each TP [0:15] and the resulting distribution of the *y-error* was statistically evaluated by its μ and σ values. Figure

26 shows such an error distribution for TP1. The results for all TPs have been summarized in table 5. Here σ, μ, and the absolute *y-error* in average are presented.

TP	μ [mV]	σ [mV]	| y-error| [mV]
1	-5.3×10^{-14}	0.815	0.6
2	-6.1×10^{-14}	0.912	0.8
3	-6.9×10^{-14}	0.911	0.6
4	-7.2×10^{-14}	0.829	0.7
5	-5.0×10^{-14}	0.624	0.5
6	-4.0×10^{-14}	0.506	0.4
7	-6.1×10^{-14}	0.793	0.7
8	-9.7×10^{-14}	0.852	0.6
9	-5.8×10^{-14}	0.696	0.5
10	-5.2×10^{-14}	0.821	0.6
11	-8.3×10^{-14}	0.673	0.5
12	-7.0×10^{-14}	0.470	0.4
13	-5.5×10^{-14}	0.449	0.4
14	-6.2×10^{-14}	0.502	0.4
15	-4.2×10^{-14}	0.386	0.3
16	-5.9×10^{-14}	0.351	0.3
mean-value	~0.0	0.662	0.5

Table 5: Standard deviation and mean-value of error distribution for each TP for fixed mean = 7 including rounded up *y-error* in absolute numbers averaged over temperature.

By looking at the μ value one can see that the values are in the order of 10^{-14} which can be seen as 0 in this case. It follows that the mean error becomes zero and calculations were performed correctly. The *y-errors* are assumed to be normally distributed. And therefore 68 % of the values are in a range of $0 \pm \sigma$. In order to evaluate whether this deviation is sufficiently small to describe the system as linear and at what point it is no longer linear, the influence of the *y-error* on the calculation of the TC was considered. If the linearity is sufficient, the two results differed only minimally. However, if the *y-error* increases, the results quickly differ greatly and the linearity is no longer sufficient. In table 5 the *y-errors* are shown as absolute value and averaged over temperature. It is important to consider these errors in absolute terms, as they can cancel each other out.

In this observation, it was found out that from a *y-error* of more than 0.8 mV **in average over temperature**, the difference between the two TC methods increases above 5 %. Since the TC is given in ppm and here 5 % deviation clearly no longer reflect the reality, curves with a *y-error* higher than 0.8 mV will not be considered in this work.

If we now take a closer look at table 5, we can see that there are already some absolute *y-errors* close to this range. However, this is due to the fact that the representation was averaged over all slope settings. If we look at each slope separately, as it is done in table 6 we see that the

majority of the values are below the limit, although here, too, the average was taken over the twelve temperature points.

slope	0	1	2	3	4	5	6	7
y-error [mV]	1.2	1.3	0.7	0.4	0.5	0.7	0.6	0.7
slope	8	9	10	11	12	13	14	15
y-error [mV]	0.6	0.7	0.6	0.7	0.6	0.6	0.6	0.6

Table 6: Y-errors of TP4 at configuration mean = 7. The *y-errors* are shown in absolute mean values of temperature.

From the data reported it can be concluded that the system itself is sufficiently linear in most of its operating points, especially close to its operating point. This means that for the linear range the TC can be used as a description and the non-linear measurement data do not have to be used. An example for TP1 is shown in appendix H. Figure 63 shows the measurement data over temperature where figure 64 presents the regression line for each slope including the error bars. Figure 26 shows the corresponding distribution of the *y-errors*.

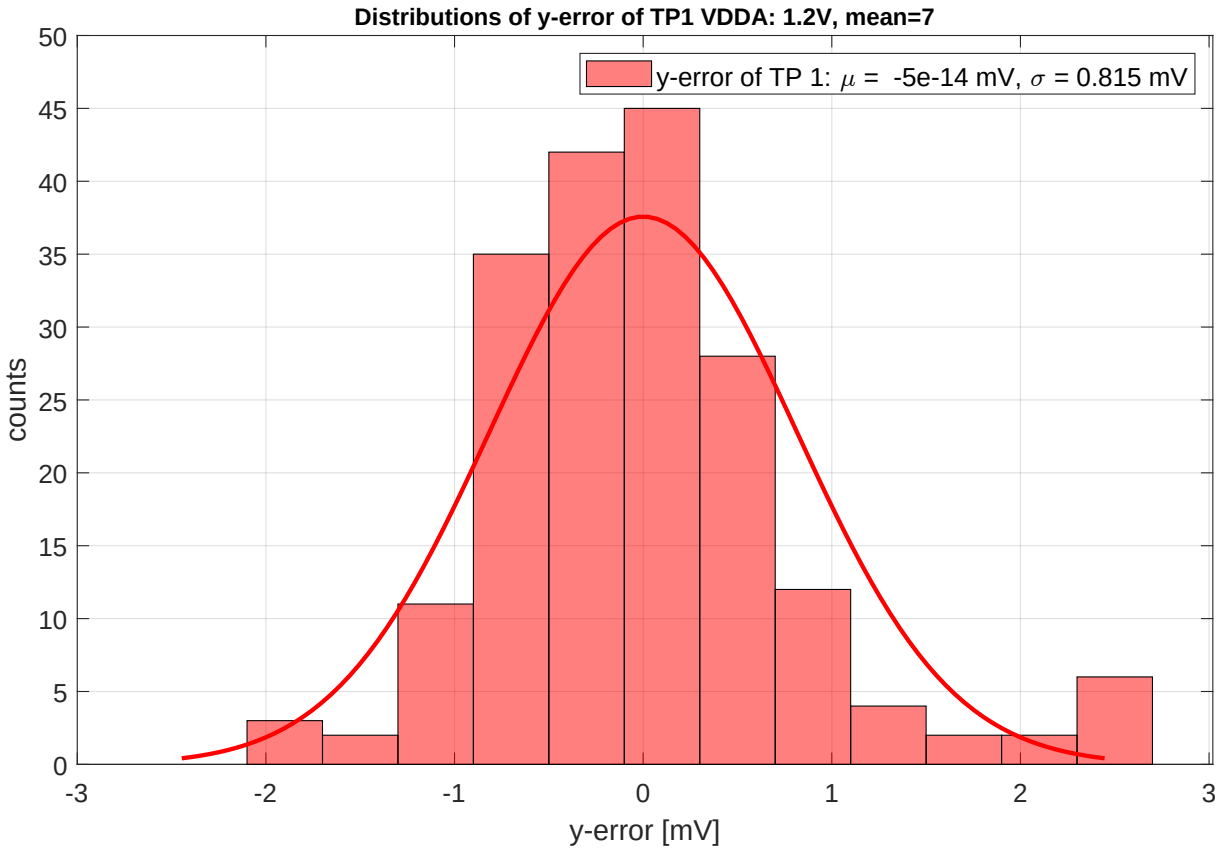


Figure 26: Distribution of y-errors between measurement and linear regression of TP1 at configuration: mean = 7.

The temperature coefficient is determined by the two BGR extreme values which occur for the minimum and maximum temperature. Due to this relationship, the location of the occurrence of

the maximum *y-error* must also be considered. If this is often found at the two extreme values, i.e. –40 °C or at 70 °C, the use of the linearized model should be considered despite the above mentioned properties. When evaluating the *y-error* as a function of temperature for the majority of the test points, the maximum deviation in absolute as well as percentage terms was found at –40 °C. For none of the TPs the maximum deviation was found at 70 °C. However, the maxima that occurred here, were only slightly larger relative to the second largest deviation, which again occurred randomly. Expressed in figure 65 shown in appendix G, the percentage difference between the two maxima was in the range of the 2nd to 3rd decimal place. Although a trend is discernible here, the dependence of the maximum deviation on the temperature can also be regarded as negligible, since the deviations are very small in percentage terms and hardly change the temperature coefficients. Figure 27 and 65 (in appendix H) show the distributions of the of the *y-error* versus temperature once in absolute numbers and once as a percentage for TP14.

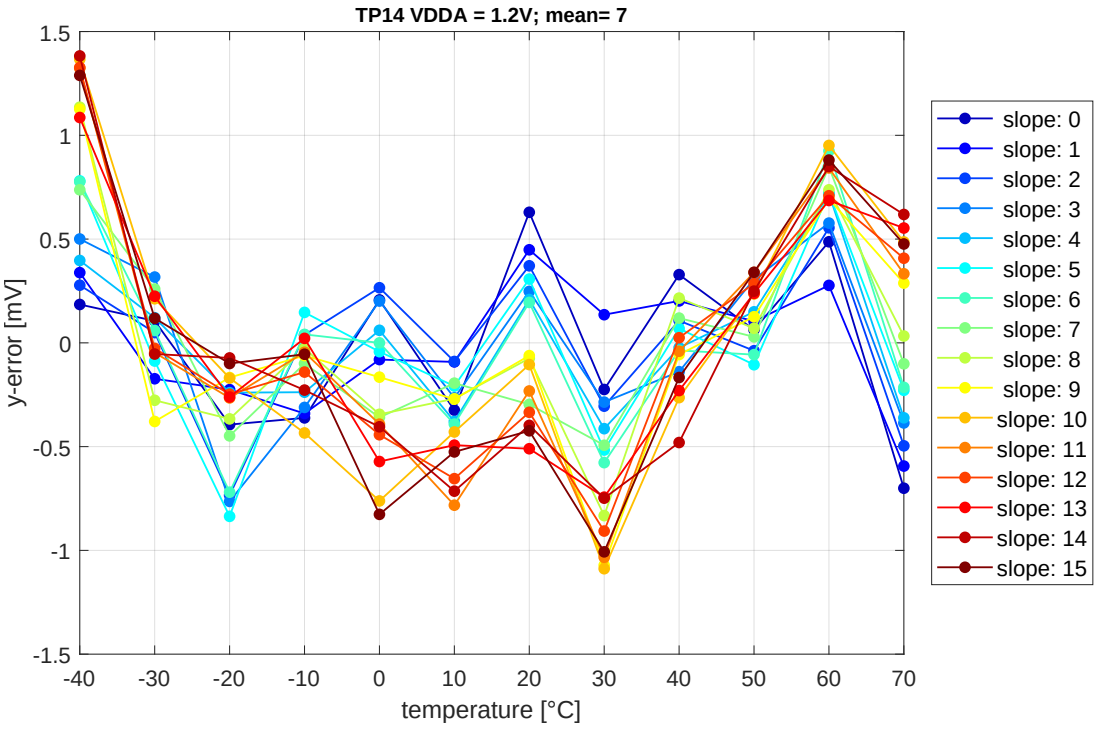


Figure 27: Distribution of y-errors between measurement and linear regression of TP14 at configuration: mean = 7.

5.3.1 Nonlinearities of the system

For configurations where both the slope and mean settings are in the top range, nonlinearities were detected for all TPs. The magnitude of this non-linearity is directly related to the percentage *y-error*. However, this effect only occurs at output voltages far outside the operating point of 600 mV. Even in the extreme example shown in figure 28, the maximum percentage *y-error* is less than 1.2 %. The nonlinearity is also visible when considering the mean distribution for

the slope setting, which is shown in plot 29. However, it is less noticeable here. The effect therefore hardly plays a role in the evaluation of the minimum distance, since it only occurs with configurations whose BGR voltage is significantly far away from the operating point. But it does have an effect when considering the TC, which will be explained in the following chapter.

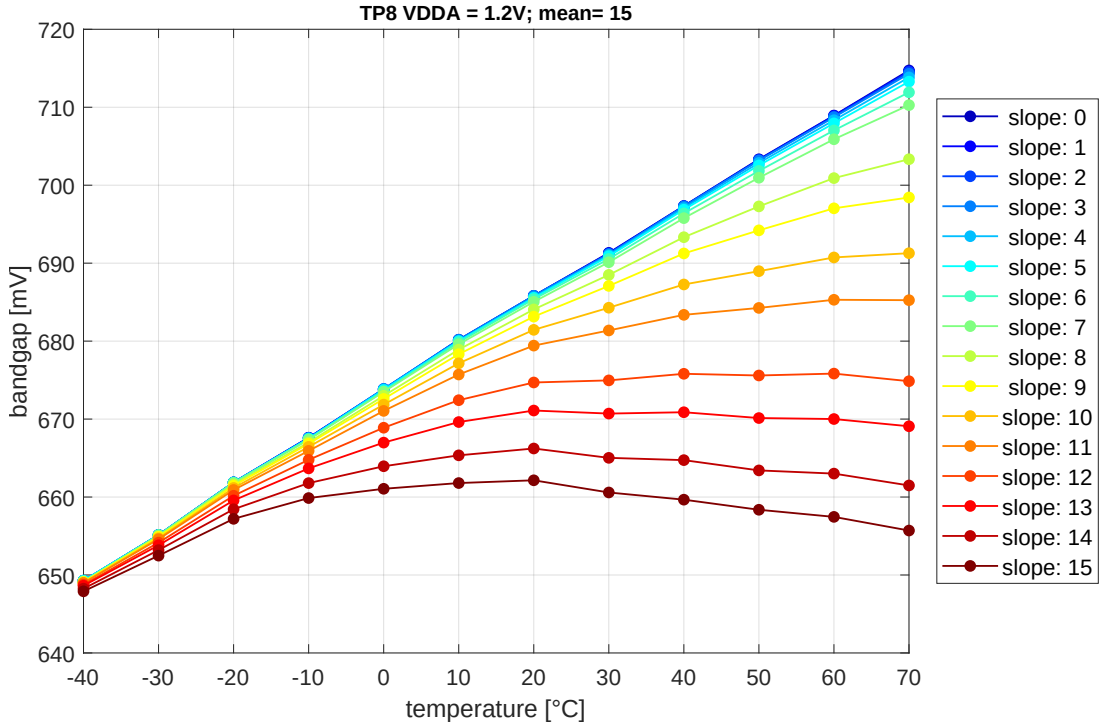


Figure 28: Measured data points of TP8 at mean setting 15. Different colors represent the slope setting [0:15].

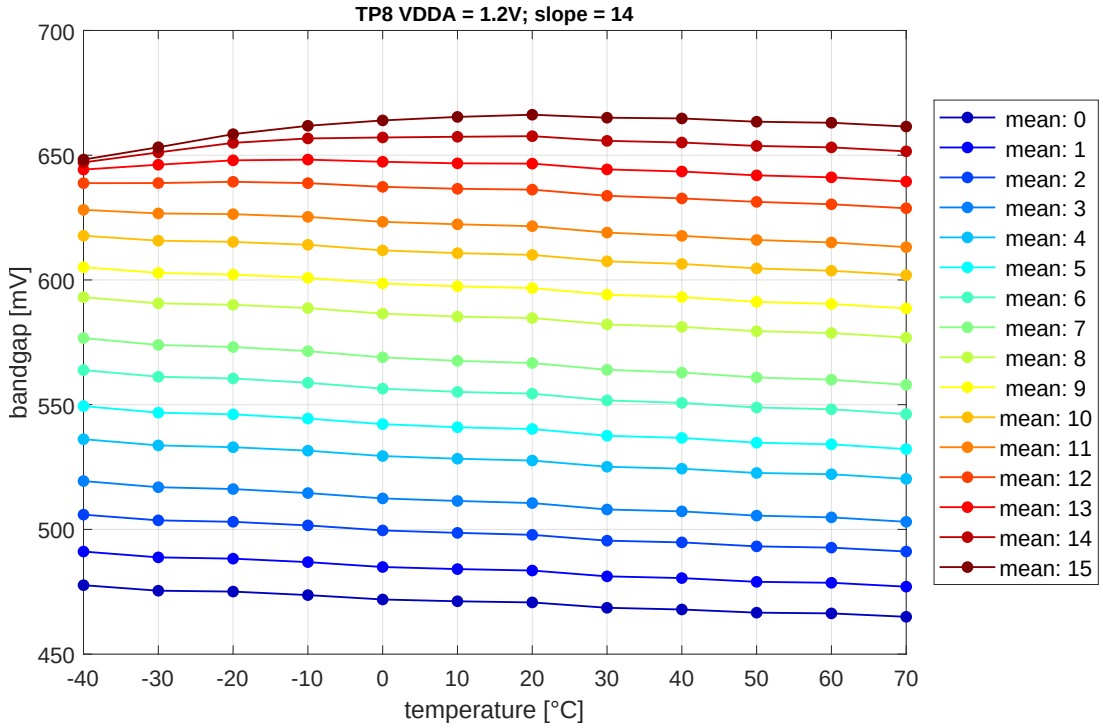


Figure 29: Measured data points of TP8 at slope setting 14. Different colors represent the mean setting [0:15].

5.4 Determine the best configuration

This section explains which methods can be used to find a configuration of slope and mean setting for each TP that meets the requirements for TC and BGR. On the one hand the TC should be as low as possible and on the other hand the BGR voltages should be as close as possible to 600 mV. Figure 17 shows a 3 dimensional plot all BGR voltages for TP1 at input voltage of 1.2 V. The light blue plane represents the design point. To find a configuration for each TP two different methods were developed and compared.

Method 1:
Focus: lowest TC
In this method, the TC was first calculated for all configurations and each TP. Then the configuration that has the lowest TC is determined. Since up to this point no consideration of the BGR voltage has taken place, in a further step the slope setting which yielded the lowest TC value is fixed for each TP. This slope setting is then used to determine the mean setting for one temperature (30 °C) which is closest to 600 mV. This mean setting is then used to determine the TC which becomes minimum for this mean setting. And finally the corresponding slope setting is determined.

Method 2:
Focus: lowest BGR voltage error

First a limit for the TC is selected. Based on the simulations already performed by Pezzoli [11], which resulted in a value of approx. 35 ppm/℃ for the design point, this was set to ±50 to ensure that sufficient configuration options are available. The TC limits are visualized by the reddish planes in figure 30, which shows a 3 dimensional plot of all TCs for TP1. Comparing this plot to the 3D plots shown in figure 17 one can see that the TC plot has one dimension less, since the temperature has already been included in its calculation. With this method, all configurations above the limits are already sorted out in the first step and are not considered further. Then, in a next step, the average value of the twelve BGR values for each temperature is determined for each remaining configuration. Subsequently, the configuration for the respective TP is finally determined for the average value that shows the smallest deviation from the design point.

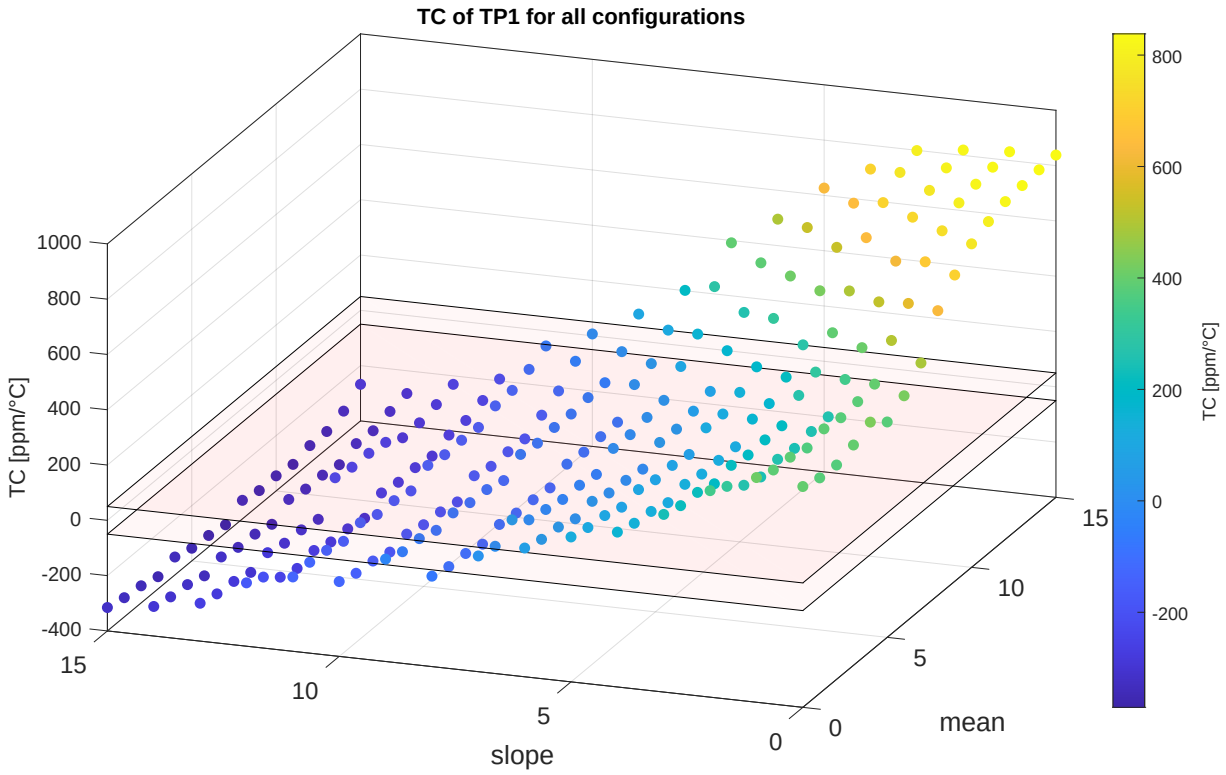


Figure 30: 3D plot of TP1 for all configurations and TCs. The red layers represent the lower and upper limit for the TC (±50 ppm/℃) used in Method 2.

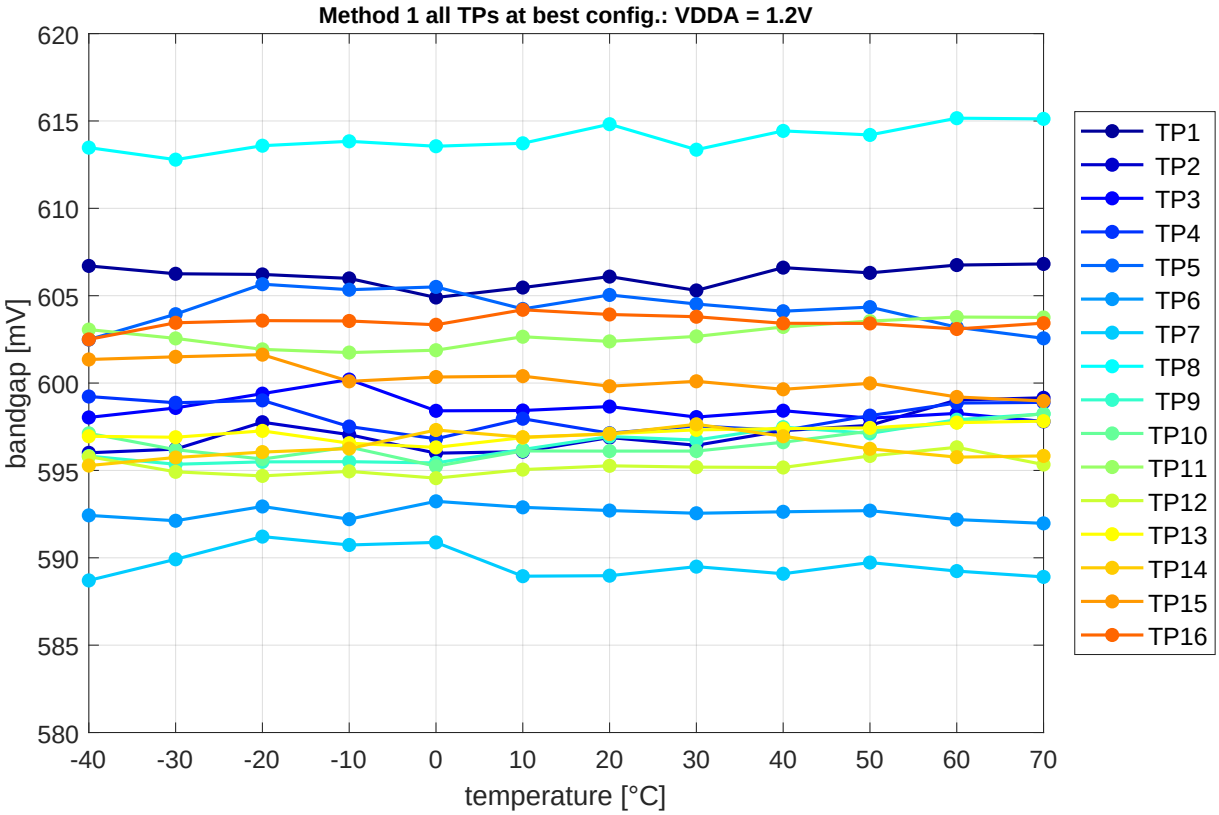


Figure 31: BGR voltages for all TPs at their best configuration found by Method 1.

The configurations found using Method 1 provide satisfactory results. The resulting BGR voltages are shown in figure 31. Even at a superficial glance, two outliers can be identified, namely TP7 and TP8. However, in order to be able to make a statement about the quality of the method, it makes sense to look at the distribution here as well. The mean value for all 16 TPs is 599.29 mV with a standard deviation of 5.7 mV, which can be seen in figure 33. Which configurations for the respective TP led to this result is shown in table 7.

The resulting BGR voltages of Method 2 are shown in figure 32. In this case, it is much more difficult to detect outliers when looking at the data superficially. Therefore, the distribution was also used here for an objective view. Since there is sufficient data available and temperature errors are negligible, a normal distribution is assumed here, shown in figure 33.

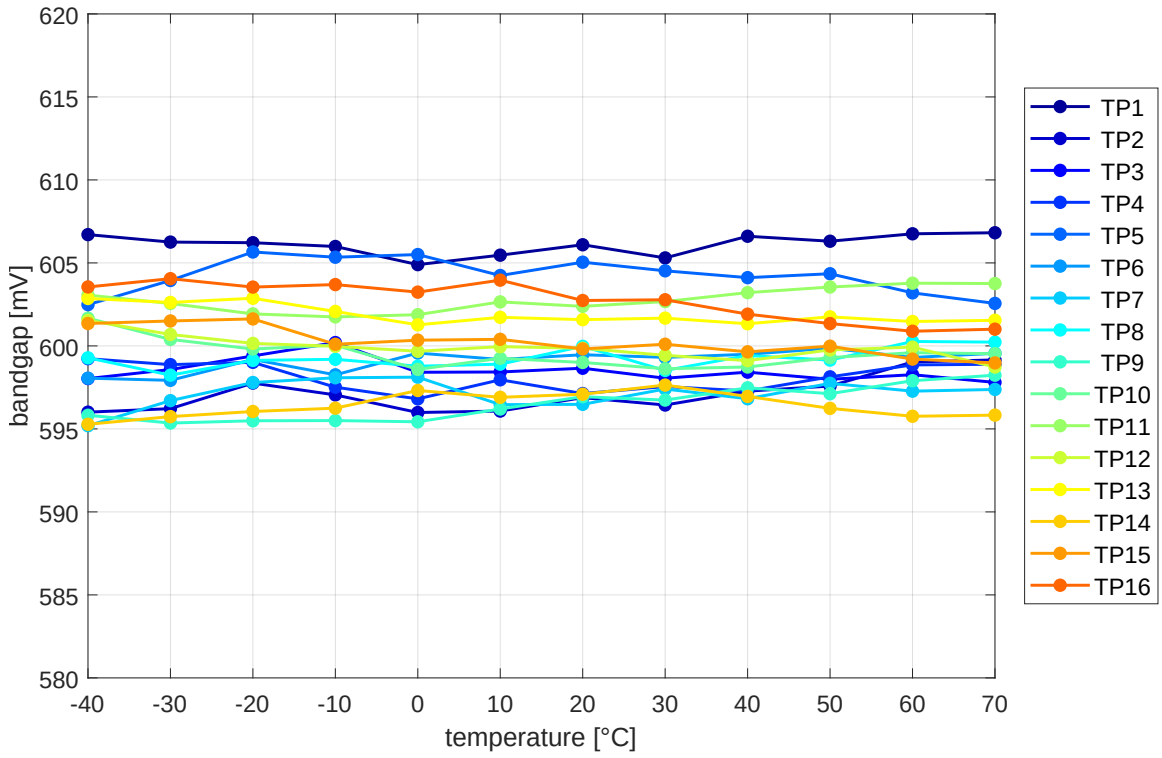


Figure 32: Distribution of BGR voltages for all TPs at their best configuration found by Method 2.

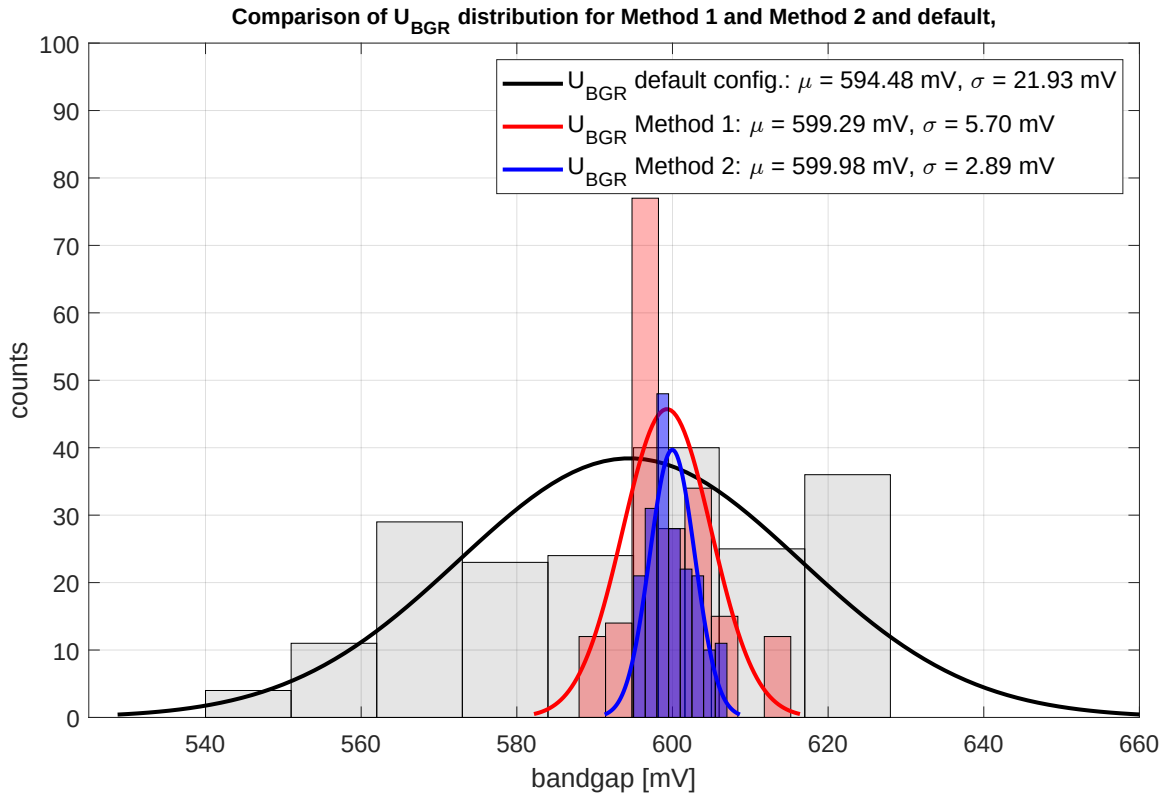


Figure 33: Distributions of U_{BGR} for all TPs at configurations found by Method 1, Method 2 and default (slope = mean = 7).

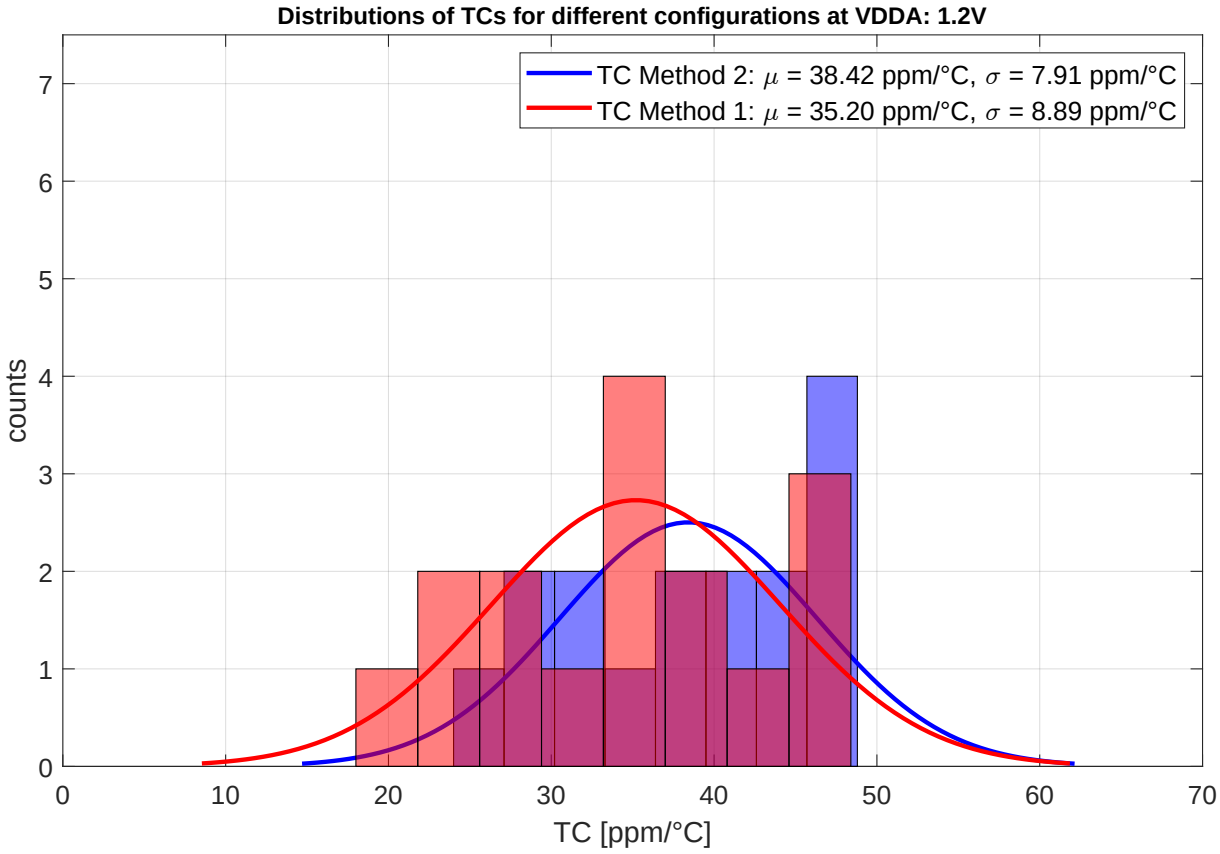


Figure 34: Comparison of distributions of |TC| for all TPs at their best configuration found by Method 1 and Method 2.

Although the configurations match to some extent, they provide slightly different results when viewed across all TPs. **Method 1** gives slightly better results in terms of TC, which can be seen in figure 34. For reasons of better comparability, the distributions of absolute TC values were considered.

Method 2 gives a better distribution of BGR voltages around the design point. It should be emphasized that, although it seems minimal, the standard deviation of the voltages around the design point was reduced from 6 mV to 3 mV in **Method 2** compared to **Method 1**. It should also be noted that TP5 and TP15 show their best configurations already close to the limits of the resistances (slope: 14; mean: 15). It can also be seen here that the slope settings for almost all TPs are in the upper trim range (slope > 7). The mean settings are also all distributed in the upper three quartiles.

	Method 1				Method 2			
TP	$\bar{U}_{\mathrm{BGR}}$ [mV]	slope	mean	|TC| [ppm/℃]	$\bar{U}_{\mathrm{BGR}}$ [mV]	slope	mean	|TC| [ppm/℃]
1	606.1	7	9	28.84	606.1	7	9	28.84
2	597.1	7	6	48.29	597.1	7	6	48.29
3	598.5	9	7	36.47	598.5	9	7	36.47
4	598.1	6	6	36.83	598.1	6	6	36.83
5	604.2	14	15	47.70	604.2	14	15	47.70
6	592.5	11	9	19.35	599.1	10	9	29.30
7	589.7	9	9	38.69	597.1	8	9	44.51
8	614.0	7	6	35.16	599.3	7	5	30.81
9	596.5	7	5	43.93	596.5	7	5	43.93
10	596.6	5	5	45.81	599.5	6	6	46.83
11	602.8	7	6	30.66	602.8	7	6	30.66
12	595.3	8	8	26.99	599.9	9	9	42.58
13	597.1	8	8	22.85	601.9	9	9	24.18
14	596.4	14	15	35.91	596.4	14	15	35.91
15	600.3	8	10	40.16	600.3	8	10	40.16
16	603.5	9	11	25.50	602.7	11	12	47.75
μ	599.3			35.2	600.0			38.4

Table 7: Results of the configurations achieved with the two methods.

5.5 Dependency of the output voltage on the input voltage

As already investigated, the BGR voltage depends on the temperature. This dependence can be strongly reduced and linearized with the methods carried out in section 5.4. However, there is another dependence of BGR voltages, namely that on the input voltage. Different input voltages cause different output voltages. The design of the arcadia testboard was developed for stability of the BGR voltages on input voltage variations of up to $\pm 10\,\%$. This means that the BGR voltages should remain largely constant in a range of the input voltage of 1.08 V to 1.32 V. To check how constant the BGR voltages react to fluctuations in the input voltage, the linearity can be examined as already in section 3.2. Another parameter is the line regulation, which was already explained in section 3.5. With the help of these two methods it is examined in the following for which range of the input voltage the BGR voltage can be considered *constant*. The basis of this evaluation is the data collected with the V_{in}/V_{out} measurement.

5.5.1 Jumps

When looking at the V_{in}/V_{out} measurements, an unexpected effect was noticed. Jumps in the output voltages occur at some test points. Comparing the two figures 35 and 36, it can be seen that the occurrence of these effect is dependent on whether the input voltage is step wise increased from 0 V to 1.32 V (fig. 35) or decreased from 1.32 V to 0 V (fig. 36). These jumps were not foreseen in the design and for now there is no conclusion about why they appear. However, comparing the results from the V_{in}/V_{out} measurement with the same configuration of the *characterization* measurement (slope = mean = 7) and looking at the BGR voltages at 1.08 V and 1.32 V input voltage respectively, it appears that this effect did not occur in the *characterization* measurements and is thus at least partially caused by the step change of the input voltage, since the circuit was not designed for this operation.

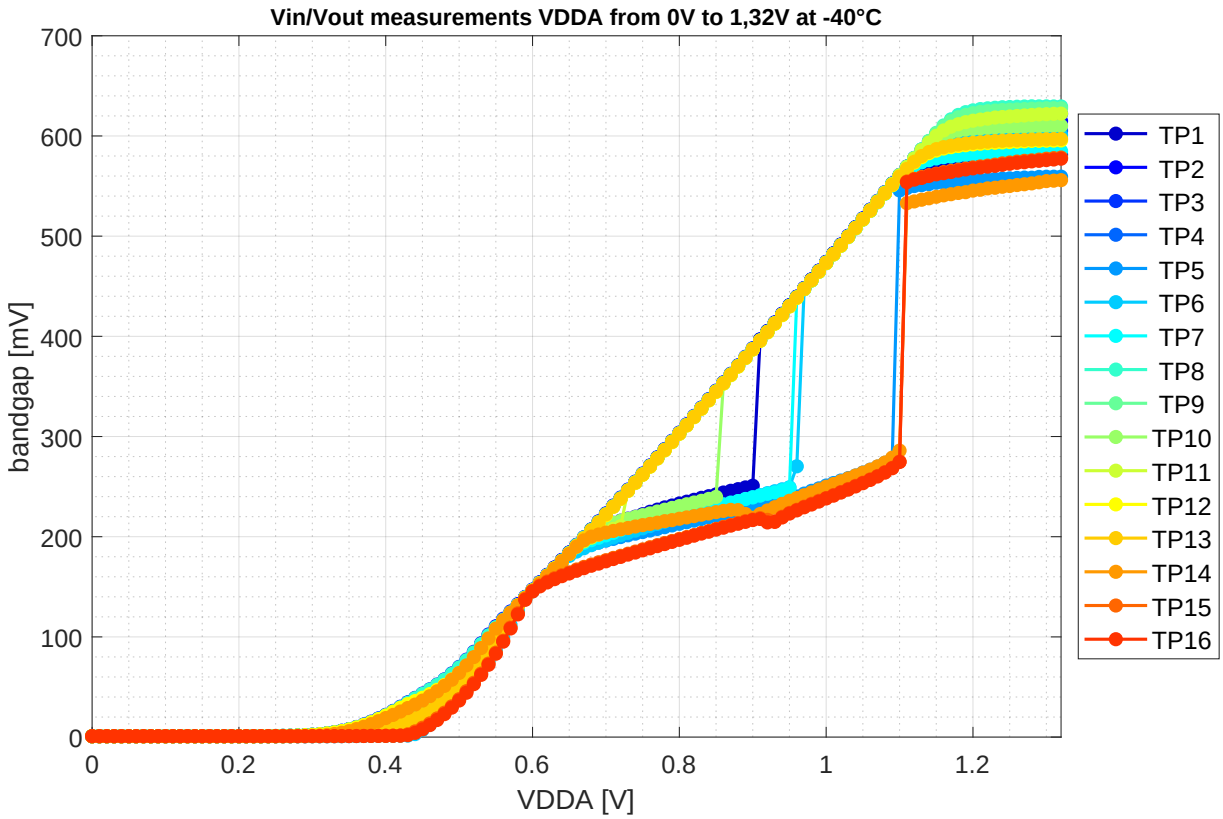


Figure 35: V_{in}/V_{out} curve of all TPs at –40 °C. Input voltage was increased from 0 V to 1.32 V in steps of 10 mV.

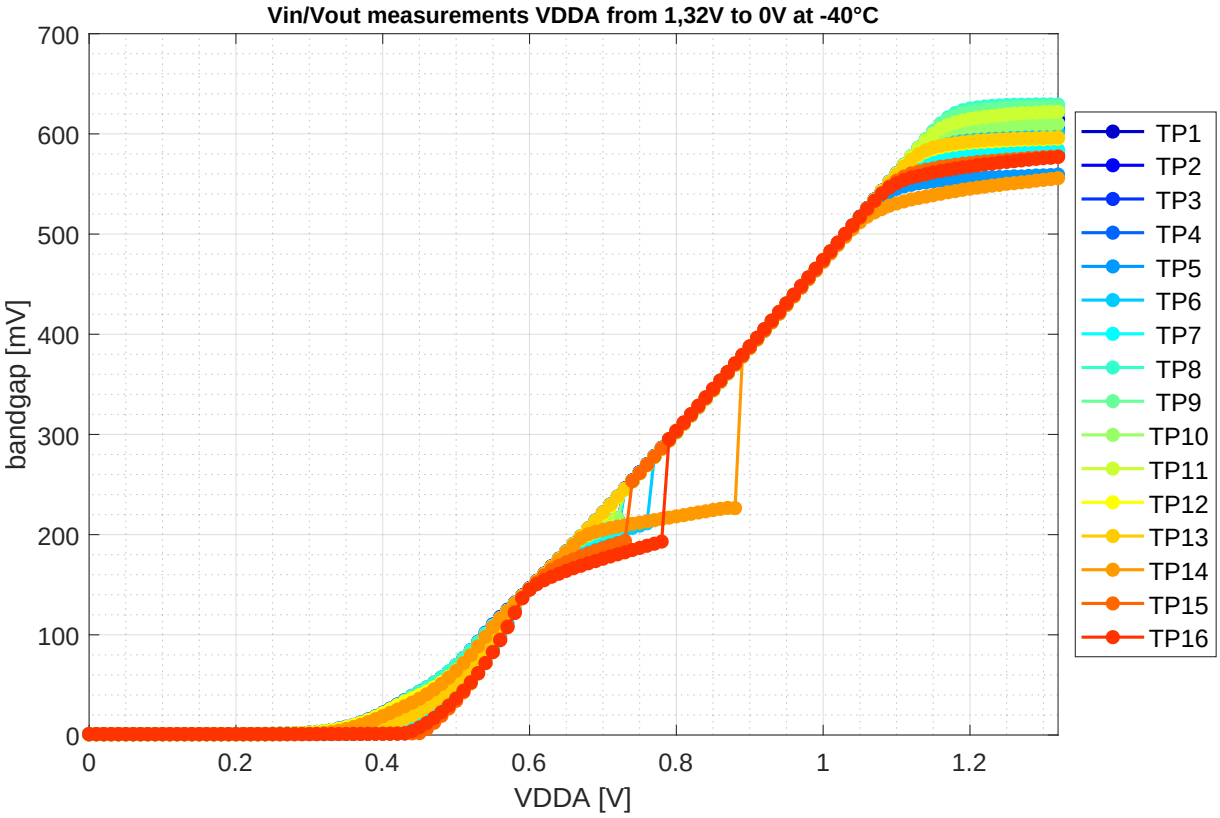


Figure 36: V_{in}/V_{out} curve of all TPs at –40 °C. Input voltage was decreased from 1.32 V to 0 V in steps of 10 mV.

5.5.2 Dependency on temperature

Furthermore, this effect occurs to different degrees for different temperatures. In this context, this means that the jumps still occur at higher input voltages. Figures 37 and 38 show the intended operating range of 1.2 V± 10 % once for 30 °C and once for –40 °C. The jump in BGR voltages was found to occur at 30 °C for particularly high input voltages. At this temperature the jumps occur at up to ~1.16 V, which is already within the operating range of the circuit. Jumps in the operating range also occur at –40 °C, but the effect is a little smaller here. At 70 °C, jumps also occur, but outside the operating range, which is shown in Appendix I. A correlation with the temperature has therefore been established, but it is not linear.

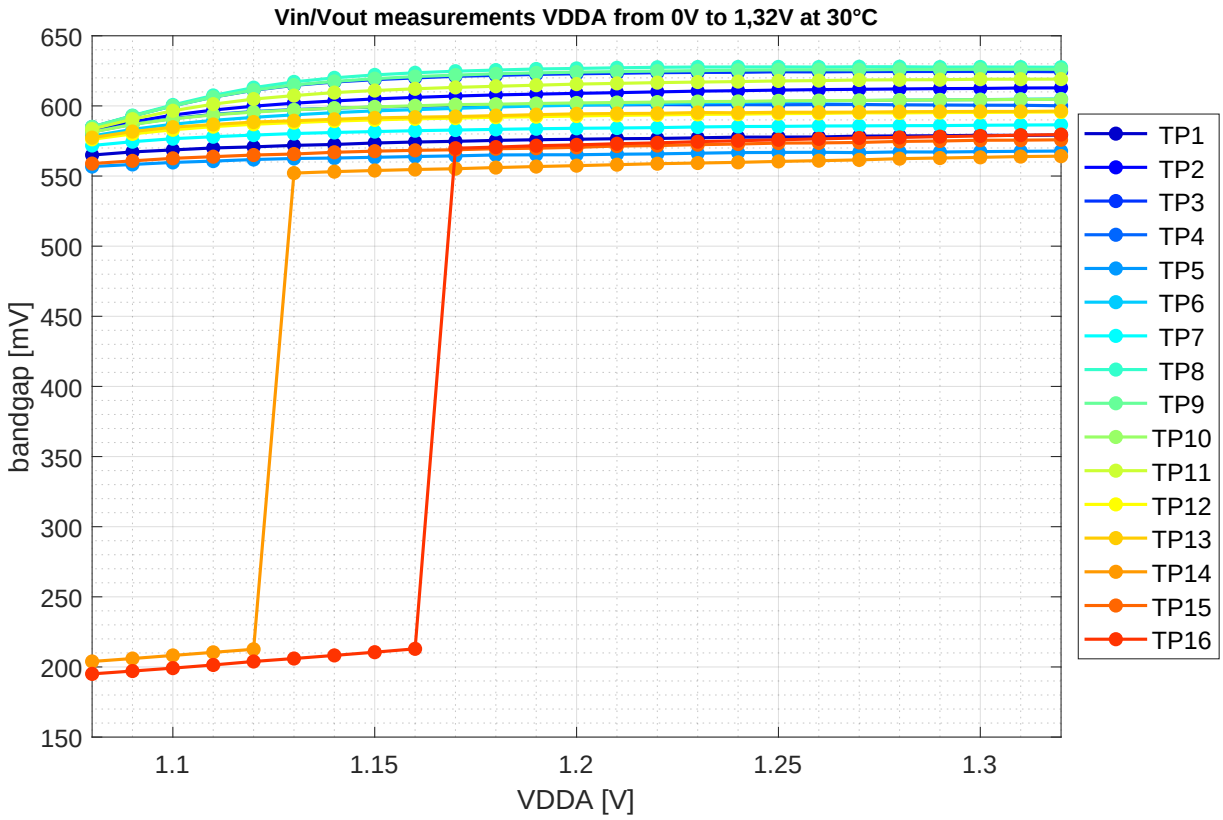


Figure 37: V_{in}/V_{out} curve of all TPs in operating range at 30 °C. Input voltage was increased from 0 V to 1.32 V in steps of 10 mV.

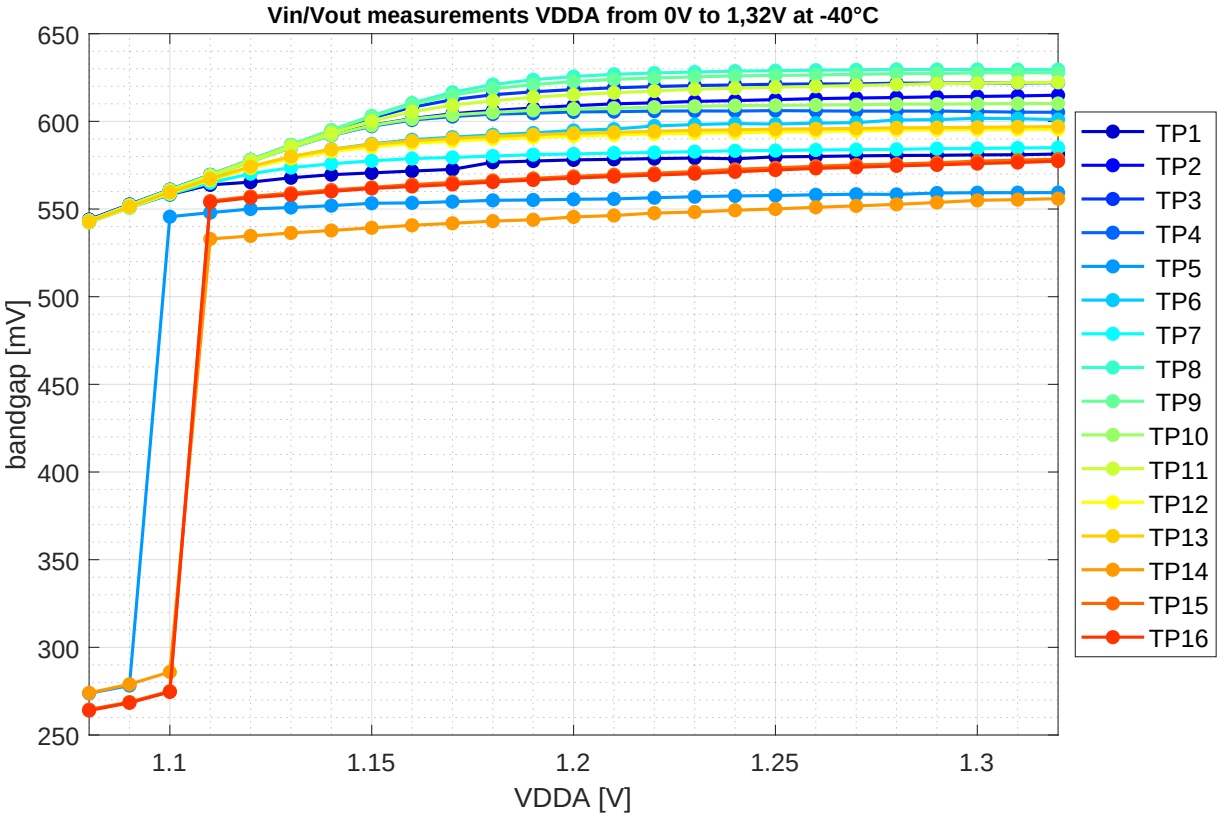


Figure 38: V_{in}/V_{out} curve of all TPs in operating range at –40 °C. Input voltage was increased from 0 V to 1.32 V in steps of 10 mV.

5.5.3 Linearity over input voltage

Because of the reasons mentioned in the previous section, it is assumed that the jumps do not occur in the characterization measurement, only the measurements from 1.32 V - 0 V are considered in the linearity analysis. Since the circuit is designed for input voltage fluctuations of ± 10 %, it must be examined whether the circuit can actually be operated in this range practically. For this purpose, the investigation of the linearity can be used again. This can be described with the linear regression or, more common for voltage sources, with the line regulation. First, I would like to discuss the study of linearity using linear regression.

As one can already assume from the two figures 40 and 41, the linearity of the BGR voltages is not given over the entire operating range. Objectively, this nonlinearity can again be numerically substantiated with the help of the *y-error*. This was first evaluated for the entire working range of 1.2 V±10 % and for each TP. The results show a deviation of the measured values from the regression line of up to more than 10 mV. This resulting *y-error* is too large for all TPs to describe the BGR voltages as a linear function of the input voltage. The working range of 1.2 V±10 % can therefore not be considered useful in reality.

Therefore, in the next step a range was defined in which the linearity of the output voltage to the input voltage is sufficiently given. Since the BGR voltages for input voltages above the operating

point of 1.2 V hardly change, only the lower limit was examined. For this purpose, the input voltage range was varied and the averaged absolute *y-errors* for each TP were considered. Starting from a lower limit of 1.15 V, an absolute averaged *y-error* of less than 4 mV was found for all TPs. Although this value is above the limit defined in section 3.2, an average value was considered here and, as can be easily seen, the measured values for the lower edge of the input voltage have a larger *y-error* than the rest and thus distort the average value. In figure 39, the *y-error* is plotted once for the entire operating range and once for the limited operating range for each TP.

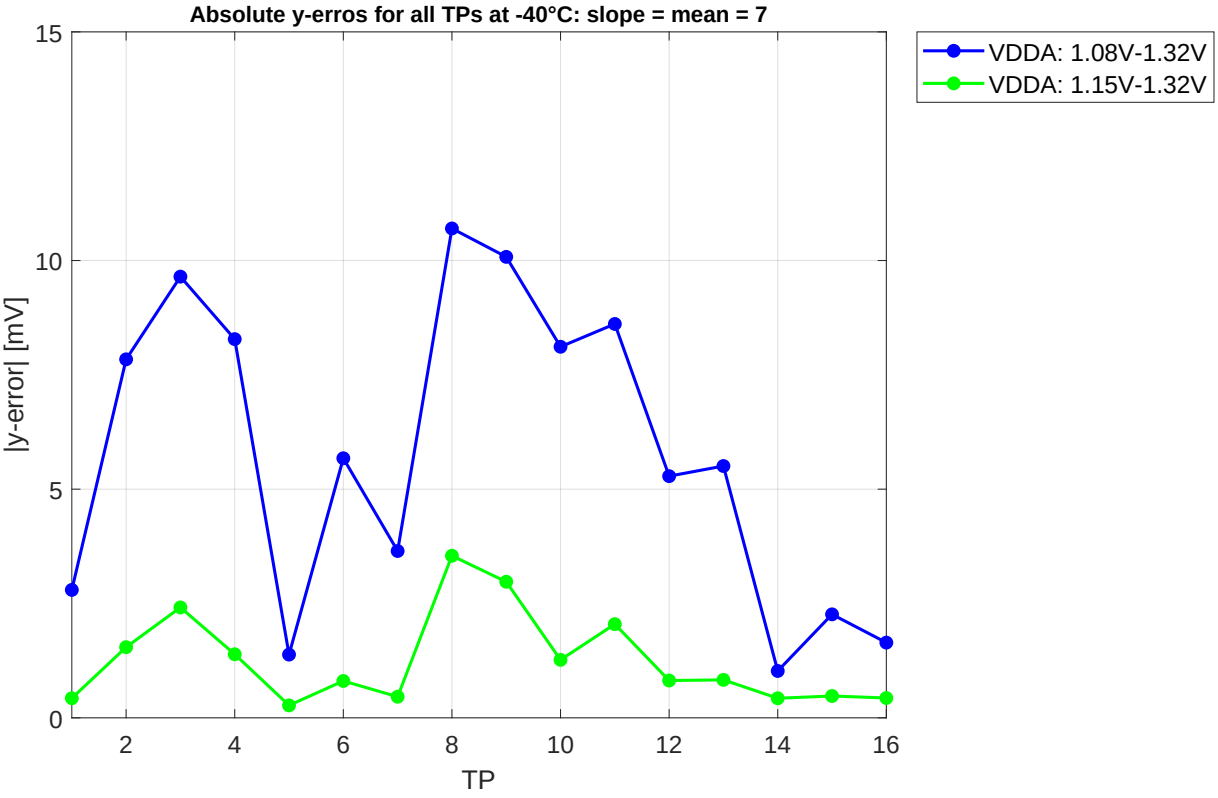


Figure 39: Absolute y-errors from V_{in}/V_{out} measurement for each TP. Graph shows different input ranges. The y-errors haven been averaged.

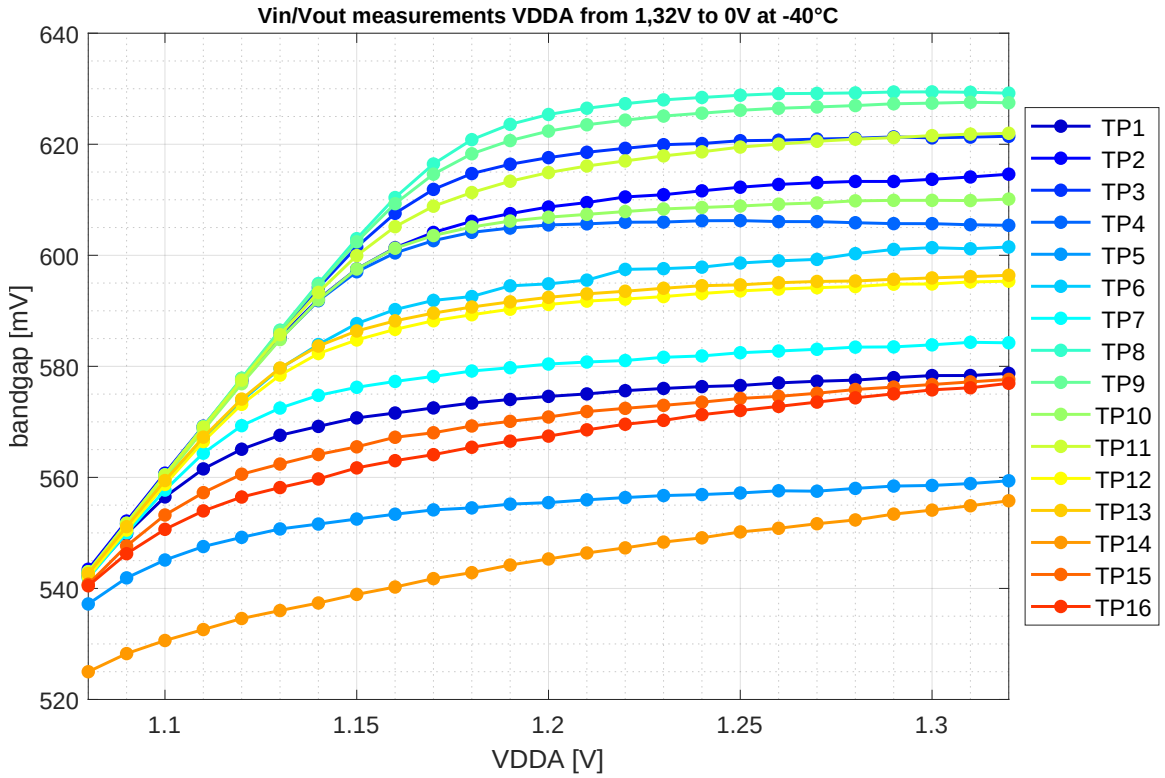


Figure 40: V_{in}/V_{out} curve of all TPs in operating range at –40 °C. Input voltage was decrease from 1.32 V to 0 V in steps of 10 mV.

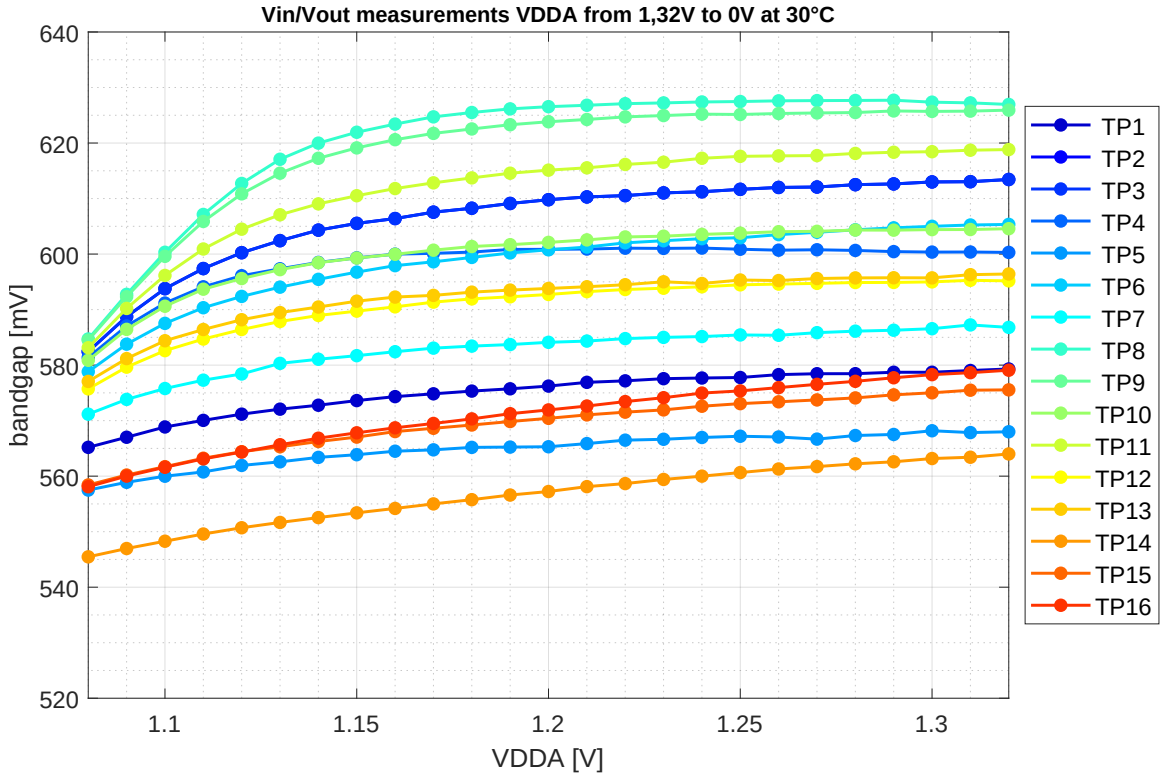


Figure 41: V_{in}/V_{out} curve of all TPs in operating range at 30 °C. Input voltage was decrease from 1.32 V to 0 V in steps of 10 mV.

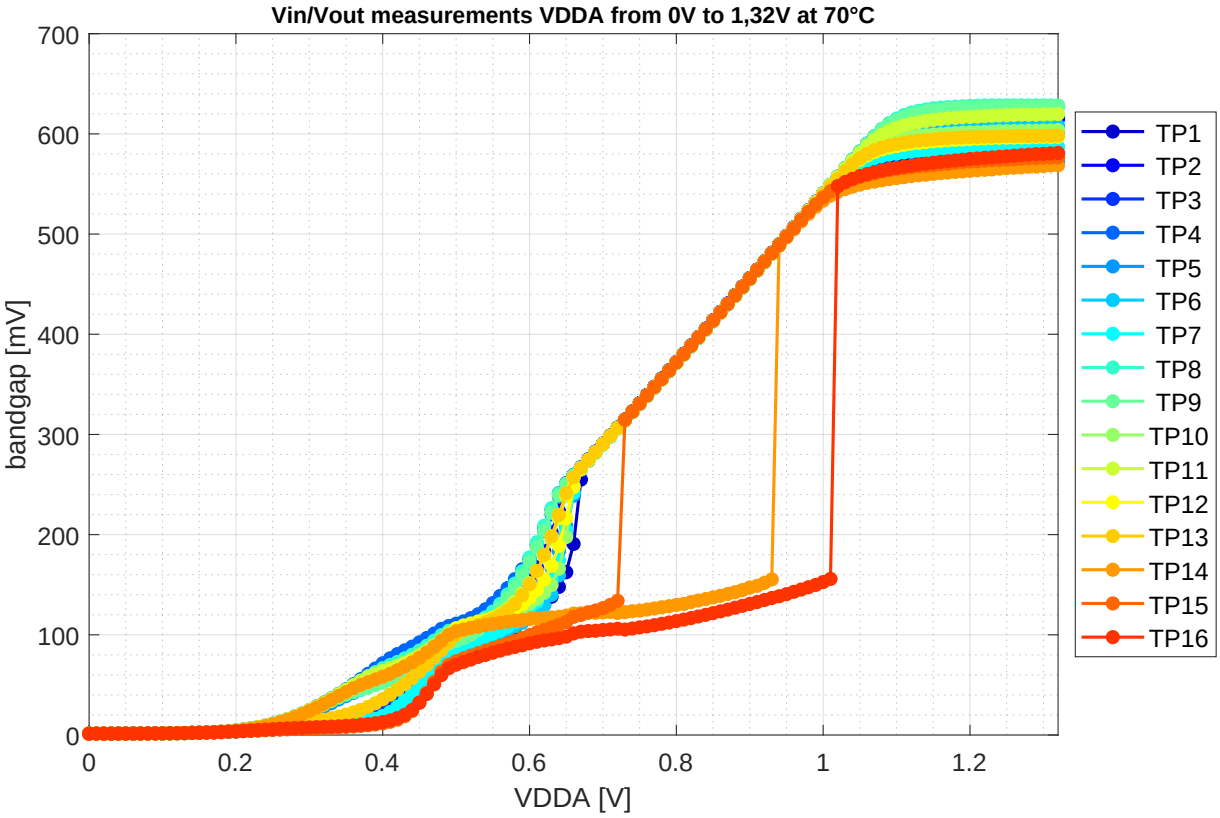


Figure 42: V_{in}/V_{out} curve of all TPs at 70 °C.

5.5.4 Line regulation

By looking at the line regulation, a statement about how the output voltage changes in relation to the input voltage per volt can be made. Also here only the measurements from 1.32 V - 0 V were used for the consideration of the line regulation. This was then determined as described in section 3.5 for all TPs and the temperatures –40 °C, 30 °C and 70 °C. Figure 43 shows the results for the entire operating range. It is noticeable that the line regulation at –40 °C degrees is almost 4 times as high as at 70 °C. This is due to the fact that, as already seen in section 5.5.3, the linearity decreases for lower temperatures.
If we now compare the course of the entire range with that of the limited range, which is shown in plot 44, we can see that the line regulation has approximately halved for all three temperatures. Especially for the positive temperatures, reasonable results could be achieved. Despite the halving of the line regulation, the course of the –40 °C line is still strikingly above the other two. If we compare the curves for the *y-error* and the line regulation, we see that they are similar in their behavior. The test points **TP3, TP8, TP9** and **TP11** show outliers in both plots. Which means that linearity for these TPs is worse with respect to the others. **TP4** and **TP5** on the other hand, show a strikingly positive behavior in both courses. For these TPs, a very high degree of linearity could be demonstrated.

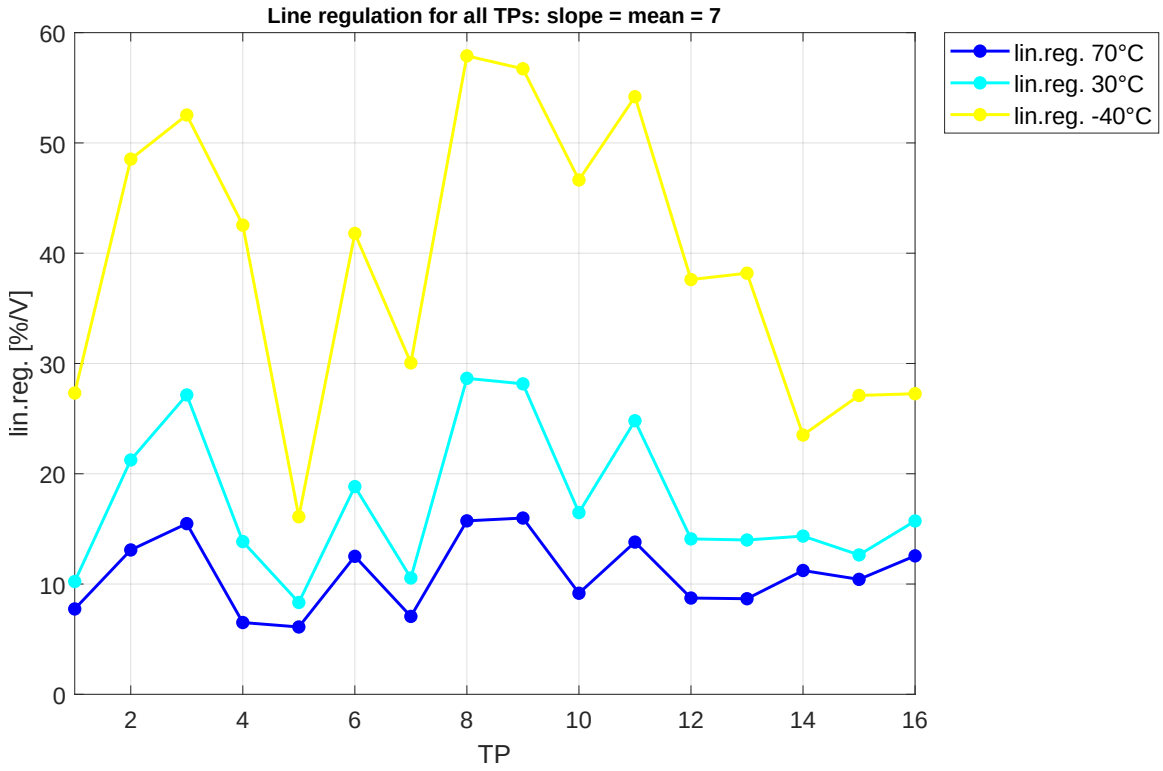


Figure 43: Line regulation of all TPs at different temperatures calculated for VDDA voltage range between 1.08 V to 1.32 V.

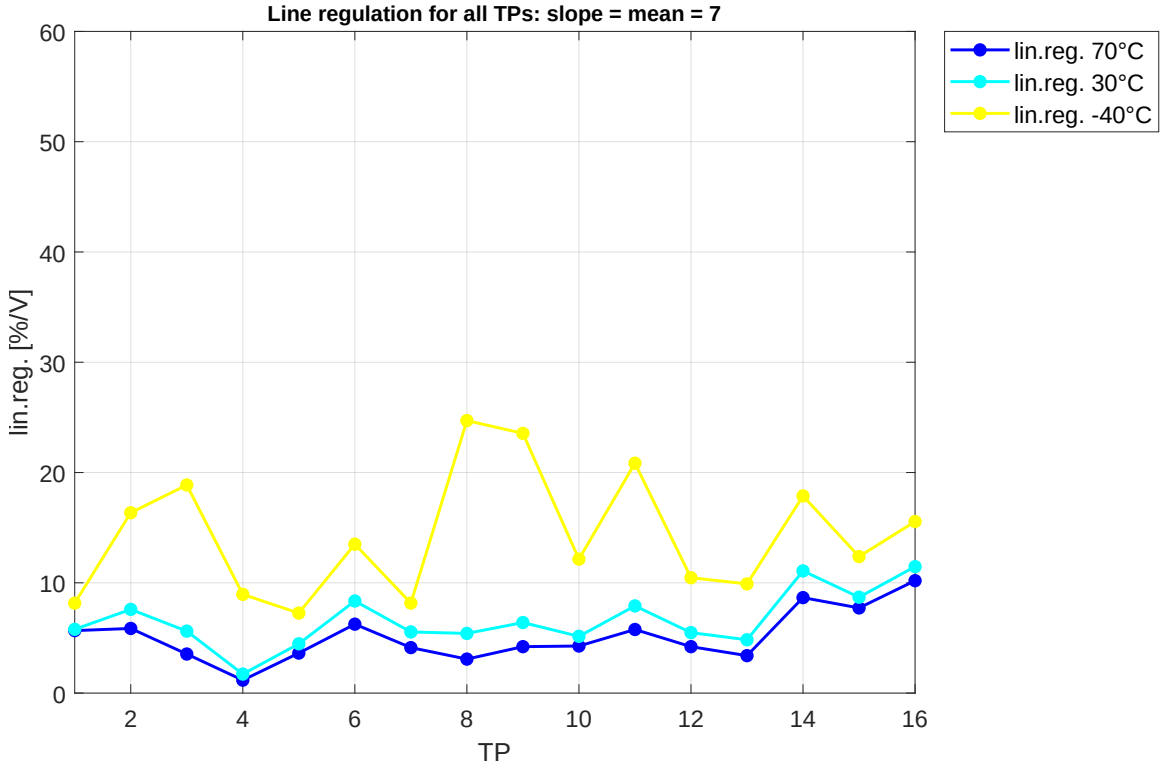


Figure 44: Line regulation of all TPs at different temperatures calculated for VDDA voltage range between 1.15 V to 1.32 V.

5.5.5 Effect on the configuration

It could already be proven in the previous sections that there is a clear correlation between the input voltage and the BGR voltages. In this part of the thesis I will now present the effect of different input voltages on the configuration found with Method 2 in section 5.4. The following three plots show the curves of all 16 TPs for their best configuration found with Method 2. The only difference in the plots are the input voltages. Figure 45 shows the curves at an input voltage of 1.2 V, which is the value they were calibrated at. Figure 46 shows the curves at 1.32 V and figure 47 shows the courses at 1.08 V input voltage.
Looking at the plots in Figure 46, one can see that the BGR voltages tend to have slightly higher values compared to 1.2 V, but in general their distribution does not deviate too much from the plot at 1.2 V input voltage. They are still in a range of about –5 mV to 15 mV.
The curves for the input voltage of 1.08 V, on the other hand, obviously have a nonlinear behavior. The BGR voltages scatter at approximately equal σ from 540 mV to 590 mV looking from –40 °C to 70 °C. A bandgap voltage with such a temperature curve is not suitable for use in high-energy physics applications. Therefore, the BGR voltages tested here can only be operated for input voltages in the range of:

$$1.15\,\mathrm{V} \leq \mathrm{VDDA} \leq 1.32\,\mathrm{V}$$

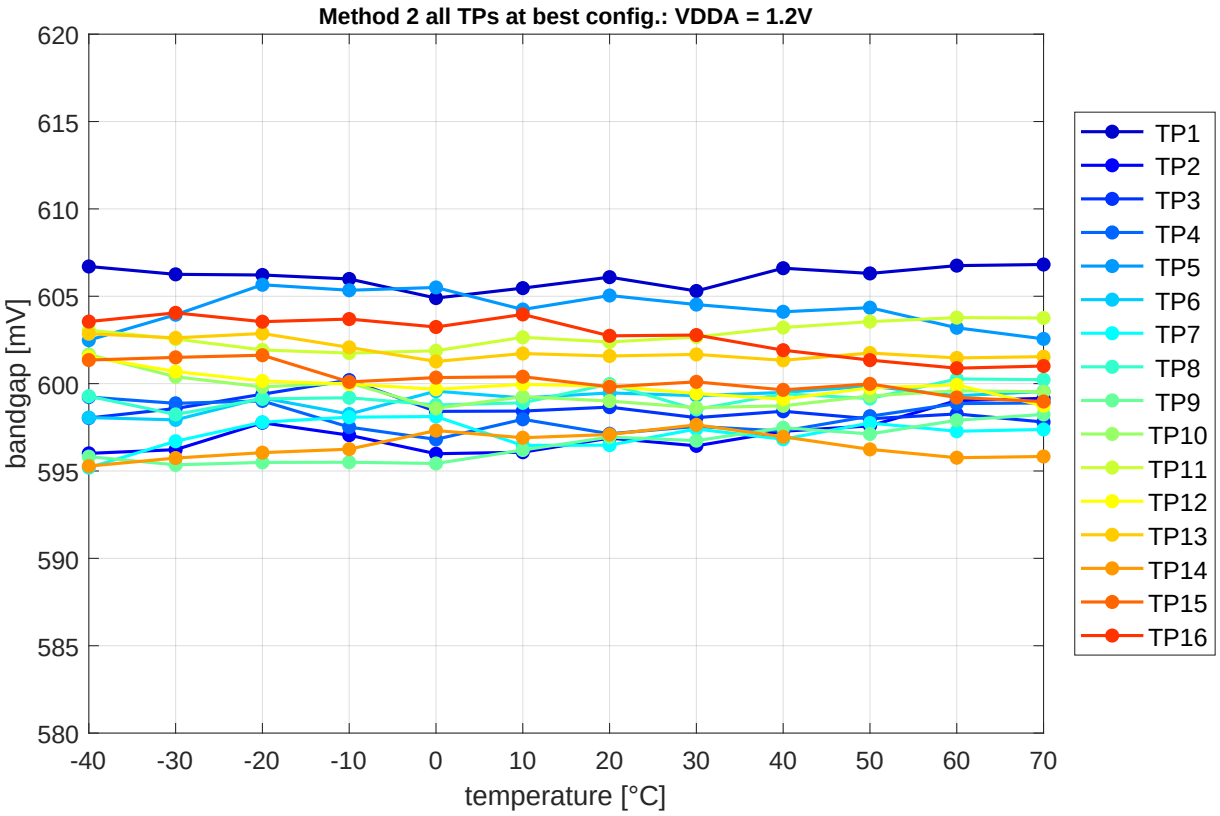


Figure 45: BGR voltages for best configuration found with Method 2 of each TP at VDDA 1.2 V.

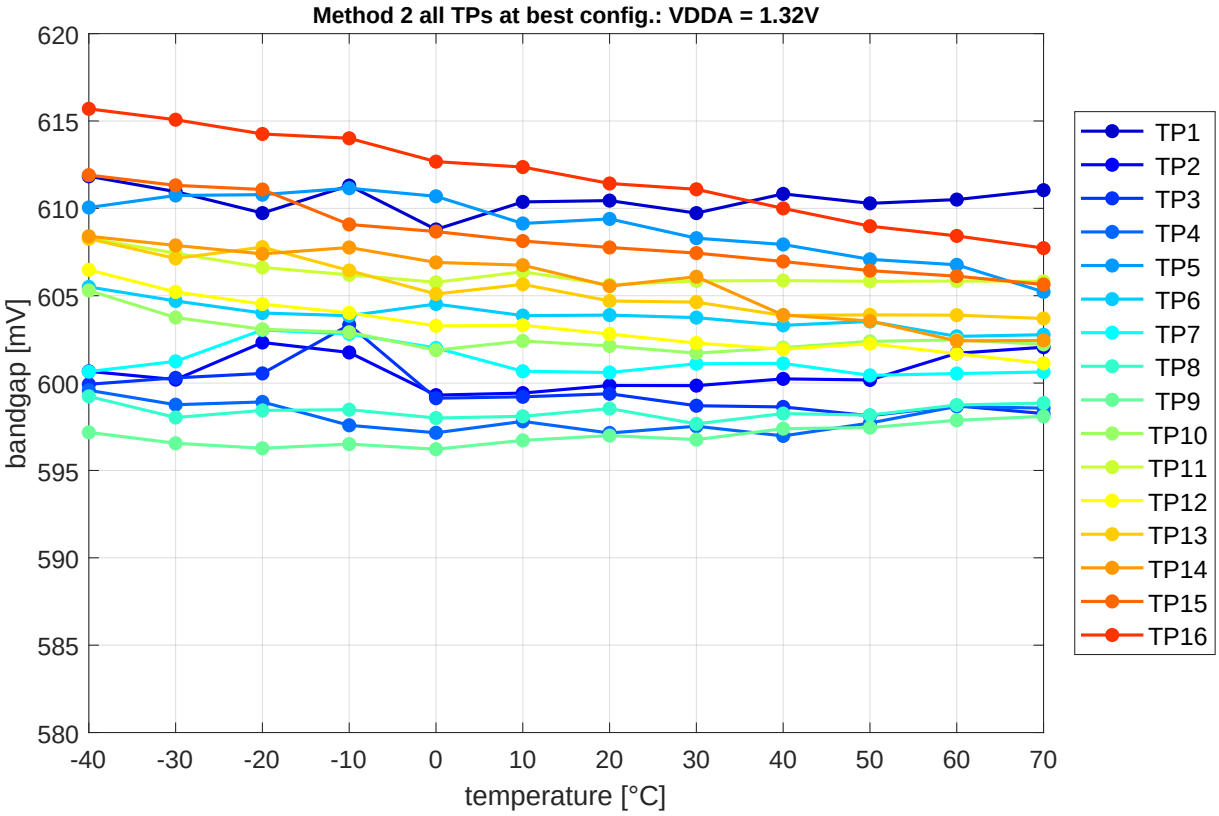


Figure 46: BGR voltages for best configuration found with Method 2 of each TP at VDDA 1.32 V.

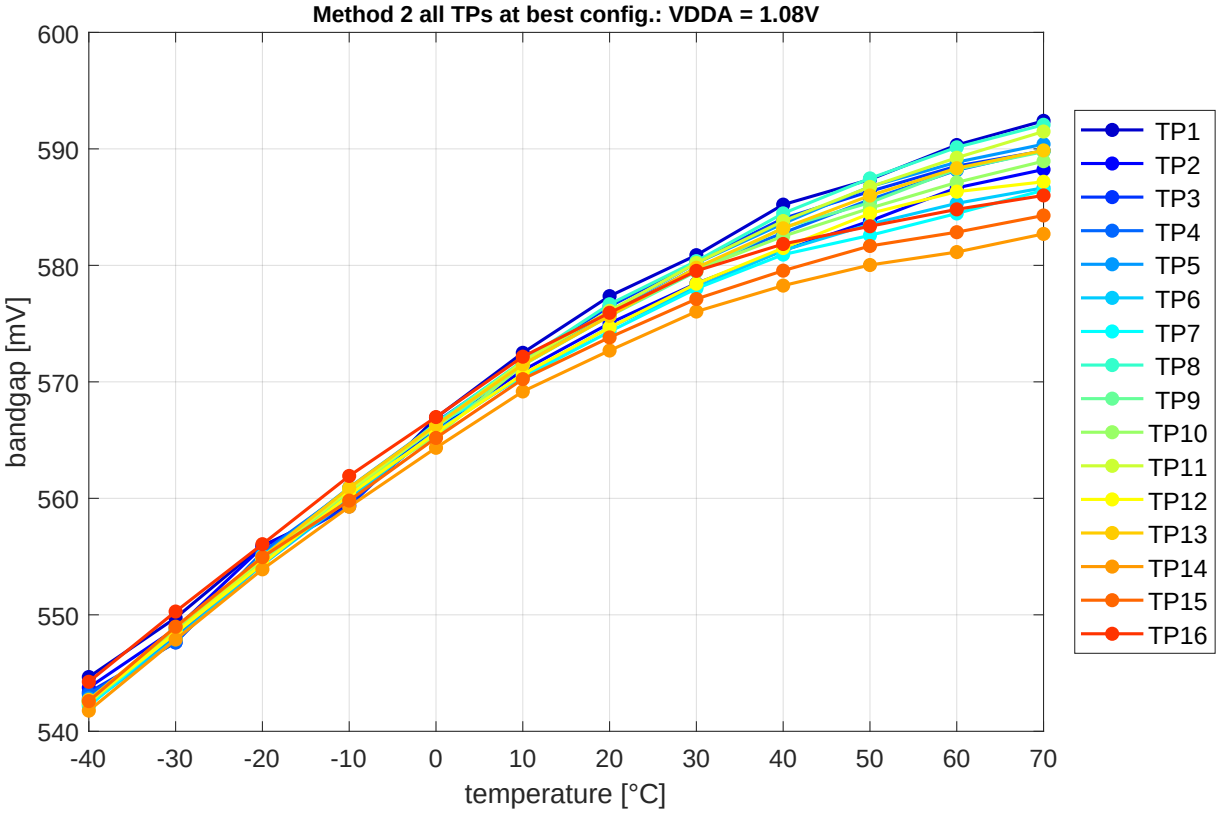


Figure 47: BGR voltages for best configuration found with Method 2 of each TP at VDDA 1.08 V.

6 Discussion

In this chapter, I would like to present the significance of the previously obtained results. Furthermore, I would like to compare the measurement results with the simulation results and analyze possible deviations and improvements.

6.1 The quality of trimming

In the evaluation of the trim quality, the following four parameters were evaluated in section 5.2:

- **range and location of BGR voltages**
- **step width of mean**
- **step width of slope**
- **cross dependencies of mean and slope**

In the following explanations, I would like to evaluate these parameters and make a statement about the quality of the trimming based on the results and set them in context with the configurations found by Method 2.

6.1.1 Range of trimming

When looking at the range, differences between the scatter of the minima and maxima became apparent. This low dispersion of the maxima indicates saturation effects. This effect has an impact on the range. For TPs with a high minimum value, the total range is up to 50 mV lower than for TPs with a lower minimum value. The range is therefore mainly dependent on the μ value of the TP. Regarding the position of the range, it can be summarized that the TPs with the largest deviation from the design point all have μ values below 600 mV. **TP5** and **TP14** are particularly noteworthy here, with an average of just ~560 mV. Despite the fluctuations in the range length, the measurement results are satisfactory. As shown in section 5.4, a configuration could be found for each TP that sufficiently meets the requirements. This also means that the 4-bit resistors provide a sufficient span of the output voltage. However, the position of the range should be adjusted in the final design. If we look at the best configuration for both methods, we see that all configurations except **TP4** are above and partly at the upper edge of the slope and mean settings. For **TP5** and **TP14**, both methods resulted in the same configuration of:

slope = 14 and mean = 15

For these test points the range had to be fully utilized to trim to the desired output voltage of 600 mV. However, this is not due to the span, but to the position of the μ of the two TPs.

6.1.2 Step width analysis

The step width analysis of the mean settings showed especially around the operating point in most configurations a sufficient setting range of about 9 mV to 17 mV per bit. However, the effect of saturation also appears here. For high mean values and low slope values, the step width is approximately 0. In this range, trimming is no longer possible. This is due to the fact that the maximum output voltage is limited by a hitherto unknown reason.
When analyzing the slope step width, the differences were not so obvious. To get a better overview I plotted the data from table 4 in figure 48. Here it can be stated that especially with the extreme settings for high slope values, the step width approaches 0. The same is true for high mean settings and low slope settings. In addition, it can be seen that high slope settings, for which the output voltage becomes maximally negative, the step widths become very small. Here, the temperature gradient can only be changed very slightly or not at all. If we now also look at the course of the BGR voltages (fig. 51a), we see that bit-wise changes in high slope settings cause a change in the mean values rather that in the temperature gradient.

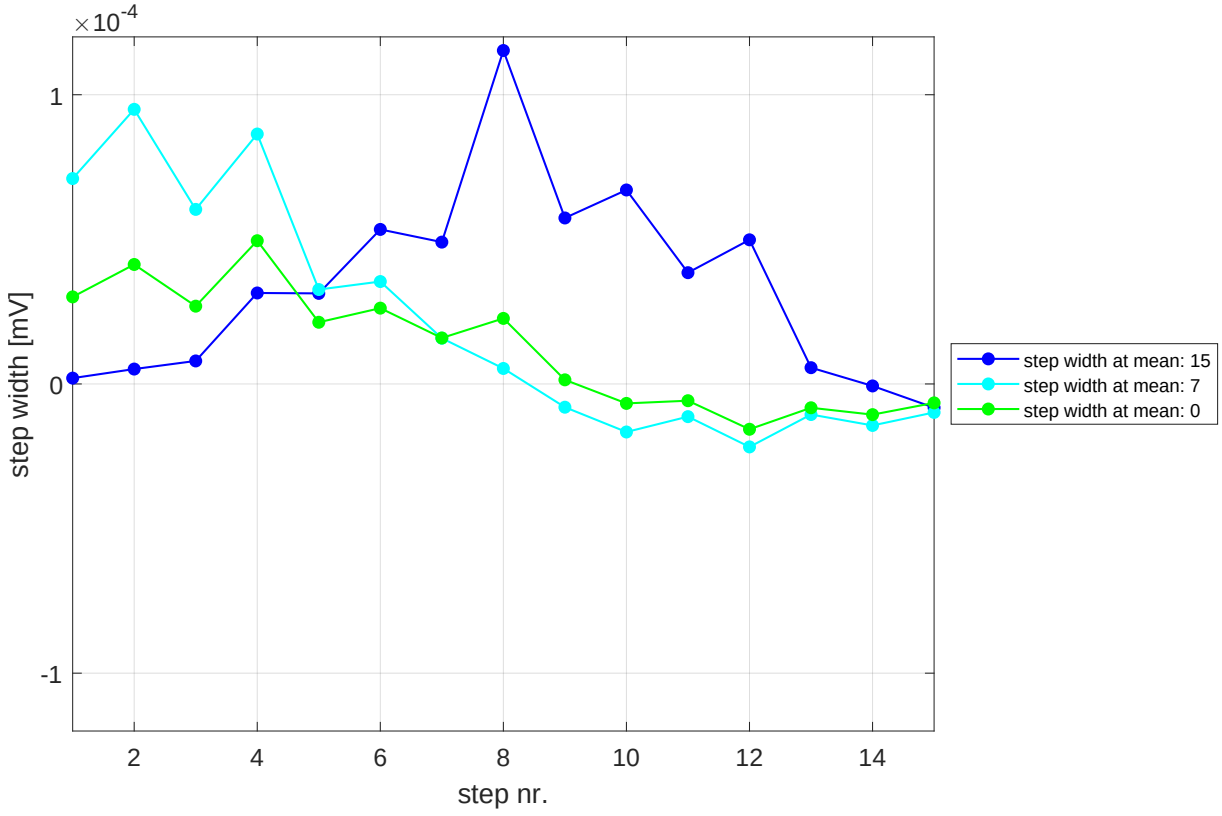


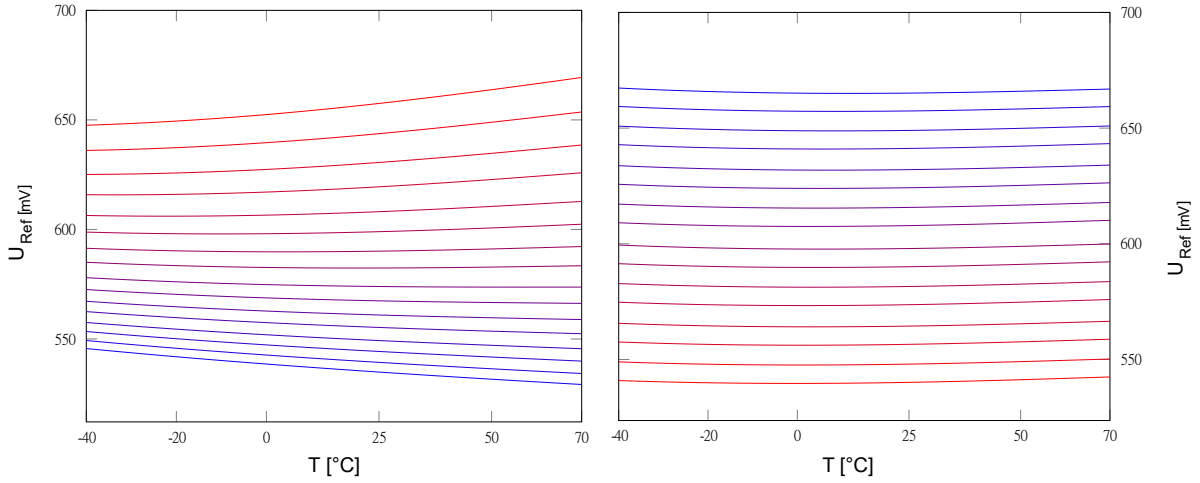
Figure 48: Difference between bit-wise slope changes in average for all TPs at different mean settings.

6.1.3 Dependency between mean and slope

The clear correlation between mean and slope could be proven for all TPs using the correlation coefficient. However, as can be seen in figure 49a, this correlation already exists in the simulation and is therefore most likely unavoidable. Based on the results presented in section 5.4, it can be stated that despite this correlation, a good compromise between accuracy for the design point and dependence on the temperature can be found.

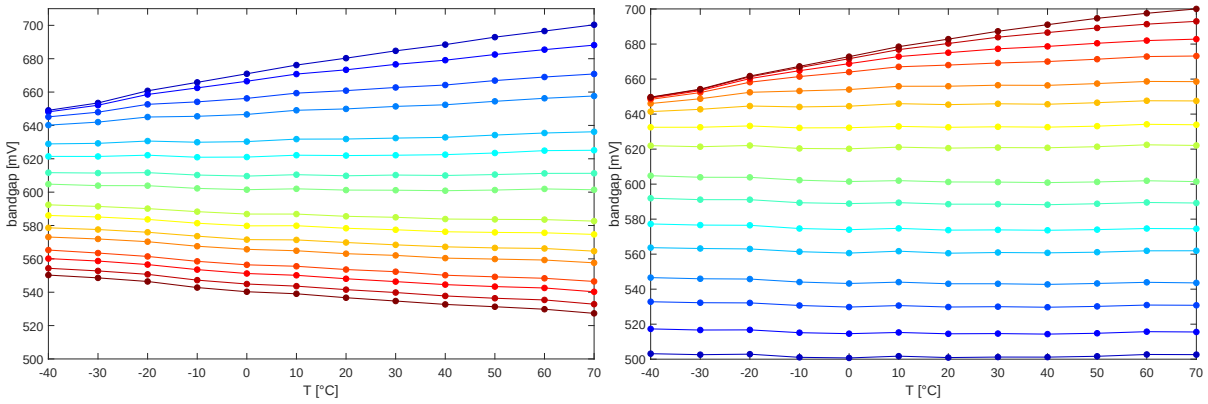
6.2 Linearity

If we compare the measurement results with the simulation results obtained in the dissertation of Pezzoli [11], we can see at first glance that the step sizes in the simulation are clearly more regular than in reality, both for the mean and for the slope settings. In addition, the saturation effect, which is clearly visible for low temperatures, high mean settings and low slope settings, does not appear in the simulations. This effect amplifies the nonlinear behavior especially for high mean settings, which can be clearly seen when comparing the two figures 49b and 50b. Although this effect occurs in most cases outside the operating point, it should be taken into account for the final design of the demonstrator.
Comparing the two figures 49a and 50a, it can be seen that the linearity for changes in the slope settings was predicted more poorly in the simulations than it actually is in the measurements. Looking only at the simulation for the different slope settings, it is noticeable that the dependence between slope and mean is already clearly visible here. In order to reduce this dependence, the design would have to be changed. If one would like to have a good differentiation of the settings for this temperature range, the pivot point of the curves in figure 49a would have to be at +15 °C. However, it is clearly in the negative area outside the temperature range.



(a) Simulation of BGR voltages for different slope values. (b) Simulation of BGR voltages for different mean values.

Figure 49: Bandgap voltage reference **simulations** made by Pezzoli [11] for the considered temperature range.

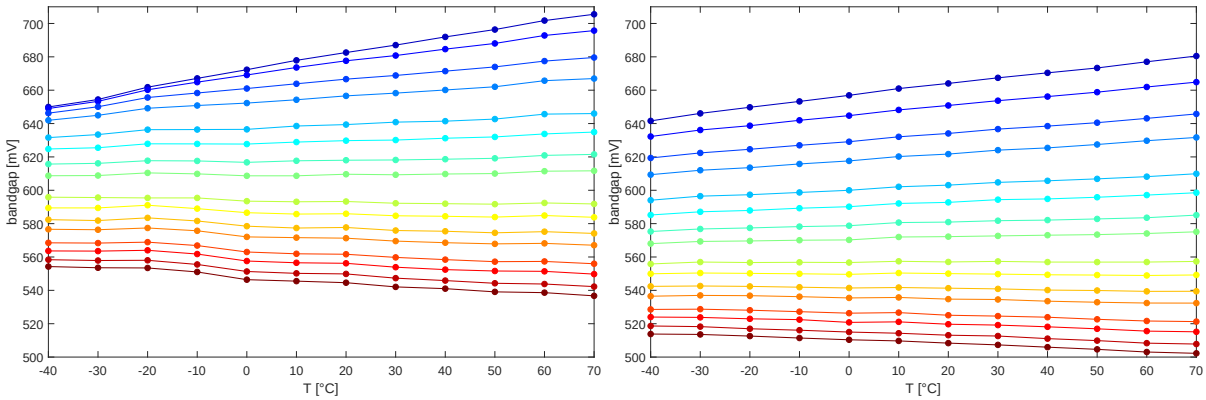


(a) Measurement of BGR voltages for different slope values. Color code: blue ≙ slope 0; dark red ≙ slope 15

(b) Measurement of BGR voltages for different slope values. Color code: blue ≙ mean 0; dark red ≙ mean 15

Figure 50: Bandgap voltage **measurements** of TP4. Results show the bandgap voltage output for the considered temperature range in picture (a) with **mean fixed** at decimal 7 and picture (b) **slope fixed** at decimal 7 for VDDA at 1.2 V

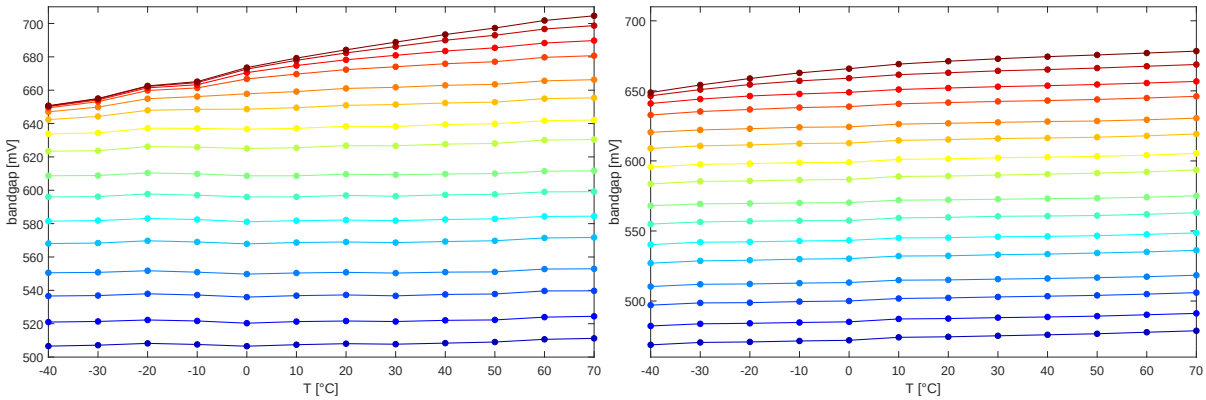
The following four figures show the BGR voltages of TP2 and TP16 for different slope values and different mean settings. These two TPs were selected because they showed very different qualities of linearity. In section 5.5.3, it was found that TP16 has a very high degree of linearity compared to TP2. This is especially visible when looking at the BGR voltages for different mean settings, shown in figures 52a and 52b. For both the slope and the mean settings, the step width are more pronounced for TP16 than for TP2, especially for low slope settings and high mean settings. In addition, it appears that the saturation effect, which increases for lower temperatures, is the cause of the nonlinearities.



(a) Measurement of BGR voltages for different slope settings of **TP2**. Color code: blue ≙ slope 0; dark red ≙ slope 15.

(b) Measurement of BGR voltages for different slope settings of **TP16**. Color code: blue ≙ slope 0; dark red ≙ slope 15.

Figure 51: Bandgap voltage measurements for different slope settings. Results show the bandgap voltage output for the considered temperature range in picture (a) **TP2** and picture (b) **TP16** with **mean fixed** at decimal 7 for VDDA at 1.2 V.



(a) Measurement of BGR voltages for different mean settings of **TP2**. Color code: blue ≙ mean 0; dark red ≙ mean 15.

(b) Measurement of BGR voltages for different mean settings of **TP16**. Color code: blue ≙ mean 0; dark red ≙ mean 15.

Figure 52: Bandgap voltage measurements for different mean settings. Results show the bandgap voltage output for the considered temperature range in picture (a) **TP2** and picture (b) **TP16** with **slope fixed** at decimal 7 for VDDA at 1.2 V.

6.3 Determine the best configuration

Before I go into the results for trimming, I would like to briefly explain the default configuration. So far, the circuit has been used without changing the trimming resistors, because it is not trivial to find an optimal setting point due to the correlation between mean and slope settings. In the standard configuration, both resistors are in the middle setting: slope = mean = 7. The distribution for all TPs in this configuration shows a μ of 595 mV with a σ of 22 mV, while the TC has a μ of 100 ppm/°C with a σ of 80 ppm/°C. The spread of the TC is therefore extraordinarily

high.
These values could already be significantly reduced by Method 1. The μ value of the BGR voltages is 599.3 mV with a σ of ~6 mV. Here the TC values scatter with a σ of 9 ppm/°C around the μ value ~35 ppm/°C. This is a clear improvement in all areas, especially in view of the temperature gradient.
With Method 2 the BGR voltages scatter with a sigma of ~3 mV exactly around the design point of 600 mV. The temperature response is slightly higher on average than it is at Method 1, but has a smaller deviation. In the simulations carried out by Pezzoli, a TC of 34.4 ppm/°C was determined. If one compares the TCs of the two methods with the simulation, one can see that they correspond very well. The measurements show very satisfactory results for this property. However, it is likely that these results can still be improved because, as already mentioned, a configuration at the upper limit of the setting was found for TP5 and TP14.

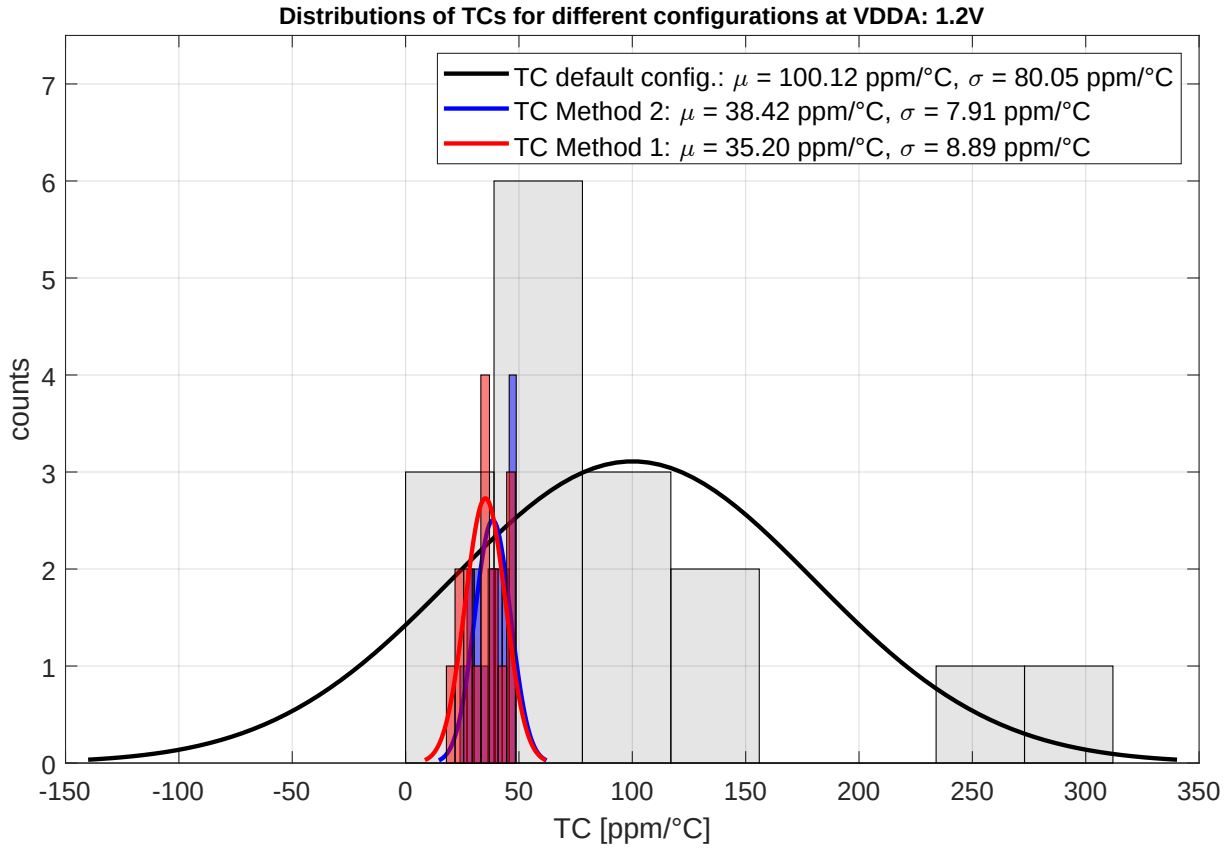


Figure 53: Distributions of |TC| for all TPs at configurations found by Method 1, Method 2 and default configuration (slope = mean = 7).

7 Summary

In this work, two methods for calibrating the BGR voltages could be found. One method, which gives better results in terms of the temperature coefficient, while the other gives a lower scatter of the BGR voltage. With both methods, the output voltage dispersion and the temperature response could be significantly improved compared to the default settings without trimming. On the basis of the measurements, it could be shown that with the help of the selected 4-bit resistors it is possible to trim the BGR voltages both in their voltage level and in their temperature gradient. For the final design of the board, however, some properties should be adapted.

Above all, the saturation effects should be investigated beforehand. These have a decisive influence on the quality of the trim. This results in an unwanted upper limit, which is reduced even further for low temperatures. This affects the trim range and the step width, especially the mean step width. The position of the μ values of the range was sufficiently close to the working point for almost all TPs. Except for the two outliers TP5 and TP14. Here it should be investigated whether this is a specimen scattering or whether there are other deviations.

In addition, nonlinearities were detected for very high slope and mean settings. These should be avoided in this application, as the temperature behavior should be as linear as possible. However, the effect could only be detected for BGR voltages far outside the operating point and outside the σ values found with methods 1 and 2. It therefore hardly affects the trim for 600 mV, but limits the total trim range, as these configurations are unusable.
The degree of linearity proved to be extraordinarily high, especially around the operating point. Which was also reflected in the TCs. They lie in the range of < 40 ppm/°C. This means that the BGR voltages change in average by less than 4 mV over the entire temperature range of 110 °C.

However, the dependence of the BGR voltages on the input voltage was found to be critical. In particular, the range in which the circuit should actually behave approximately linearly is limited. This is even amplified at low temperatures. The stability of the circuit to a fluctuation of the input voltage of ±10 % was disproved in the measurements and only proven for 1.2 V −4 % +10 %.
The line regulation for the limited range gave satisfactory results, but could not confirm the results of the simulation. It is difficult to make a comparison here, as the simulation does not provide any information about the temperature for which the line regulation was determined. However, even the best results at 70 °C exceed the simulation results by a factor of 2.

The occurrence of the jumps could be detected during the V_{in}/V_{out} measurements for different input voltages. However, since this is a different type of measurement from the characterization, it must be analyzed whether these effects also occur in other measurements. If this is the case, the range in which the circuit can be operated would be significantly limited especially the

lower input limit of the circuit. As it was shown in section 5.5.5 the BGR voltages can not be usefully operated for an input voltage of 1.08 V. This was not foreseen in the design and has to be investigated in further measurements. Based on the measurements performed, this effect can not be explained certainty. However, it is likely that the startup circuit plays a role here, as it enters the second stable state at a certain input voltage.
With the help of these results, the final board layout for the demonstrator can be improved.

8 Future work

In further investigations, the cause of the saturation effect should be determined and remedied. Only then the 4-bit resistors can be operated usefully over their entire range. In addition, it should be investigated whether the physical properties of the resistors themselves influence the temperature dependence.

Furthermore, the board must be exposed to an irradiation of 250 Mrad and then further characterization must be carried out to determine how well the circuit reacts to high irradiation, as occurs in high-energy physics.

In addition, with Method 2, the lower limit for TC would have to be made dynamic in further calibrations, as otherwise it can happen that too few configurations remain after the first sorting.

Also the relationship between input voltage and output voltage should be investigated more closely in further measurements. Since no LDO is used for voltage regulation in the later design, the circuit must be stable for fluctuations in the input voltage. However, this is not the case in the current design.

Bibliography

[1] R. A. Blauschild et al. “A new NMOS temperature-stable voltage reference”. In: *IEEE Journal of Solid-State Circuits* 13.6 (1978), pp. 767–774. ISSN: 0018-9200. DOI: 10.1109/JSSC.1978.1052048.

[2] The CMS Collaboration et al. “The CMS experiment at the CERN LHC”. In: *Journal of Instrumentation* 3.08 (2008), S08004–S08004. DOI: 10.1088/1748-0221/3/08/S08004.

[3] Holger Göbel. *Einführung in die Halbleiter-Schaltungstechnik*. 6. Aufl. 2019. Berlin, Heidelberg: Springer Berlin Heidelberg, 2018. ISBN: 978-3-662-56563-6. URL: http://nbn-resolving.org/urn:nbn:de:bsz:31-epflicht-1543175.

[4] Michael Hauschild. *Neustart des LHC: CERN und die Beschleuniger: Die Weltmaschine anschaulich erklärt*. essentials. Wiesbaden: Springer Spektrum, 2016. ISBN: 978-3-658-13479-2. DOI: 10.1007/978-3-658-13479-2.

[5] Ekbert Hering, Klaus Bressler, and Jürgen Gutekunst, eds. *Elektronik für Ingenieure und Naturwissenschaftler*. 7., aktualisierte und verbesserte Auflage. Berlin and Heidelberg: Springer Vieweg, 2017. ISBN: 978-3-662-54214-9. DOI: 10.1007/978-3-662-54214-9.

[6] Xiangliang Jin and Bingjun Xiong. “Low-Voltage Bandgap Reference Based on Deep Submicron Technology”. In: *2020 7th International Conference on Information Science and Control Engineering (ICISCE)*. IEEE, 2020, pp. 1923–1927. ISBN: 978-1-7281-6406-9. DOI: 10.1109/ICISCE50968.2020.00378.

[7] Yoshihiro Kaneko, ed. *ITC-CSCC 2020: The 35th International Technical Conference on Circuits/Systems, Computers and Communications : Fri, July 3rd, 2020-Mon, July 6th, 2020, WINC Aichi, Aichi Industry & Labor Center, Nagoya, Japan (online)*. Piscataway, NJ: IEEE, 2020. ISBN: 978-4-88552-328-1. URL: https://ieeexplore.ieee.org/servlet/opac?punumber=9179648.

[8] Rafal Kleczek and Pawel Grybos. “Low voltage area efficient current-mode CMOS bandgap reference in deep submicron technology”. In: *2014 Proceedings of the 21st International Conference Mixed Design of Integrated Circuits and Systems (MIXDES)*. IEEE, 2014, pp. 247–251. ISBN: 978-83-63578-05-3. DOI: 10.1109/MIXDES.2014.6872194.

[9] Wenguan Li, Ruohe Yao, and Lifang Guo. “A low power CMOS bandgap voltage reference with enhanced power supply rejection”. In: *2009 IEEE 8th International Conference on ASIC*. IEEE, 2009, pp. 300–304. ISBN: 978-1-4244-3869-3. DOI: 10.1109/ASICON.2009.5351450.

[10] Shivang Patel and Amisha Naik. “Design of Start-up Enabled Bandgap Voltage Reference”. In: *2022 6th International Conference on Devices, Circuits and Systems (ICDCS)*. IEEE, 2022, pp. 18–22. ISBN: 978-1-6654-8094-9. DOI: 10.1109/ICDCS54290.2022.9780858.

[11] Matteo Pezzoli. *Development of IP blocks in a 110 nm CMOS technology for monolithic pixel sensors: Dissertation*. Università degli Studi di Pavia, Department of Electrical, Computer and Biomedical Engineering, 2021.

[12] Behzad Razavi. *Design of analog CMOS integrated circuits*. Second editon. New York: McGraw Hill Education, 2016. ISBN: 978-0-07-252493-2. URL: https://ebookcentral.proquest.com/lib/kxp/detail.action?docID=6327694.

[13] Peter Ruge, Carolin Birk, and Manfred Wermuth. *Das Ingenieurwissen: Mathematik und Statistik*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013. ISBN: 978-3-642-40474-0. URL: http://nbn-resolving.org/urn:nbn:de:bsz:31-epflicht-1603379.

[14] G. Traversi et al. “Characterization of bandgap reference circuits designed for high energy physics applications”. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 824 (2016), pp. 371–373. ISSN: 01689002. DOI: 10.1016/j.nima.2015.09.103.

[15] Y. P. Tsividis and R. W. Ulmer. “A CMOS voltage reference”. In: *IEEE Journal of Solid-State Circuits* 13.6 (1978), pp. 774–778. ISSN: 0018-9200. DOI: 10.1109/JSSC.1978.1052049.

[16] Tommaso Vergine et al. “A 65 nm Rad-Hard Bandgap Voltage Reference for LHC Environment”. In: *IEEE Transactions on Nuclear Science* 63.3 (2016), pp. 1762–1767. ISSN: 0018-9499. DOI: 10.1109/TNS.2016.2550581.

List of Figures

List of Tables

List of Code

List of Abbreviations

ADC	Analog-digital converter
arcadia	Advanced Readout CMOS Architectures with Depleted Integrated sensor Arrays
BGR	Bandgap voltage reference
BJT	bipolar junction transistor
CERN	Conseil européen pour la recherche nucléaire
CMOS	Complementary metal–oxide–semiconductor
CSV	comma-separated values
CTAT	Complementary to Absolute Temperature Component
DAC	Digital-analog converter
DC	direct current
DRAM	Dynamic random-access memory
FPGA	field-programmable gate array
IC	integrated circuit
INFN	Istituto Nazionale di Fisica Nucleare (National Institute of Nuclear Physics)
LDO	Low-dropout regulator
MOSFET	metal–oxide–semiconductor field-effect transistor
op amp	operational amplifier
PLL	phase-locked loop
PTAT	Proportional to Absolute Temperature Component
TC	Temperature coefficient
TP	test point

A ARCADIA

The arcadia project aims to develop an alternative to the widely used hybrid pixel detectors. Here, sensor and readout electronics are produced separately from each other and later brought together via bump-bonding. A new approach has been found for so-called *Monolithic CMOS sensors*. These sensors are primarily intended to reduce costs and material consumption. They are used especially in high-energy physics. However, there has been little research in this area so far. The large scale demonstrator is intended to show prove of concept for Monolithic CMOS sensors used for high-energy physics applications. The main challenges are the radiation hardness and a very short readout time. The whole sensor should also consume as little power as possible. These sensors will be used primarily for physics experiments at CERN.

B CERN: a brief introduction

Since the middle of the 20th century exists the idea to build a particle accelerator to gain conclusions on theories how our universe works. Since Europe consists of many rather small countries, it was decided after the 2nd World War to work together on this project. This idea gave birth to *Conseil Européen pour la Recherche Nucléaire* (CERN), which at that time had as its motto "Research for Peace". It has set itself the goal to build the largest particle accelerator in the world and thus to become the center for basic research. The newest and most powerful accelerator to date is called the High Luminosity Large Hadron Collider (HL-LHC). The HL-LHC is the part of the upgrade from the Large Hadron Collider (LHC). It is one of the so-called circular accelerators and makes it possible to generate collisions of very high-energy in a relatively small space. The HL-LHC consists of four main experiments: CMS, LHCb, ATLAS and ALICE. Each of this experiments has its own technology and research area. Therefore, this work deals exclusively with **Compact Muon Solenoid** experiment (CMS). Despite enormous challenges, this facility was able to provide proof of the existence of the Higgs Boson as early as 2012. At that time, the accelerator was operated at just half its maximum energy. However, in order to provide further evidence, the center of mass energy of the accelerator must be significantly increased. This entails a number of improvements, which will be performed in the so-called *Phase 2 upgrade*. [4]

B.1 The Phase 2 upgrade

With the *Phase 2 upgrade* which will be implemented in 2024 the LHC will change its name to High Luminosity-LHC (HL-LHC). The name change is based on the fact that the number of collisions is directly related to the luminosity. Thus, if one increases the luminosity, the number of collisions also increases. The *Phase 2 upgrade* will be completed by 2026 with a proton-proton collision energy of 14 TeV and an integrated luminosity of 3000 fb^{-1}. [4]
At the designed luminosity, the detectors therefore observe an event rate of approximately 10^9 inelastic events/s. This leads to a number of tremendous experimental challenges. The event selection process (trigger) must reduce the rate to about 100 events/s for storage and subsequent analysis. The very short time span between bundle transitions of 25 ns, has significant implications for the design of the readout and trigger systems. [2]

C Measurement programs

```
# Author: Yannick Konstandin
# University of Bergamo
# June/2022
# bandgap voltage
# set power-supply
# read output from multimeter
# plot data

import visa
import pyvisa
import csv
import time
import numpy as np
from ctypes import util
from agilent_34461A import Agilent34461A
from agilent_34461A import Power_supply
import matplotlib.pyplot as plt
import pandas as pd
import decimal
from tkinter import *
import tkinter.messagebox

##############################################
# this must be changes manually
TP = 1      # testpoint
tmp = 30   # temperature
VCC = 2.5   # VCCIN
##############################################

# Pop-up
top = Tk()
top.geometry(f"200x50+{800}+{400}") # size and position of window
tkinter.messagebox.showinfo('Slope and mean set to 7?','Starting measurement of TP'+str(TP)+' at '+str(tmp)+'C.')
top.destroy()

plt.close("all")
rm = pyvisa.ResourceManager('@py')
ps_agilent = rm.open_resource('USB0::2391::36632::MY52350174::0::INSTR')
agilent = Agilent34461A(rm)

#volt = agilent.get_voltage()
```

```
#print('Voltage:', volt, 'V')

header = ['set voltage', 'bandgap voltage','Temperature']

# define voltage range and steps
volt_in = np.arange(0,1.33,0.01)
#volt_in = np.arange(1.32,-0.01, -0.01)
results = pd.DataFrame()

for voltage in volt_in:
  ps_agilent.write('VOLT %1.2f' % voltage)
  ps_agilent.write('SENS:CURR:PROT 2') #compliance in A
  time.sleep(0.5)
  print('Vin: %1.3f' % voltage)
  vin_real = float(ps_agilent.query('MEAS:VOLT:DC?'))
  print('Vin real: %1.3f' % vin_real)
  voltage_dec = decimal.Decimal(voltage)

  temp = {}
  temp['Vin'] = float(voltage)
  temp['Vin_real'] = vin_real
  temp['Vout'] = agilent.get_voltage_rang_1()
  temp['temp'] = tmp
  temp['VCC']  = VCC
  results = results.append(temp, ignore_index=True)

if tmp < 0:
    temper = tmp*-1
    results.to_csv('/home/microlab/Documenti/Python_scripts/Band_gap/VIN_VOUT/
                    Results_TP'+str(TP)+'_TEMP_m'+str(temper)+'_0_UP.csv')
else:
    results.to_csv('/home/microlab/Documenti/Python_scripts/Band_gap/VIN_VOUT/
                    Results_TP'+str(TP)+'_TEMP_'+str(tmp)+'_0_UP.csv')

#x = np.linspace(0, 1.4, 1)
plt.plot(results['Vin'],results['Vout'], 'ro')
plt.xlabel('Vin [V]')
plt.ylabel('bandgap [V]')
plt.title('Bandgap measurement')
plt.grid(b=True, which='major', color='grey', linestyle='--')
#change this
plt.legend(['TP'+str(TP)+' at '+str(tmp)+'C'])
ps_agilent.write('SYST:BEEP 1000,0.7')
plt.show()
```

Code 3: Measurement program Vin/Vout

```
# Author: Yannick Konstandin
# University of Bergamo
# June/2022
# bandgap voltage
# set power-supply
# set registers
# read output from multimeter
# plot data

import logging
import time
from random import random
from math import floor
from tabulate import tabulate
from pyarcadia.test import Test
from pyarcadia.sequence import SubSequence
from pyarcadia.data import Pixel
import sys
import visa
import pyvisa
import csv
import time
import numpy as np
from ctypes import util
from agilent_34461A import Agilent34461A
from agilent_34461A import Power_supply
import matplotlib.pyplot as plt
import pandas as pd
import decimal
from tkinter import *
import tkinter.messagebox


####################################
#must be changed manually
TP = 1      # set testpoint
tmp = 70    # set temperature
####################################

plt.close("all")
rm = pyvisa.ResourceManager('@py')
ps_agilent = rm.open_resource('USB0::2391::36632::MY52350174::0::INSTR')
agilent = Agilent34461A(rm)
x = Test()

#set output of PS
ps_agilent.write('VOLT 1.2')
ps_agilent.write('SENS:CURR:PROT 2') #compliance in A
#print('Voltage:', volt, 'V')
```

```
50
51 # define voltage range and steps
52 word_bits = np.arange(0,16,1)
53 mean = np.arange(0,16,1)
54 #slope = np.arange(15,-1,-1)
55 slope = np.arange(0,16,1)
56 ps_volts = np.array([1.08, 1.2, 1.32])
57 i = np.arange(0, 3, 1)
58
59 results = pd.DataFrame()
60
61 # Pop-up
62 top = Tk()
63 top.geometry(f"200x10+{800}+{400}") # size and position of window
64 tkinter.messagebox.showinfo('Bandgap characterization','Starting measurement of
      TP'+str(TP)+' at '+str(tmp)+'C.')
65 top.destroy()
66 #top.mainloop()
67
68 register = TP -1
69 register_mean = 'BIAS'+str(register)+'_BGR_MEAN'
70 register_slope = 'BIAS'+str(register)+'_BGR_SLOPE'
71
72 if tmp < 0:
73     temper = tmp*-1
74     filename =
          '/home/microlab/Documenti/Python_scripts/Band_gap/final_characterization/
75 ____Results_TP'+str(TP)+'_REG_m'+str(temper)+'.csv'
76 else:
77     filename =
          '/home/microlab/Documenti/Python_scripts/Band_gap/final_characterization/
78 ____Results_TP'+str(TP)+'_REG_'+str(tmp)+'.csv'
79
80 for n in i:
81     PS_VOLT = ps_volts[n]
82     ps_agilent.write('VOLT %1.2f' % PS_VOLT)
83     for slopes in slope:
84         # set slope value
85         x.chip.write_gcrpar(register_slope, slopes)
86         slope0 = x.chip.read_gcrpar(register_slope)
87         print('slope: '+str(slope0)+'')
88
89         for word in mean:
90             # set mean value
91             x.chip.write_gcrpar(register_mean, word)
92             mean0 = x.chip.read_gcrpar(register_mean)
93             print('slope:', slope0,'mean:', mean0)
94             time.sleep(0.01)
95             #readback voltage of PS
96             vin_real = float(ps_agilent.query('MEAS:VOLT:DC?'))
```

```
            Iin_real = float(ps_agilent.query('MEAS:CURR:DC?'))
            print('PS Iout measure: %1.4f' %Iin_real)
            print('PS Vout set to: %1.4f' %PS_VOLT)
            print('Vin real: %1.4f' % vin_real)
            #read voltage of TP
            volt = agilent.get_voltage_rang_1()
            temp = {}
            temp['TP'] = TP
            temp['reg'] = register
            temp['temp'] = tmp
            temp['Vin_real'] = vin_real
            temp['Vin'] = PS_VOLT
            temp['SLOPE'] = slope0
            temp['MEAN'] = mean0
            temp['Volt'] = volt
            results = results.append(temp, ignore_index=True)

results.to_csv(filename)
plt.plot(results['SLOPE'],results['Volt'], 'b.')
plt.xlabel('Slope')
plt.ylabel('bandgap [V]')
plt.title('Bandgap measurement of TP'+str(TP)+' at '+str(tmp)+'C')
plt.grid(visible=True, which='major', color='grey', linestyle='--')
#plt.legend([''])
ps_agilent.write('SYST:BEEP 1000,0.7')
plt.show()
```

Code 4: Measurement program for final characterization

D Data analysis

```
%% Author: Yannick Konstandin
% University of Bergamo
% Professor Traversi
% arcadia testboard
% bandgap characterization (-40C to 70C)
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clear;
clc;
close all;

% read csv files
```

```
14
15 for i=1:16
16   dm40=dir("Results_TP"+i+"_REG_m40.csv");
17   xm40=readmatrix(dm40(1).name);
18   eval(['TP' num2str(i) '_m40=xm40']);
19
20   dm30=dir("Results_TP"+i+"_REG_m30.csv");
21   xm30=readmatrix(dm30(1).name);
22   eval(['TP' num2str(i) '_m30=xm30']);
23
24   dm20=dir("Results_TP"+i+"_REG_m20.csv");
25   xm20=readmatrix(dm20(1).name);
26   eval(['TP' num2str(i) '_m20=xm20']);
27
28   dm10=dir("Results_TP"+i+"_REG_m10.csv");
29   xm10=readmatrix(dm10(1).name);
30   eval(['TP' num2str(i) '_m10=xm10']);
31
32   d0=dir("Results_TP"+i+"_REG_0.csv");
33   x0=readmatrix(d0(1).name);
34   eval(['TP' num2str(i) '_0=x0']);
35
36   d10=dir("Results_TP"+i+"_REG_10.csv");
37   x10=readmatrix(d10(1).name);
38   eval(['TP' num2str(i) '_10=x10']);
39
40   d20=dir("Results_TP"+i+"_REG_20.csv");
41   x20=readmatrix(d20(1).name);
42   eval(['TP' num2str(i) '_20=x20']);
43
44   d30=dir("Results_TP"+i+"_REG_30.csv");
45   x30=readmatrix(d30(1).name);
46   eval(['TP' num2str(i) '_30=x30']);
47
48   d40=dir("Results_TP"+i+"_REG_40.csv");
49   x40=readmatrix(d40(1).name);
50   eval(['TP' num2str(i) '_40=x40']);
51
52   d50=dir("Results_TP"+i+"_REG_50.csv");
53   x50=readmatrix(d50(1).name);
54   eval(['TP' num2str(i) '_50=x50']);
55
56   d60=dir("Results_TP"+i+"_REG_60.csv");
57   x60=readmatrix(d60(1).name);
58   eval(['TP' num2str(i) '_60=x60']);
59
60   d70=dir("Results_TP"+i+"_REG_70.csv");
61   x70=readmatrix(d70(1).name);
62   eval(['TP' num2str(i) '_70=x70']);
63 end
```

```
64
65 MP_m40 = [TP1_m40 TP2_m40 TP3_m40 TP4_m40 TP5_m40 TP6_m40 TP7_m40 TP8_m40 ...
66              TP9_m40 TP10_m40 TP11_m40 TP12_m40 TP13_m40 TP14_m40 TP15_m40
                    TP16_m40];
67
68 MP_m30 = [TP1_m30 TP2_m30 TP3_m30 TP4_m30 TP5_m30 TP6_m30 TP7_m30 TP8_m30 ...
69               TP9_m30 TP10_m30 TP11_m30 TP12_m30 TP13_m30 TP14_m30 TP15_m30
                     TP16_m30];
70
71 MP_m20 = [TP1_m20 TP2_m20 TP3_m20 TP4_m20 TP5_m20 TP6_m20 TP7_m20 TP8_m20 ...
72               TP9_m20 TP10_m20 TP11_m20 TP12_m20 TP13_m20 TP14_m20 TP15_m20
                     TP16_m20];
73
74 MP_m10 = [TP1_m10 TP2_m10 TP3_m10 TP4_m10 TP5_m10 TP6_m10 TP7_m10 TP8_m10...
75              TP9_m10 TP10_m10 TP11_m10 TP12_m10 TP13_m10 TP14_m10 TP15_m10
                    TP16_m10];
76
77 MP_0 = [TP1_0 TP2_0 TP3_0 TP4_0 TP5_0 TP6_0 TP7_0 TP8_0 ...
78              TP9_0 TP10_0 TP11_0 TP12_0 TP13_0 TP14_0 TP15_0 TP16_0];
79
80 MP_10 = [TP1_10 TP2_10 TP3_10 TP4_10 TP5_10 TP6_10 TP7_10 TP8_10 ...
81              TP9_10 TP10_10 TP11_10 TP12_10 TP13_10 TP14_10 TP15_10 TP16_10];
82
83 MP_20 = [TP1_20 TP2_20 TP3_20 TP4_20 TP5_20 TP6_20 TP7_20 TP8_20 ...
84              TP9_20 TP10_20 TP11_20 TP12_20 TP13_20 TP14_20 TP15_20 TP16_20];
85
86 MP_30 = [TP1_30 TP2_30 TP3_30 TP4_30 TP5_30 TP6_30 TP7_30 TP8_30 ...
87              TP9_30 TP10_30 TP11_30 TP12_30 TP13_30 TP14_30 TP15_30 TP16_30];
88
89 MP_40 = [TP1_40 TP2_40 TP3_40 TP4_40 TP5_40 TP6_40 TP7_40 TP8_40 ...
90              TP9_40 TP10_40 TP11_40 TP12_40 TP13_40 TP14_40 TP15_40 TP16_40];
91
92 MP_50 = [TP1_50 TP2_50 TP3_50 TP4_50 TP5_50 TP6_50 TP7_50 TP8_50 ...
93              TP9_50 TP10_50 TP11_50 TP12_50 TP13_50 TP14_50 TP15_50 TP16_50];
94
95 MP_60 = [TP1_60 TP2_60 TP3_60 TP4_60 TP5_60 TP6_60 TP7_60 TP8_60 ...
96              TP9_60 TP10_60 TP11_60 TP12_60 TP13_60 TP14_60 TP15_60 TP16_60];
97
98 MP_70 = [TP1_70 TP2_70 TP3_70 TP4_70 TP5_70 TP6_70 TP7_70 TP8_70 ...
99              TP9_70 TP10_70 TP11_70 TP12_70 TP13_70 TP14_70 TP15_70 TP16_70];
100
101 MP_ALL = [MP_m40 MP_m30 MP_m20 MP_m10 MP_0 MP_10 MP_20 MP_30 MP_40 MP_50 MP_60
      MP_70];
102 %% plot many TPs Vout / slope
103 % showing different colors for mean value
104 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
105 clc;
106 close all;
107
108 vin = 1.2; % change plot for v_in (1.08; 1.2; 1.32)
```

```
109
110 table = MP_10;
111 color = jet(17);
112
113 figure('Name','bandgap characterization','NumberTitle','off');grid on, grid minor
114
115 %for i=9:9:72
116 for i=81:9:144
117     %j = TP * 9;
118     j = i/9;
119     if j <= 8
120         subplot(2,4,j);grid on, grid minor
121         hold on
122     else
123         subplot(2,4,j-8);grid on, grid minor
124         hold on
125     end
126     for p = 1:769
127         if table(p,6) == vin
128         c = table(p,8)+1;
129         plot(table(p,7),table(p,i),"LineWidth",1.5,'Marker','.','MarkerSize',18,
130         'LineStyle','none','Color',color(c,:))
131         volt = table(p,6);
132         ylabel('bandgap [V]')
133         xlabel('slope')
134         title("TP"+j+" VDDA = "+volt+"V at "+table(p,4)+"C")
135         labels=num2str((0:15).','mean: %d');
136 ________%legend(labels)
137 ________%legend('Location','southwest')
138 ________end
139 ____end
140 end
141 %% plot one TP Vout / slope
142 % showing different colors for mean value
143 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
144 clc;
145 close all;
146
147 vin = 1.2; % change plot for v_in (1.08; 1.2; 1.32)
148 TP = 16;
149
150 table = MP_10;
151 color = jet(17);
152
153 figure('Name','bandgap characterization','NumberTitle','off');grid on, grid minor
154 hold on
155 for p = 1:769
156 ____if table(p,6) == vin
157 ____c = table(p,8)+1;
158 ____plot(table(p,7),table(p,TP*9),"LineWidth",1.5,'Marker','.','MarkerSize',18,
```

```
    'LineStyle','none','Color',color(c,:))
    volt = table(p,6);
    ylabel('bandgap [V]')
    xlabel('slope')
    title("TP"+TP+" VDDA = "+volt+"V at "+table(p,4)+"C")
    labels=num2str((0:15).','mean: %d');
        legend(labels)
        legend('Location','southwest')
        end
end

%% plot single TP Vout / mean
% showing different colors for mean value
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clc
close all

TP = 1;
v_in = 1.2;
table = MP_m10;

color = jet(17);
figure('Name','bandgap characterization','NumberTitle','off');grid on, grid minor
hold on

for p = 1:769
    if table(p,6) == v_in
    c = table(p,7)+1;
    plot(table(p,8), table(p,TP*9),"LineWidth",1.5,'Marker','.','MarkerSize',18,
    'LineStyle','none','Color',color(c,:));grid on, grid minor
    volt = table(p,6);
    ylabel('bandgap [V]')
    xlabel('mean')
    title("TP"+TP+" VDDA = "+volt+"V at "+table(p,4)+"C");
    legend('slope: 0','','','','','','','','','','','','','','','', ...
        'slope: 1','','','','','','','','','','','','','','','','', ...
        'slope: 2','','','','','','','','','','','','','','','','', ...
        'slope: 3','','','','','','','','','','','','','','','','', ...
        'slope: 4','','','','','','','','','','','','','','','','', ...
        'slope: 5','','','','','','','','','','','','','','','','', ...
        'slope: 6','','','','','','','','','','','','','','','','', ...
        'slope: 7','','','','','','','','','','','','','','','','', ...
        'slope: 8','','','','','','','','','','','','','','','','', ...
        'slope: 9','','','','','','','','','','','','','','','','', ...
        'slope: 10','','','','','','','','','','','','','','','','', ...
        'slope: 11','','','','','','','','','','','','','','','','', ...
        'slope: 12','','','','','','','','','','','','','','','','', ...
        'slope: 13','','','','','','','','','','','','','','','','', ...
        'slope: 14','','','','','','','','','','','','','','','','', ...
        'slope: 15','','','','','','','','','','','','','','','','')
```

```
209     legend('Location','southeast')
210     end
211 end
212
213
214 %% plot all 16 TPs Vout / mean
215 % showing different colors for mean value
216 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
217 clc
218 close all
219
220 color = jet(17);
221 figure('Name','bandgap characterization','NumberTitle','off');
222
223 v_in = 1.2;
224 table = MP_10;
225
226 %for i=9:9:72
227 for i=81:9:144
228     j = i/9;
229     if j <= 8
230         subplot(2,4,j);grid on, grid minor
231     hold on
232     else
233         subplot(2,4,j-8);grid on, grid minor
234         hold on
235     end
236     for p = 1:769
237         if table(p,6) == v_in
238         c = table(p,7)+1;
239         plot(table(p,8),
                table(p,i),"LineWidth",1.5,'Marker','.','MarkerSize',12,'LineStyle',
240         'none', 'Color',color(c,:));grid on, grid minor
241         volt = table(p,6);
242         ylabel('bandgap [V]')
243         xlabel('mean')
244         %ylim([0.42 0.67])
245         title("TP"+j+" VDDA = "+volt+"V at "+table(p,4)+"C");
246         %legend('slope: 0','','','','','','','','','','','','','','','', ...
247 %             'slope: 1','','','','','','','','','','','','','','','', ...
248 %             'slope: 2','','','','','','','','','','','','','','','', ...
249 %             'slope: 3','','','','','','','','','','','','','','','', ...
250 %             'slope: 4','','','','','','','','','','','','','','','', ...
251 %             'slope: 5','','','','','','','','','','','','','','','', ...
252 %             'slope: 6','','','','','','','','','','','','','','','', ...
253 %             'slope: 7','','','','','','','','','','','','','','','', ...
254 %             'slope: 8','','','','','','','','','','','','','','','', ...
255 %             'slope: 9','','','','','','','','','','','','','','','', ...
256 %             'slope: 10','','','','','','','','','','','','','','','', ...
257 %             'slope: 11','','','','','','','','','','','','','','','', ...
```

```
% 'slope: 12','','','','','','','','','','','','','','','', ...
% 'slope: 13','','','','','','','','','','','','','','','', ...
% 'slope: 14','','','','','','','','','','','','','','','', ...
% 'slope: 15','','','','','','','','','','','','','','','')
        %legend('Location','southeast')
        end
    end
end

%% plot single TP Vout / temp for all slopes
% mean = 7
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clc;
close all;

MP_ALL = [MP_m40 MP_m30 MP_m20 MP_m10 MP_0 MP_10 MP_20 MP_30 MP_40 MP_50 MP_60
    MP_70];

vin = 1.2; % change plot for v_in (1.08; 1.2; 1.32)
TP = 1;
color = jet(17);

figure('Name','bandgap characterization','NumberTitle','off');grid on, grid minor
hold on
long = size(MP_ALL);

for i=9*TP:144:long(1,2)
    if TP > 0 && TP <= 16
        MAX = 0;
    for p=1:769
        c = MP_ALL(p,i-2)+1;
        if MP_ALL(p,6) == vin && MP_ALL(p,i-1) == 7
            plot(MP_ALL(p,i-5),MP_ALL(p,i),"LineWidth",1.5,'Marker','.',
            'MarkerSize',18, 'LineStyle','-','Color',color(c,:));
            volt = MP_ALL(p,6);
            ylabel('bandgap [V]')
            xlabel('temperature [C]')
            xlim([-40 70])
            title("TP"+TP+" VDDA = "+volt+"V; mean= "+MP_ALL(p,8)+"")
            labels=num2str((0:15).','slope: %d');
            legend(labels);
            legend('Location','southwest')
        end
    end
    else
        fprintf('TP out of range\n')
    end
end

%% plot all TP Vout / temp for slope=7 mean=7
```

```
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clc;
%close all;

MP_ALL = [MP_m40 MP_m30 MP_m20 MP_m10 MP_0 MP_10 MP_20 MP_30 MP_40 MP_50 MP_60 MP_70];

vin = 1.2; % change plot for v_in (1.08; 1.2; 1.32)
TP = 1;
color = jet(17);

figure('Name','bandgap characterization','NumberTitle','off');grid on, grid minor
hold on
long = size(MP_ALL);
for TP=1:16
    for i=9*TP:144:long(1,2)
        if TP > 0 && TP <= 16
            MAX = 0;
        for p=1:769
            c = TP;
            if MP_ALL(p,6) == vin && MP_ALL(p,i-1) == 7 && MP_ALL(p,i-2) == 7
                plot(MP_ALL(p,i-5),MP_ALL(p,i),"LineWidth",1.5,'Marker','.','MarkerSize',18,'LineStyle','-','Color',color(c,:));
                volt = MP_ALL(p,6);
                ylabel('bandgap [V]')
                xlabel('temperature [C]')
                xlim([-40 70])
                title("All TPs VDDA = "+volt+"V; mean= "+MP_ALL(p,8)+"; slope= "+MP_ALL(p,7)+"")
                labels=num2str((0:15).','slope: %d');
                legend(labels);
                legend('Location','southwest')
            end
        end
        else
            fprintf('TP out of range\n')
        end
    end
end
%% plot only one slope
clc
close all

figure('Name','bandgap characterization','NumberTitle','off');grid on, grid minor
hold on
z = 1;
for TP= 1:16
    for i=9*TP:144:long(1,2)
```

```
        if TP > 0 && TP <= 16
            for p=1:long(1,1)
                if MP_ALL(p,6) == 1.2 && MP_ALL(p,i-1) == 7 && MP_ALL(p,i-2) == 7
                    TP_slope7(z,1:3) = MP_ALL(p,[i-5 i i-2])
                    z = z + 1;
                end
            end
        end
    end
    start = TP+(TP-1)*11;
    ende = start + 11;
    plot(TP_slope7(start:ende,1),TP_slope7(start:ende,2),"LineWidth",1.5,
    'Marker','.','MarkerSize',18,'LineStyle','-','Color',color(TP,:));
    ylabel('bandgap [V]')
    xlabel('temperature [C]')
    xlim([-40 70])
    title("All TPs for VDDA = 1.2V; mean= 7; slope= 7");
    labels=num2str((1:16).','TP%d');
    legend(labels);
    legend('Location','southwest')
    writematrix(TP_slope7,'C:\Users\Yannick\OneDrive - KONSTANDIN
    GMBH\Dokumente\Technikum\4. Semester\Project
    Traversi\Matlab\Data\TP1_TEMP_SLOPE7.csv');
end


%% plot single TP Vout / temp for all mean
% slope = 7
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clc;
close all;

MP_ALL = [MP_m40 MP_m30 MP_m20 MP_m10 MP_0 MP_10 MP_20 MP_30 MP_40 MP_50 MP_60
    MP_70];

vin = 1.2; % change plot for v_in (1.08; 1.2; 1.32)
TP = 8;
slope_=10;
color = jet(17);

figure('Name','bandgap characterization','NumberTitle','off');grid on, grid minor
hold on
long = size(MP_ALL);

for i=9*TP:144:long(1,2)
    if TP > 0 && TP <= 16
        for p=1:769
            c = MP_ALL(p,i-1)+1;
            if MP_ALL(p,6) == vin && MP_ALL(p,i-2) == slope_
                plot(MP_ALL(p,i-5),MP_ALL(p,i),"LineWidth",1.5,'Marker','.',
```

```
␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣'MarkerSize',18,'LineStyle','-','Color',color(c,:));
␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣volt␣=␣MP_ALL(p,6);
␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣ylabel('bandgap [V]')
␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣xlabel('temperature [C]')
␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣xlim([-40␣70])
␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣title("TP"+TP+"␣VDDA␣=␣"+volt+"V;␣slope=␣"+MP_ALL(p,i-2)+"")
␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣labels=num2str((0:15).','mean:␣%d');
                legend(labels);
                legend('Location','southwest')
            end
        end
    end
end
%% print data to csv
clc;
close all;

MP_ALL = [MP_m40 MP_m30 MP_m20 MP_m10 MP_0 MP_10 MP_20 MP_30 MP_40 MP_50 MP_60
    MP_70];

 TP=1;
    z = 1;
    clear("A");
    for i= 9*TP:144:long(1,2)
        for p = 1:769
            if MP_ALL(p,i-3) == 1.2 && MP_ALL(p,i-1) == 7
                A(z,1:4) = MP_ALL(p,[i i-5 i-2 i-1]);
                z = z+1;
            end
        end
    end
    writematrix(A,'C:\Users\Yannick\OneDrive␣-␣KONSTANDIN␣
        GMBH\Dokumente\Technikum\4.␣Semester\Project␣
        Traversi\Matlab\Data\TP1_TEMP.csv');



%% calculate and plot TC
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clc
%close all

MP_ALL = [MP_m40 MP_m30 MP_m20 MP_m10 MP_0 MP_10 MP_20 MP_30 MP_40 MP_50 MP_60
    MP_70];
long = size(MP_ALL);

range = 258:513;

```

```
A = zeros(192,4);

% extract all data for VDDA 1,2V and mean = 7
for TP=1:16
    z = 1;
    clear("A");
    for i= 9*TP:144:long(1,2)
        for p = 1:769
            if MP_ALL(p,i-3) == 1.2 && MP_ALL(p,i-1) == 7
                A(z,1:4) = MP_ALL(p,[i i-5 i-2 i-1]);
                eval(['TP' num2str(TP) '_MEAN_7=A']);
                z = z+1;
            end
        end
    end
end

% find max/min for each slope
long = size(TP1_MEAN_7);

for TP = 1:16
Table = eval(['TP' int2str(TP) '_MEAN_7']);
clear("TP_buff")
    for slope = 0:15
        z = 1;
        for p=1:long(1,1)
            if Table(p,3) == slope
                TP_buff(z,1:2) = Table(p,[1 3])
                z = z + 1;
            end

        end
        TP_MIN(slope+1,1:2) = min(TP_buff)
        TP_MAX(slope+1,1:2) = max(TP_buff);
        TP_ALL = [TP_MAX TP_MIN];
        clear("TP_buff")
    end
    eval(['TP' int2str(TP) '_EXTR=␣TP_ALL']);
end

% get vout 30C
for TP = 1:16
Table = eval(['TP' int2str(TP) '_MEAN_7']);
    for slope = 0:15
        z = 1;
        for p=1:long(1,1)
            if Table(p,3) == slope && Table(p,2) == 30
                TP_buff(z,1:2) = Table(p,[1 3]);
                z = z + 1;
            end
```

```
        end
        TP_30(slope+1,1:2) = TP_buff;
        clear("TP_buff");
    end
    eval(['TP' int2str(TP) '_vout_30=␣TP_30']);
end

% calculate TC
for TP=1:16
clear("TC_buff")
table_extr = eval(['TP' num2str(TP) '_EXTR']);
TP_vout_30 = eval(['TP' num2str(TP) '_vout_30']);
    for slope = 1:16
        V_MAX = table_extr(slope,1);
        V_MIN = table_extr(slope,3);
        Vout_30 = TP_vout_30(slope,1);
        delta = V_MAX-V_MIN;

        TC = ((delta/Vout_30)/110)*10^6;

        TC_buff(slope,1) = slope-1;
        TC_buff(slope,2) = TC;
    end
    eval(['TP' num2str(TP) '_TC=TC_buff'])
end


% plot TCs
color = jet(17);

figure('Name','bandgap␣characterization','NumberTitle','off');grid on, grid minor
hold on
for TP=1:16
    table_tc = eval(['TP' num2str(TP) '_TC']);
    TC = table_tc(:,2);
    slope = table_tc(:,1);
    c = TP;
    plot(slope,TC,"LineWidth",1.5,'Marker','.','MarkerSize',18,'LineStyle','-',
    'Color',color(c,:));grid on, grid minor;
end

ylabel('TC␣[ppm/C]')
xlabel('slope')
title("Temperature␣coefficients")
labels=num2str((1:16).','TP: %d');
legend(labels);
legend('Location','southwest')

%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
%%%%%%%%%%%%%%      find best TC        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
clc
%close all

MP_ALL = [MP_m40 MP_m30 MP_m20 MP_m10 MP_0 MP_10 MP_20 MP_30 MP_40 MP_50 MP_60 MP_70];
long = size(MP_ALL);

for TP=1:16
    z = 1;
    clear("B");
    for i= 9*TP:144:long(1,2)
        for p = 1:769
            if MP_ALL(p,i-3) == 1.2
                B(z,1:4) = MP_ALL(p,[i i-5 i-2 i-1]);
                z = z+1;
            end
        end
    end
    eval(['TP' num2str(TP) '_MEAN_SLOPE=B']);
end

% write data to csv
% for TP = 1:16
%     BUFF = eval(['TP' num2str(TP) '_MEAN_SLOPE']);
%     NEW = BUFF(:,[1 3 4]);
%     eval(['TP' num2str(TP) '_VOUT=NEW']);
% end
% F = [TP1_VOUT TP2_VOUT TP3_VOUT TP4_VOUT TP5_VOUT TP6_VOUT TP7_VOUT TP8_VOUT TP9_VOUT ...
%     TP10_VOUT TP11_VOUT TP12_VOUT TP13_VOUT TP14_VOUT TP15_VOUT TP16_VOUT];
VOUT = [TP1_MEAN_SLOPE TP2_MEAN_SLOPE TP3_MEAN_SLOPE TP4_MEAN_SLOPE TP5_MEAN_SLOPE TP6_MEAN_SLOPE TP7_MEAN_SLOPE TP8_MEAN_SLOPE...
        TP9_MEAN_SLOPE TP10_MEAN_SLOPE TP11_MEAN_SLOPE TP12_MEAN_SLOPE TP13_MEAN_SLOPE TP14_MEAN_SLOPE TP15_MEAN_SLOPE TP16_MEAN_SLOPE];
writematrix(VOUT, "C:\Users\Yannick\OneDrive - KONSTANDIN GMBH\Dokumente\Technikum\4. Semester\Project Traversi\Matlab\Data\VOUT_TPs.csv");



long = size(TP1_MEAN_SLOPE);

% find min/max

for TP = 1:16
    Table = eval(['TP' int2str(TP) '_MEAN_SLOPE']);
    %clear("TP_buff")
```

```
        x = 1;
        i = 1;
    for slope = 0:15
        z = 1;
        for p=1:long(1,1)
            if Table(p,3) == slope
                TP_buffer(z,1:4) = Table(p,[1 2 3 4]);
                z = z + 1;
            end
        end

        %clear("EXTR_ALL")
        for mean = 0:15
            for t = 1:192
                if TP_buffer(t,4) == mean
                TP_EXT(x,1:4) = TP_buffer(t,[1 2 3 4]);
                x = x+1;
                end
            end
            x = 1;
            MIN_ = min(TP_EXT);
            MAX_ = max(TP_EXT);
            EXTR_ALL(i, 1:8) = [MAX_ MIN_];
            %clear("TP_EXT")
            i = i + 1;
        end
    end
    eval(['TP' int2str(TP) '_EXTR_ALL= EXTR_ALL']);
end

% get vout 30C
for TP = 1:16
Table = eval(['TP' int2str(TP) '_MEAN_SLOPE']);
z = 1;
    for slope = 0:15
        for p=1:long(1,1)
            if Table(p,3) == slope && Table(p,2) == 30
                TP_buff_30(z,1:4) = Table(p,[1 2 3 4]);
                z = z + 1;
            end
        end
        TP_30 = TP_buff_30;
    end
    eval(['TP' int2str(TP) '_vout_all_30= TP_30']);
end

% calculate TC
long = size(TP1_EXTR_ALL);

for TP=1:16
```

```
641 table_tc_all = eval(['TP' num2str(TP) '_EXTR_ALL']);
642 TP_vout_all_30 = eval(['TP' num2str(TP) '_vout_all_30']);
643
644     for column = 1:long(1,1)
645         V_MAX_all = table_tc_all(column,1);
646         V_MIN_all = table_tc_all(column,5);
647         Vout_30_all = TP_vout_all_30(column,1);
648         delta_V = V_MAX_all-V_MIN_all;
649
650         TC_all = ((delta_V/Vout_30_all)/110)*10^6;
651
652         TC_buffer(column,1) = TC_all;
653         TC_buffer(column,2) = table_tc_all(column,3);
654         TC_buffer(column,3) = table_tc_all(column,4);
655
656     end
657     eval(['TP' num2str(TP) '_TC_ALL=TC_buffer']);
658 end
659
660 % write TC in csv
661 TC_ALL = [TP1_TC_ALL TP2_TC_ALL TP3_TC_ALL TP4_TC_ALL TP5_TC_ALL TP6_TC_ALL
        TP7_TC_ALL TP8_TC_ALL...
662     TP9_TC_ALL TP10_TC_ALL TP11_TC_ALL TP12_TC_ALL TP13_TC_ALL TP14_TC_ALL
        TP15_TC_ALL TP16_TC_ALL];
663
664 writematrix(TC_ALL, "C:\Users\Yannick\OneDrive - KONSTANDIN
        GMBH\Dokumente\Technikum\4. Semester\Project
        Traversi\Matlab\Data\TC_all_TPs.csv");
665
666 R = [TP1_TC_ALL; TP2_TC_ALL; TP3_TC_ALL; TP4_TC_ALL; TP5_TC_ALL; TP6_TC_ALL;
        TP7_TC_ALL; TP8_TC_ALL; ...
667     TP9_TC_ALL; TP10_TC_ALL; TP11_TC_ALL; TP12_TC_ALL; TP13_TC_ALL; TP14_TC_ALL;
        TP15_TC_ALL; TP16_TC_ALL];
668
669 MIN_TC = min(R);
670
671 % plot TCs
672 % plot TC over slope and change color for mean
673 color = jet(17);
674
675 figure('Name','bandgap characterization','NumberTitle','off');grid on, grid minor
676 hold on
677 for TP=1:16
678     table_tc = eval(['TP' num2str(TP) '_TC_ALL']);
679     TC_ = table_tc(:,1);
680     slope_ = table_tc(:,2);
681     %mean = table_tc()
682     c = TP;
683     plot(slope_,TC_,"LineWidth",1.5,'Marker','.','MarkerSize',18,'LineStyle',
684     'none', 'Color',color(c,:));grid on, grid minor;
```

```
685 end
686
687 ylabel('TC [ppm/C]')
688 xlabel('slope')
689 title("Temperature coefficients")
690 labels=num2str((1:16).','TP: %d');
691 legend(labels);
692 legend('Location','southwest')
693
694
695
696
697
698 %% Plot Vin / Vout for every TP
699 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
700 clc
701 close all
702
703 for i=1:16
704   up40=dir("Results_TP"+i+"_TEMP_m40_0_UP.csv");
705   xup40=readmatrix(up40(1).name);
706   eval(['TP' num2str(i) '_m40_vin=xup40']);
707
708   down40=dir("Results_TP"+i+"_TEMP_m40_132_DOWN.csv");
709   xdown40=readmatrix(down40(1).name);
710   eval(['TP' num2str(i) '_m40_vin_down=xdown40']);
711
712   up30=dir("Results_TP"+i+"_TEMP_30_0_UP.csv");
713   xup30=readmatrix(up30(1).name);
714   eval(['TP' num2str(i) '_30_vin=xup30']);
715
716   down30=dir("Results_TP"+i+"_TEMP_30_132_DOWN.csv");
717   xdown30=readmatrix(down30(1).name);
718   eval(['TP' num2str(i) '_30_vin_down=xdown30']);
719
720   up70=dir("Results_TP"+i+"_TEMP_70_0_UP.csv");
721   xup70=readmatrix(up70(1).name);
722   eval(['TP' num2str(i) '_70_vin=xup70']);
723
724   down70=dir("Results_TP"+i+"_TEMP_70_132_DOWN.csv");
725   xdown70=readmatrix(down70(1).name);
726   eval(['TP' num2str(i) '_70_vin_down=xdown70']);
727 end
728 VIN_m40_up = [TP1_m40_vin TP2_m40_vin TP3_m40_vin TP4_m40_vin TP5_m40_vin
      TP6_m40_vin TP7_m40_vin TP8_m40_vin ...
729             TP9_m40_vin TP10_m40_vin TP11_m40_vin TP12_m40_vin TP13_m40_vin
                  TP14_m40_vin TP15_m40_vin TP16_m40_vin];
730
731 VIN_m40_down = [TP1_m40_vin_down TP2_m40_vin_down TP3_m40_vin_down
      TP4_m40_vin_down TP5_m40_vin_down TP6_m40_vin_down TP7_m40_vin_down
```

```
      TP8_m40_vin_down ...
732                 TP9_m40_vin_down TP10_m40_vin_down TP11_m40_vin_down
                        TP12_m40_vin_down TP13_m40_vin_down TP14_m40_vin_down
                        TP15_m40_vin_down TP16_m40_vin_down];
733
734 VIN_30_down = [TP1_30_vin_down TP2_30_vin_down TP2_30_vin_down TP4_30_vin_down
      TP5_30_vin_down TP6_30_vin_down TP7_30_vin_down TP8_30_vin_down ...
735                 TP9_30_vin_down TP10_30_vin_down TP11_30_vin_down
                        TP12_30_vin_down TP13_30_vin_down TP14_30_vin_down
                        TP15_30_vin_down TP16_30_vin_down];
736
737 VIN_30_up = [TP1_30_vin TP2_30_vin TP3_30_vin TP4_30_vin TP5_30_vin TP6_30_vin
      TP7_30_vin TP8_30_vin TP9_30_vin TP10_30_vin ...
738                 TP11_30_vin TP12_30_vin TP13_30_vin TP14_30_vin TP15_30_vin
                        TP16_30_vin];
739
740 VIN_70_down = [TP1_70_vin_down TP2_70_vin_down TP3_70_vin_down TP4_70_vin_down
      TP5_70_vin_down TP6_70_vin_down TP7_70_vin_down TP8_70_vin_down...
741                 TP9_70_vin_down TP10_70_vin_down TP11_70_vin_down
                        TP12_70_vin_down TP13_70_vin_down TP14_70_vin_down
                        TP15_70_vin_down TP16_70_vin_down];
742
743 VIN_70_up = [TP1_70_vin TP2_70_vin TP3_70_vin TP4_70_vin TP5_70_vin TP6_70_vin
      TP7_70_vin TP8_70_vin ...
744                 TP9_70_vin TP10_70_vin TP11_70_vin TP12_70_vin TP13_70_vin
                        TP14_70_vin TP15_70_vin TP16_70_vin];
745
746
747 color = jet(17);
748 figure('Name','bandgap␣characterization','NumberTitle','off');grid on, grid minor
749 hold on
750
751 table = VIN_70_up;
752 j = 1;
753 for i=4:6:96
754     if j <= 8
755         plot(table(:,2),table(:,i),"LineWidth",1.5,'Marker','.','MarkerSize',18,
756         'Color',color(j,:));
757         j = j + 1;
758     else
759         plot(table(:,2),table(:,i),"LineWidth",1.5,'Marker','.','MarkerSize',18,
760         'Color', color(j,:));
761         j = j + 1;
762     end
763 end
764 temp = table(2,5);
765 ylabel('bandgap␣[V]')
766 xlabel('Vin␣(VDDA)␣[V]')
767 xlim([0 1.32])
768 labels=num2str((1:16).','TP%d');
```

```
legend(labels)
legend('Location','northwest')
title("Bandgap measurements VDDA from 0V to 1,32V at "+temp+"C")
hold off

color = jet(17);
figure('Name','bandgap characterization','NumberTitle','off');grid on, grid minor
hold on
table = VIN_70_down;
range = 2:134;
j = 1;
for i=4:6:96
    if j <= 8
        plot(table(:,2),table(:,i),"LineWidth",1.5,'Marker','.','MarkerSize',18,
        'Color',color(j,:));
        j = j + 1;
    else
        plot(table(:,2),table(:,i),"LineWidth",1.5,'Marker','.','MarkerSize',18,
        'Color', color(j,:));
        j = j + 1;
        temp = table(2,5);
    end
end

ylabel('bandgap [V]')
xlabel('Vin (VDDA) [V]')
xlim([0 1.32])
labels=num2str((1:16).','TP%d');
legend(labels)
legend('Location','northwest')
title("Bandgap measurements VDDA from 1,32V to 0V at "+temp+"C")
hold off


%% calculate line regulation for mean = slope
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clc
close all

temp    = 70;
slope   = 7;
mean    = 1;


MP_ALL = [MP_m40 MP_m30 MP_m20 MP_m10 MP_0 MP_10 MP_20 MP_30 MP_40 MP_50 MP_60
      MP_70];
long = size(MP_ALL);
```

```
818
819
820 % create new variable
821 for TP=1:16
822     z = 1;
823     clear("A");
824     for i= 9*TP:144:long(1,2)
825         for p = 1:769
826             if MP_ALL(p,i-2) == slope && MP_ALL(p,i-1) == mean && MP_ALL(p,i-5)
                    == temp
827                 A(z,1:3) = MP_ALL(p,[i i-3 i-5]);
828                 eval(['TP' num2str(TP) '_LIN_REG=A']);
829                 z = z+1;
830             end
831         end
832     end
833 end
834
835 color = jet(17);
836 header = {'TP','slope','intersection','LR␣[%/V]', 'Vout␣(1.08)', 'Vout␣
      (1.2)','Vout␣(1.32)'};
837
838 figure('Name','bandgap␣characterization','NumberTitle','off');grid on,
839 hold on
840 for TP=1:16
841     table_LIN = eval(['TP' num2str(TP) '_LIN_REG']);
842     Vin = table_LIN(:,2);
843     Vout = table_LIN(:,1);
844     % linear regression
845     P = polyfit(Vin,Vout,1);
846     yfit = polyval(P,Vin);
847
848     % calculate line regulation
849     dVref = max(Vout)-min(Vout);
850     Vref = Vout(2);
851     dVDD = 1.32-1.08;
852     LR = (dVref/(Vref*dVDD))*100;
853
854     LIN_reg(TP,1)   = TP;
855     LIN_reg(TP,2:3) = P;
856     LIN_reg(TP,4)   = LR;
857     LIN_reg(TP,5)   = Vout(1);
858     LIN_reg(TP,6)   = Vout(2);
859     LIN_reg(TP,7)   = Vout(3);
860     c = TP;
861     r = P(1)+P(2);
862     tmp = table_LIN(1,3);
863
864     hold on;
865     plot(Vin,yfit*1000,'r-.','Color',color(c,:))
```

```
866     plot(Vin,yfit*1000,'r-.',"LineWidth",1.5,'Marker','.','MarkerSize',18,
867     'LineStyle', 'none','Color',color(c,:));
868
869 end
870 LIN_REG = [header;num2cell(LIN_reg)];
871 ylabel('Vout [mV]')
872 xlabel('VDDA [V]')
873 xlim([1.08 1.32])
874 box on;
875 title("Line regulation at "+tmp+"C")
876 legend('','TP1','','TP2','','TP3','','TP4','','TP5','','TP6','','TP7','','TP8',
877 '','TP9','','TP10','','TP11','','TP12','','TP13','','TP14','','TP15','','TP16');
878 legend('Location','northeastoutside')
879
880 % export Data
881 if tmp > 0
882 exportgraphics(gcf,['Data/LIN_REG_' num2str(tmp)
      '_7_1.pdf'],'ContentType','vector');
883 writecell(LIN_REG,['Data/LIN_REG_' num2str(tmp) '_7_1.csv']);
884 else
885 t = tmp*-1;
886 exportgraphics(gcf,['Data/LIN_REG_m' num2str(t)
      '_7_1.pdf'],'ContentType','vector');
887 writecell(LIN_REG,['Data/LIN_REG_m' num2str(t) '_7_1.csv']);
888 end
889
890 %% calculate line regulation for best config
891 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
892 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
893 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
894 clc
895 close all
896
897 temp    = 30;
898
899 MP_ALL = [MP_m40 MP_m30 MP_m20 MP_m10 MP_0 MP_10 MP_20 MP_30 MP_40 MP_50 MP_60
      MP_70];
900 long = size(MP_ALL);
901 config = readmatrix('best_TC_slope_mean.dat');
902
903
904 % create new variable
905 for TP=1:16
906     z = 1;
907     clear("D");
908     for i= 9*TP:144:long(1,2)
909         for p = 1:769
910             if MP_ALL(p,i-2) == config(TP,2) && MP_ALL(p,i-1) == config(TP,3) &&
                    MP_ALL(p,i-5) == temp
911                 D(z,1:5) = MP_ALL(p,[i i-3 i-5 i-2 i-1]);
```

```
912                 eval(['TP' num2str(TP) '_LIN_REG_BEST_CON=D']);
913                 z = z+1;
914             end
915         end
916     end
917 end
918
919 color = jet(17);
920 header = {'TP','slope (linfit)','intersection','LR [%/V]', 'Vout (1.08)', 'Vout (1.2)','Vout (1.32)','slope','mean'};
921
922 figure('Name','bandgap characterization','NumberTitle','off');grid on,
923 hold on
924 for TP=1:16
925     table_con = eval(['TP' num2str(TP) '_LIN_REG_BEST_CON'])
926     Vin = table_con(:,2);
927     Vout = table_con(:,1);
928     % linear regression
929     P = polyfit(Vin,Vout,1);
930     yfit = polyval(P,Vin);
931
932     % calculate line regulation
933     dVref = max(Vout)-min(Vout);
934     Vref = Vout(2);
935     dVDD = 1.32-1.08;
936     LR = (dVref/(Vref*dVDD))*100;
937
938     LIN_buff(TP,1)   = TP;
939     LIN_buff(TP,2:3) = P;
940     LIN_buff(TP,4)   = LR;
941     LIN_buff(TP,5)   = Vout(1);
942     LIN_buff(TP,6)   = Vout(2);
943     LIN_buff(TP,7)   = Vout(3);
944     LIN_buff(TP,8)   = table_con(1,4);
945     LIN_buff(TP,9)   = table_con(1,5);
946     c = TP;
947     r = P(1)+P(2);
948     tmp = table_con(1,3);
949
950     hold on;
951     plot(Vin,yfit*1000,'r-.','Color',color(c,:))
952     plot(Vin,yfit*1000,'r-.',"LineWidth",1.5,'Marker','.','MarkerSize',18,
953     'LineStyle','none','Color',color(c,:));
954
955 end
956
957 LIN_BEST = [header;num2cell(LIN_buff)];
958 ylabel('Vout [mV]')
959 xlabel('VDDA [V]')
960 xlim([1.08 1.32])
```

```
box on;
title("Line␣regulation␣at␣"+tmp+"C")
legend('','TP1','','TP2','','TP3','','TP4','','TP5','','TP6','','TP7','','TP8',
'','TP9','','TP10','','TP11','','TP12','','TP13','','TP14','','TP15','','TP16');
legend('Location','northeastoutside')

% export Data
if tmp > 0
exportgraphics(gcf,['Data/LIN_REG_BEST_CON_' num2str(tmp)
    '.pdf'],'ContentType','vector');
writecell(LIN_BEST,['Data/LIN_REG_BEST_CON_' num2str(tmp) '.csv']);
else
t = tmp*-1;
exportgraphics(gcf,['Data/LIN_REG_BEST_CON_m' num2str(t)
    '.pdf'],'ContentType','vector');
writecell(LIN_BEST,['Data/LIN_REG_BEST_CON_m' num2str(t) '.csv']);
end


%% Distribution of Vout
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clc
close all

temp= 30;
vin = 1.2;
slope = 7;
mean = 7;
MP_ALL = [MP_m40 MP_m30 MP_m20 MP_m10 MP_0 MP_10 MP_20 MP_30 MP_40 MP_50 MP_60
    MP_70];
long = size(MP_ALL);


% create new variable for slope=mean=7
for TP=1:16
    clear("D");
    for i= 9*TP:144:long(1,2)
        for p = 1:769
            if MP_ALL(p,i-4) == vin && MP_ALL(p, i-2) == slope && MP_ALL(p, i-1)
                == mean && MP_ALL(p, i-5) == temp
                D = MP_ALL(p,i)*1000;
            end
        end
    end
    R(TP,1) = D;
end
eval(['HIST_1=R']);

```

```
config = readmatrix('best_TC_slope_mean.dat');

% create new variable for best configuration
for TP=1:16
    clear("O");
    for i= 9*TP:144:long(1,2)
        for p = 1:769
            if MP_ALL(p,i-2) == config(TP,2) && MP_ALL(p,i-1) == config(TP,3) &&
                MP_ALL(p,i-5) == temp && MP_ALL(p,i-4) == vin
                O = MP_ALL(p,i)*1000;
            end
        end
    end
    N(TP,1) = O;
end
eval(['HIST_BEST=N']);

bins = 10;
A = sort(HIST_1(:,1));
pd = fitdist(A,'Normal');

B = sort(HIST_BEST);
pd_b = fitdist(B,'Normal');


f=figure('Name','Gaussian-distribution␣of␣TP','NumberTitle','off'); grid on
hold on

X = histfit(A,bins,'normal');
set(X(1),'FaceAlpha',.5);
set(X(1),'FaceColor','blue');
set(X(2),'Color','blue');
Z = histfit(B,bins,'normal');
set(Z(1),'FaceAlpha',.5);
set(Z(1),'FaceColor','red');
set(Z(2),'Color','red');

hold off
title("Distribution␣TP1-TP16␣at␣"+temp+"C,␣VDDA:␣"+vin+"V");
xlabel('Vout␣[mV]');
ylabel('counts');

hist1 = sprintf('Vout␣slope:␣%1i␣mean:␣%1i:␣\\mu␣=␣%3.2f,␣\\sigma␣=␣
      %3.2f',slope, mean, pd.mu, pd.sigma);
hist2 = sprintf("␣␣␣␣␣␣␣␣␣Vout␣best␣config:␣\\mu␣=␣%3.2f,␣\\sigma␣=␣
      %3.2f",pd_b.mu, pd_b.sigma);
legend(hist1,'',hist2,'')
legend('Location','northwest')
box on;
ax = gca;
```

```
%ax.XAxis.FontSize = fontsize;
%ax.YAxis.FontSize = fontsize;
ax.Legend.FontSize = 12;
f.Position = [400 280 900  550];
hold off

% export Data
if temp > 0
exportgraphics(gcf,('Data/HIST_77_BEST_COM.pdf'),'ContentType','vector');
else
t = temp*-1;
exportgraphics(gcf,('Data/HIST_77_BEST_COM.pdf'),'ContentType','vector');
end

%% Compare Distribution of TPs with different means
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% find best config when mean is fixed
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clc
close all

temp= 30;
vin = 1.2;
slope = 7;
mean = 7;

long = size(MP_ALL);

% create new variable
for TP=1:16
    clear("D");
    for i= 9*TP:144:long(1,2)
        for p = 1:769
            if MP_ALL(p,i-4) == vin && MP_ALL(p, i-2) == slope && MP_ALL(p, i-1)
                == mean && MP_ALL(p, i-5) == temp
                D = MP_ALL(p,i)*1000;
            end
        end
    end
    L(TP,1) = D;
end
eval(['HIST_77=L']);


% create new variable
for TP=1:16
    clear("B");
    z = 1;
    for i= 9*TP:144:long(1,2)
```

```
        for p = 1:769
            if MP_ALL(p,i-4) == vin && MP_ALL(p, i-2) == slope && MP_ALL(p, i-5)
                == temp
                B(z,1) = MP_ALL(p,i)*1000;
                z = z + 1;
            end
        end
    end
    BUFF(1:16,TP) = B;
end
eval(['HIST_ALL_MEAN=BUFF']);


MEAN_BEST = [1 8 593.1715;
          2 6 596.4392;
          3 5 595.4355;
          4 6 588.5502;
          5 9 595.0793;
          6 6 588.6542;
          7 7 602.0694;
          8 8 598.5203;
          9 5 596.7370;
          10 6 589.3895;
          11 5 587.9028;
          12 7 593.3469;
          13 7 594.3208;
          14 9 588.4419;
          15 8 589.9085;
          16 8 589.9023];

f=figure('Name','Gaussian-distribution␣of␣TP','NumberTitle','off'); grid on
hold on

bins = 8;
A = sort(HIST_77);
pa = fitdist(A,'Normal');
AH = histfit(A,bins,'normal');
set(AH(1),'FaceAlpha',.5);
set(AH(1),'FaceColor','red');
set(AH(2),'Color','red');

T = sort(MEAN_BEST(:,3));
pt = fitdist(T,'Normal');
TH = histfit(T,bins,'normal');
set(TH(1),'FaceAlpha',.5);
set(TH(1),'FaceColor','blue');
set(TH(2),'Color','blue');

title("Distributions␣at␣"+temp+"C,␣VDDA:␣"+vin+"V,␣slope:␣"+slope+"");
xlabel('Vout␣[mV]');
```

```
ylabel('counts');

hist1 = sprintf("Vout␣slope␣=␣mean␣=␣7:␣\\mu␣=␣%3.2f,␣\\sigma␣=␣%3.2f",pa.mu,
    pa.sigma);
hist2 = sprintf('␣␣␣␣␣␣␣␣␣␣␣Vout␣best␣mean:␣\\mu␣=␣%3.2f,␣\\sigma␣=␣%3.2f',
    pt.mu, pt.sigma);
legend(hist1,'',hist2,'')
legend('Location','northeast')
box on;
ax = gca;
%ax.XAxis.FontSize = fontsize;
%ax.YAxis.FontSize = fontsize;
ax.Legend.FontSize = 12;
f.Position = [400 280 900  550];
hold off

head = {'TP', 'mean', 'Vout'};
BEST = [head; num2cell(MEAN_BEST)];
writecell(BEST,'Data/HIST_BEST_SIGMA.csv');

header = {'Vout'};
BEST = [header; num2cell(HIST_77)];
writecell(BEST,'Data/HIST_77.csv');


% export Data
if temp > 0
exportgraphics(gcf,('Data/HIST_BEST_SIGMA.pdf'),'ContentType','vector');
else
t = temp*-1;
exportgraphics(gcf,('Data/HIST_BEST_SIGMA.pdf'),'ContentType','vector');
end

%%
% !!!! Run section "find␣best␣TC" before you run this script!!!!
% set mean best as condition and create an array with all TCs at this mean
% find min of TC and write TP, slope and mean in matri
% use fixed means and find best sigma (lowest TC)

long = size(TC_ALL)

for TP = 1:16
    z=1;
    step = 3*TP;
    for p = 1:long(1,1)
        if TC_ALL(p,step) == MEAN_BEST(TP,2)
            TC_MEAN(z,[1:3]) = TC_ALL(p,[step-2 step-1 step]);
            z = z + 1;
        end
    end
```

```
1200     [min_tc row] = min(TC_MEAN(:,1));
1201     TC_BEST_CONFIG(TP,[1 2 3]) = (TC_MEAN(row,[1 2 3]));
1202 end
1203 header = {'TC', 'slope','mean'};
1204 CELL = [header; num2cell(TC_BEST_CONFIG)];
1205 writecell(CELL,'Data/TC_BEST_CONFIG.csv');
1206
1207 % create new variable
1208 for TP=1:16
1209     z = 1;
1210     clear("J");
1211     for i= 9*TP:144:1728
1212         for p = 1:769
1213             if MP_ALL(p,i-2) == TC_BEST_CONFIG(TP,2) && MP_ALL(p,i-1) ==
                    TC_BEST_CONFIG(TP,3) && MP_ALL(p,i-4) == 1.2
1214                 J(z,1:4) = MP_ALL(p,[i i-5 i-2 i-1]);
1215                 eval(['TP' num2str(TP) '_BEST_CONFIG=J']);
1216                 z = z+1;
1217             end
1218         end
1219     end
1220 end
1221
1222
1223 % plot best config / temp
1224 clc
1225 close all
1226 f=figure('Name','bandgap characterization','NumberTitle','off');grid on, grid
       minor
1227 hold on
1228
1229 for TP = 1:16
1230     Data = eval(['TP' num2str(TP) '_BEST_CONFIG']);
1231     plot(Data(:,2),Data(:,1)*1000,"LineWidth",1.5,'Marker','.','MarkerSize',18,
1232     'LineStyle','-','Color',color(TP,:));
1233     ylabel('bandgap [mV]')
1234     xlabel('temperature [C]')
1235     xlim([-40 70])
1236     title("All TPs at best config.: VDDA = 1.2V; Temp = 30C");
1237     labels=num2str((1:16).','TP%d');
1238 legend(labels);
1239 legend('Location','eastoutside')
1240 box on
1241 ax = gca;
1242 fontsize = 12;
1243 ax.XAxis.FontSize = fontsize;
1244 ax.YAxis.FontSize = fontsize;
1245 ax.Legend.FontSize = fontsize;
1246 f.Position = [400 280 900 550];
1247 end
```

```
1248
1249 exportgraphics(gcf,('Data/TP_BEST_CONFIG.pdf'),'ContentType','vector');
1250
1251 %% Compare Distribution of TPs with different means
1252 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1253 % find best config when slope is fixed
1254 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1255
1256 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
1257 clc
1258 close all
1259
1260 temp= 30;
1261 vin = 1.2;
1262
1263 long = size(MP_ALL);
1264
1265 %HIST_BEST_SLOPE in one column
1266 z = 1;
1267 for TP=1:16
1268     for i= 9*TP:144:long(1,2)
1269         for p = 1:769
1270             if MP_ALL(p,i-4) == vin && MP_ALL(p, i-2) == TC_BEST_CONFIG(TP,2) &&
        MP_ALL(p, i-5) == temp
1271                 B(z,1) = MP_ALL(p,i)*1000;
1272                 z = z + 1;
1273             end
1274         end
1275     end
1276 end
1277 eval(['HIST_BEST_SLOPE=B']);
1278
1279 % HIST_BEST_SLOPE in Matix
1280
1281 % for TP=1:16
1282 %     clear("K");
1283 %     z = 1;
1284 %     for i= 9*TP:144:long(1,2)
1285 %         for p = 1:769
1286 %             if MP_ALL(p,i-4) == vin && MP_ALL(p, i-2) == TC_BEST_CONFIG(TP,2)
        && MP_ALL(p, i-5) == temp
1287 %                 K(z,1) = MP_ALL(p,i)*1000;
1288 %                 z = z + 1;
1289 %             end
1290 %         end
1291 %     end
1292 %     BUff(1:16,TP) = K;
1293 % end
1294 % eval(['HIST_BEST_SLOPE=BUff']);
1295
```

```
% find mean with fixed slope closest to mu (600 mV)
SLOPE_BEST = [1 9 605.3012;
              2 6 596.4392;
              3 7 596.0533;
              4 6 597.5514;
              5 15 604.5242;
              6 9 599.3101;
              7 9 597.1830;
              8 6 597.1830;
              9 5 596.7370;
              10 5 596.1081;
              11 6 602.6707;
              12 8 595.1890;
              13 8 597.3212;
              14 15 592.1728;
              15 10 600.1008;
              16 11 594.4658];

f=figure('Name','Gaussian-distribution of TP','NumberTitle','off'); grid on
hold on

bins = 10;
SL = sort(HIST_BEST_SLOPE);
psl = fitdist(SL,'Normal');
Asl = histfit(SL,bins,'normal');
set(Asl(1),'FaceAlpha',.5);
set(Asl(1),'FaceColor','red');
set(Asl(2),'Color','red');

% T = sort(MEAN_BEST(:,3));
% pt = fitdist(T,'Normal');
% TH = histfit(T,bins,'normal');
% set(TH(1),'FaceAlpha',.5);
% set(TH(1),'FaceColor','blue');
% set(TH(2),'Color','blue');

title("Distributions at "+temp+"C, VDDA: "+vin+"V, best slope, mean: 0-15");
xlabel('Vout [mV]');
ylabel('counts');

hist1 = sprintf("Vout best slope, mean 0-15: \\mu = %3.2f, \\sigma =
    %3.2f",psl.mu, psl.sigma);
%hist2 = sprintf('           Vout best mean: \\mu = %3.2f, \\sigma = %3.2f', 
    pt.mu, pt.sigma);
legend(hist1,'')
legend('Location','northeast')
ylim([0 50])
box on;
ax = gca;
%ax.XAxis.FontSize = fontsize;
```

```
%ax.YAxis.FontSize = fontsize;
ax.Legend.FontSize = 12;
f.Position = [400 280 900 550];
hold off

exportgraphics(gcf,('Data/TP_BEST_SLOPE.pdf'),'ContentType','vector');
% head = {'TP', 'mean', 'Vout'};
% BEST = [head; num2cell(MEAN_BEST)];
% writecell(BEST,'Data/HIST_BEST_SIGMA.csv');
%
% header = {'Vout'};
% BEST = [header; num2cell(HIST_77)];
% writecell(BEST,'Data/HIST_7.csv');

%%
% !!!! Run section "find best TC" before you run this script (line: 536)!!!!
% set slope best as condition and create an array with all TCs at this mean
% find min of TC and write TP, slope and mean in matri
% use fixed means and find best sigma (lowest TC)

long = size(TC_ALL);

% check where TC is min for given config
for TP = 1:16
    z=1;
    step = 3*TP;
    for p = 1:long(1,1)
        if TC_ALL(p,step) == SLOPE_BEST(TP,2)
            TC_SLOPE(z,[1:3]) = TC_ALL(p,[step-2 step-1 step]);
            z = z +1;
        end
    end
    [min_tc row] = min(TC_SLOPE(:,1));
    TC_BEST_CONFIG_FIX_SLOPE(TP,[1 2 3]) = (TC_SLOPE(row,[1 2 3]));
end

header = {'TC', 'slope','mean'};
FIX_SLOPE = [header; num2cell(TC_BEST_CONFIG_FIX_SLOPE)];
writecell(FIX_SLOPE,'Data/TC_BEST_CONFIG_FS.csv');

% create new variable for best configuration for each TP
for TP=1:16
    z = 1;
    clear("M");
    for i= 9*TP:144:1728
        for p = 1:769
            if MP_ALL(p,i-2) == TC_BEST_CONFIG_FIX_SLOPE(TP,2) && MP_ALL(p,i-1)
      == TC_BEST_CONFIG_FIX_SLOPE(TP,3) && MP_ALL(p,i-4) == 1.2
                M(z,1:4) = MP_ALL(p,[i i-5 i-2 i-1]);
                eval(['TP' num2str(TP) '_BEST_CONFIG_FS=M']);
```

```
                z = z+1;
            end
        end
    end
end


% plot best config / temp
clc
close all
f=figure('Name','bandgap characterization','NumberTitle','off');grid on, grid
    minor
hold on

for TP = 1:16
    Data = eval(['TP' num2str(TP) '_BEST_CONFIG_FS']);
    plot(Data(:,2),Data(:,1)*1000,"LineWidth",1.5,'Marker','.','MarkerSize',18,
    'LineStyle','-','Color',color(TP,:));
    ylabel('bandgap [mV]')
    xlabel('temperature [C]')
    xlim([-40 70])
    title("All TPs at best config. tuned to 600 mV: VDDA = 1.2V; Temp = 30C");
    labels=num2str((1:16).','TP%d');
    legend(labels);
    legend('Location','eastoutside')
    box on
    ax = gca;
    fontsize = 12;
    ax.XAxis.FontSize = fontsize;
    ax.YAxis.FontSize = fontsize;
    ax.Legend.FontSize = fontsize;
    f.Position = [400 280 900  550];
end

exportgraphics(gcf,('Data/TP_BEST_CONFIG_FIX_SLOPE.pdf'),'ContentType','vector');

% plot distribution of best configuration

DATA = [TP1_BEST_CONFIG_FS; TP2_BEST_CONFIG_FS; TP3_BEST_CONFIG_FS;
    TP4_BEST_CONFIG_FS; TP5_BEST_CONFIG_FS;...
        TP6_BEST_CONFIG_FS; TP7_BEST_CONFIG_FS; TP8_BEST_CONFIG_FS;
            TP9_BEST_CONFIG_FS; TP10_BEST_CONFIG_FS;...
        TP11_BEST_CONFIG_FS; TP12_BEST_CONFIG_FS; TP13_BEST_CONFIG_FS;
            TP14_BEST_CONFIG_FS; TP15_BEST_CONFIG_FS; TP16_BEST_CONFIG_FS];

f=figure('Name','Distribution of TPs','NumberTitle','off'); grid on
hold on

bins = 12;
DAT = sort(DATA(:,1)*1000);
```

```
1439 pd_d = fitdist(DAT,'Normal');
1440 Ad = histfit(DAT,bins,'normal');
1441 set(Ad(1),'FaceAlpha',.5);
1442 set(Ad(1),'FaceColor','red');
1443 set(Ad(2),'Color','red');
1444
1445 title("Distributions␣at␣"+temp+"C,␣VDDA:␣"+vin+"V,␣best␣slope/mean");
1446 xlabel('Vout␣[mV]');
1447 ylabel('counts');
1448
1449 hist1 = sprintf("Vout␣best␣slope/␣mean:␣\\mu␣=␣%3.2f,␣\\sigma␣=␣%3.2f",pd_d.mu,
       pd_d.sigma);
1450 legend(hist1,'')
1451 legend('Location','northeast')
1452
1453 box on;
1454 ax = gca;
1455 fontsize = 12;
1456 ax.XAxis.FontSize = fontsize;
1457 ax.YAxis.FontSize = fontsize;
1458 ax.Legend.FontSize = fontsize;
1459 f.Position = [400 280 900  550];
1460 hold off
1461
1462 exportgraphics(gcf,('Data/HIST_FIXED_SLOPE.pdf'),'ContentType','vector');
```

Code 5: Data analysis of measurements

E Layout of the testboard

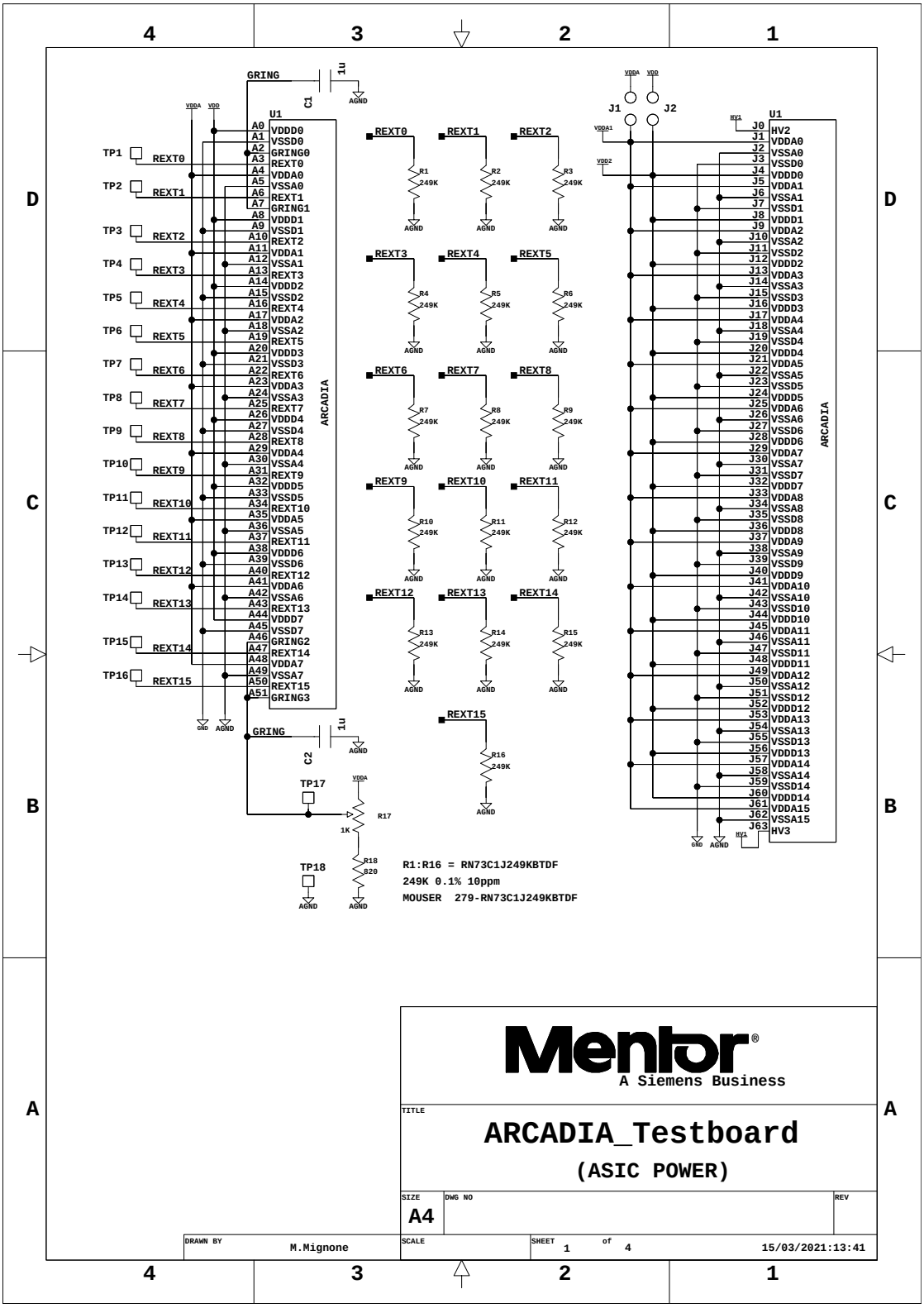


Figure 54: Schematic of the arcadia testboard (ASIC power).

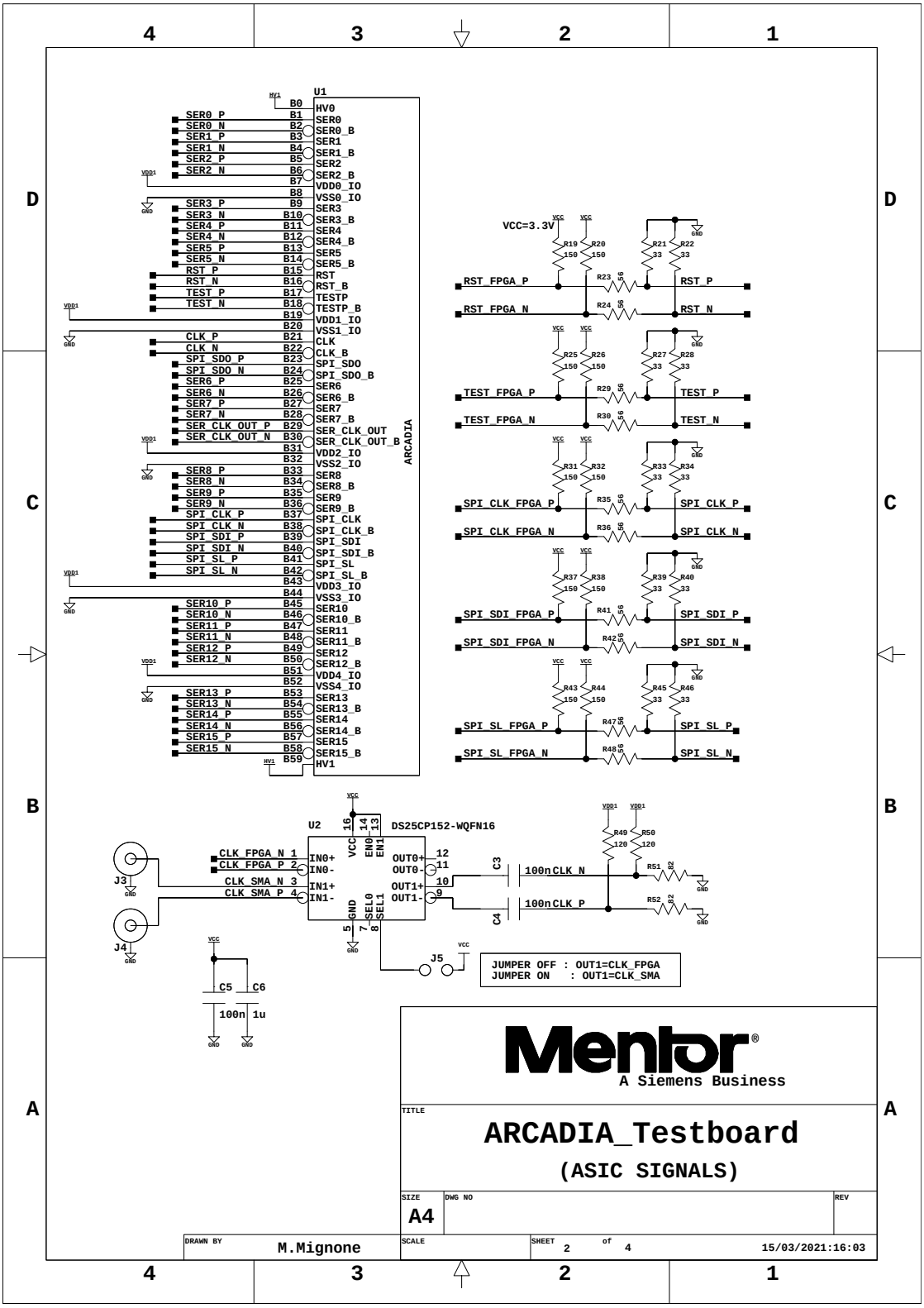


Figure 55: Schematic of the arcadia testboard (ASIC signals).

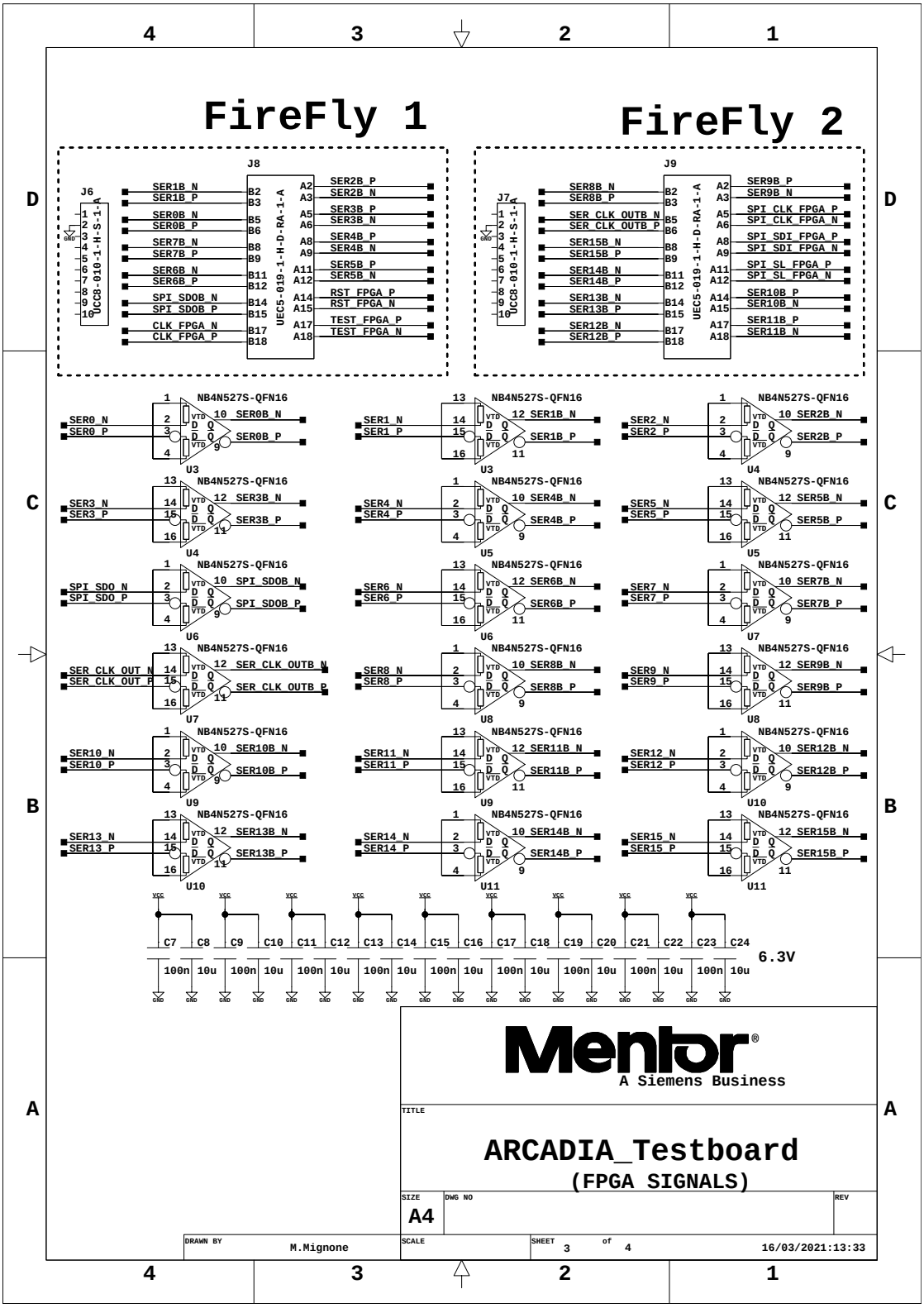


Figure 56: Schematic of the arcadia testboard (fpga signals).

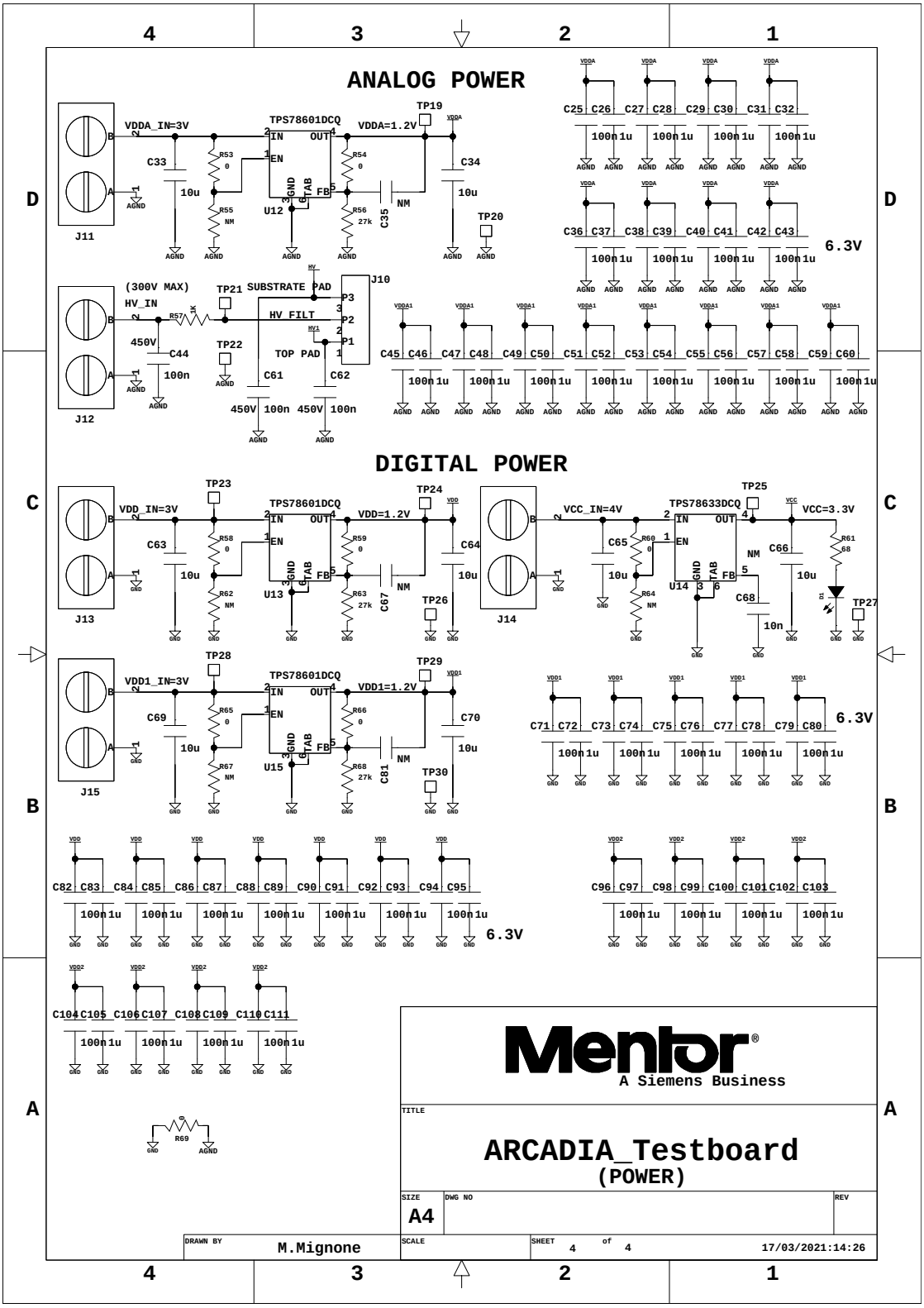


Figure 57: Schematic of the arcadia testboard (power).

F Raw data

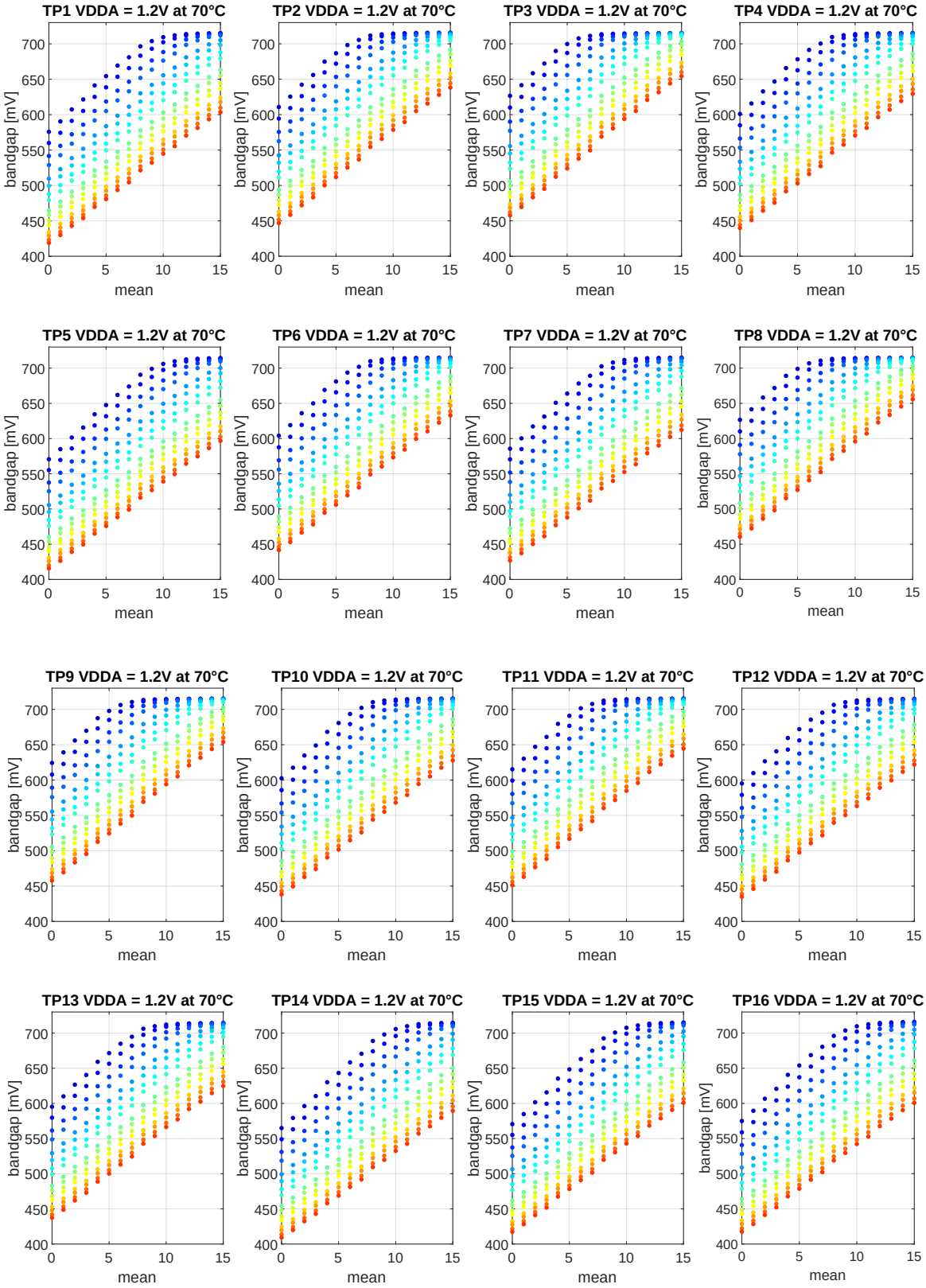


Figure 58: BGR voltage over mean settings of all 16 TPs at 70 °C. The slope is represented in different colors.

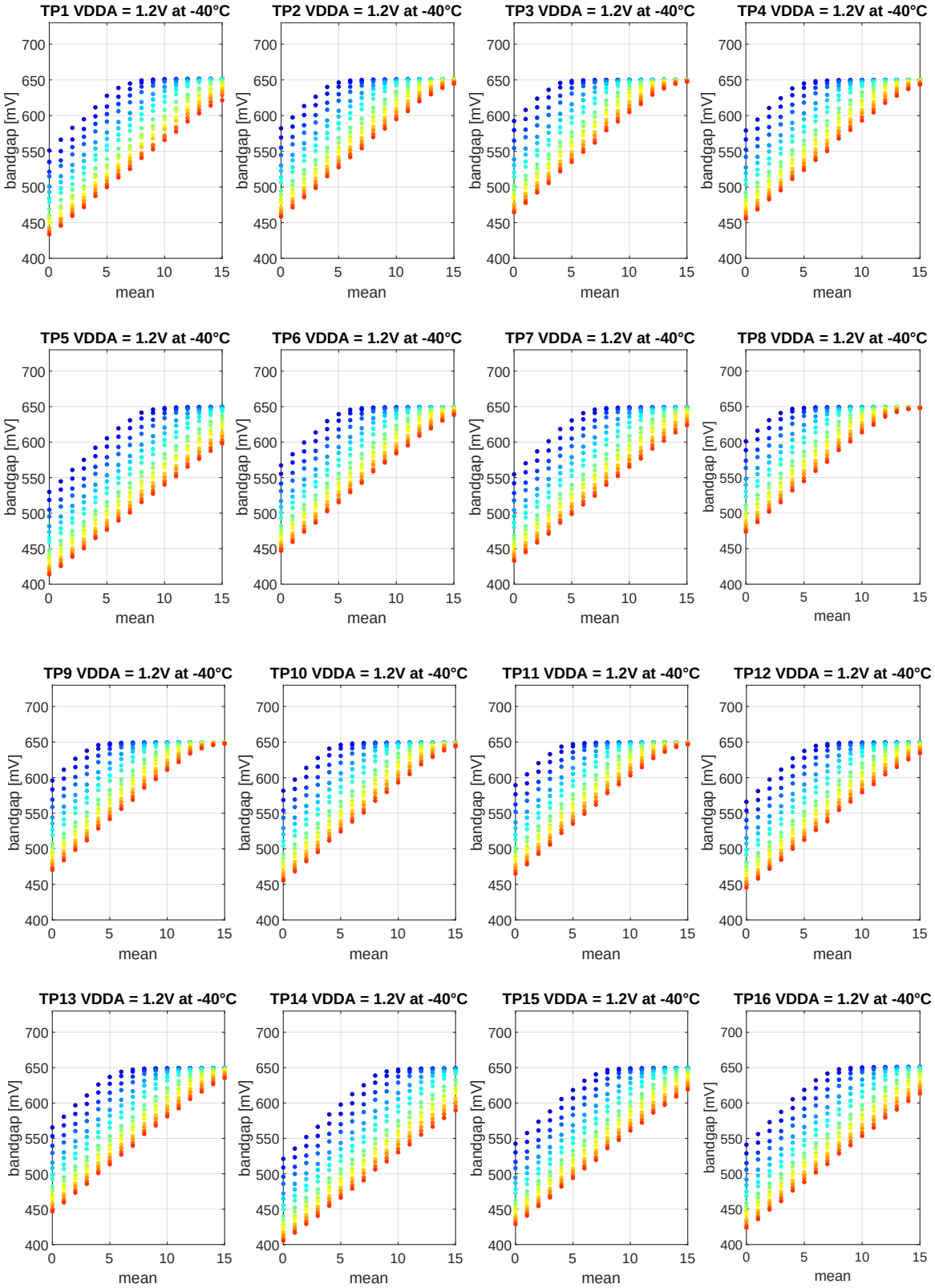


Figure 59: BGR voltage over mean settings of all 16 TPs at –40 °C. The slope is represented in different colors.

G Trimming

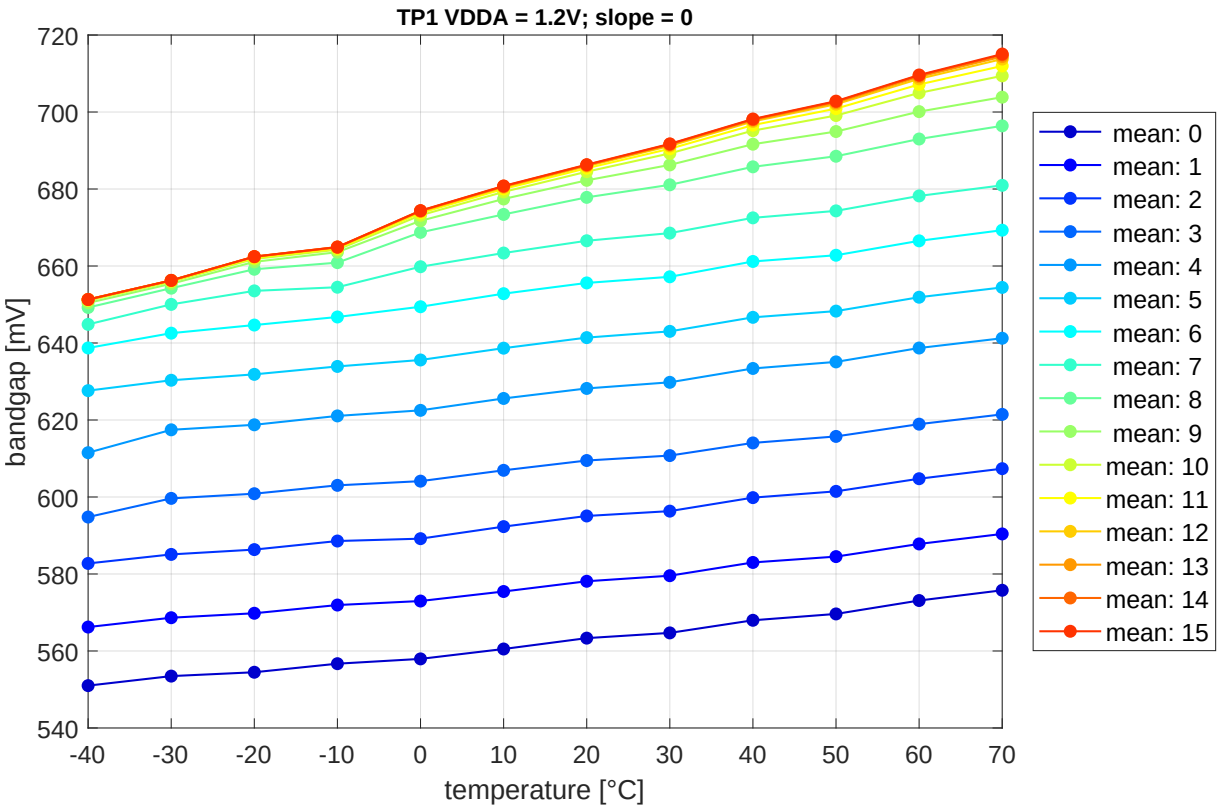


Figure 60: BGR voltage over temperature of TP1 for slope setting 0. The mean settings [0:15] are represented in different colors. The step widths are very small for mean settings higher than 10.

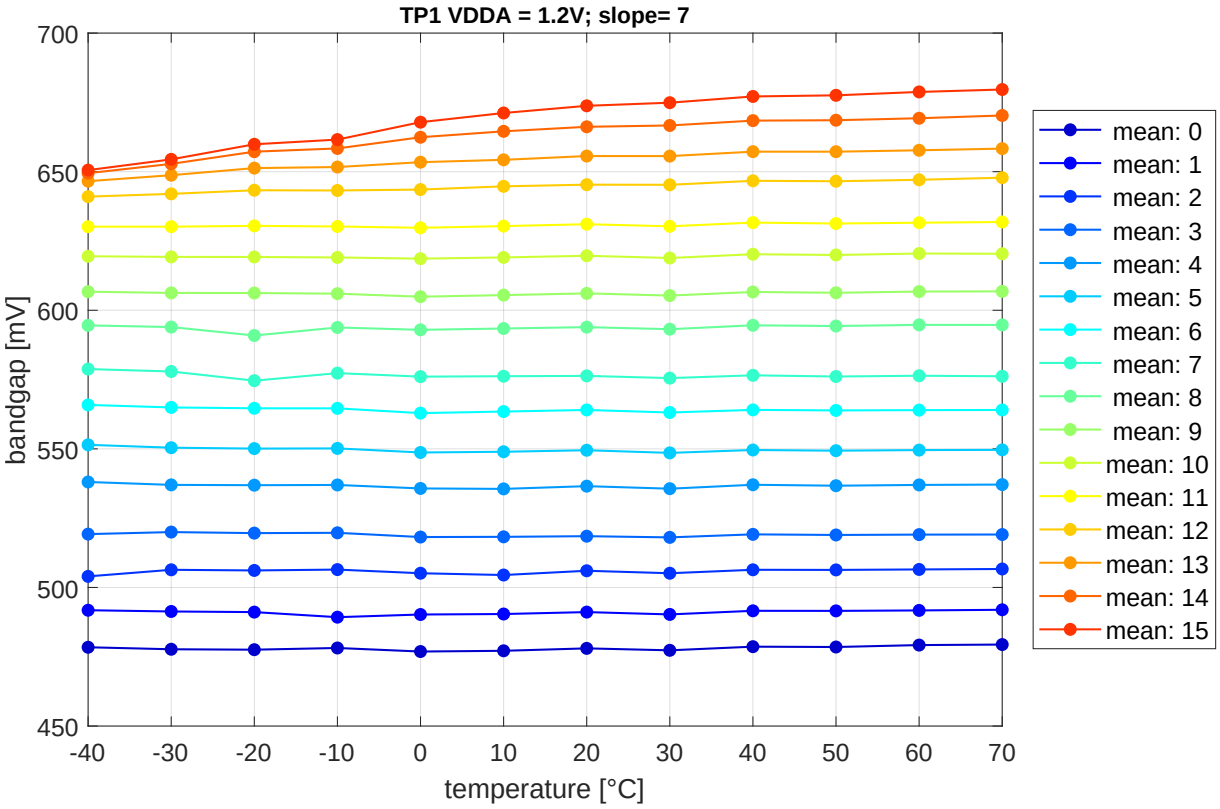


Figure 61: BGR voltage over temperature of TP1 for slope setting 7. The mean settings [0:15] are represented in different colors. The step widths are very small for mean settings higher than 13.

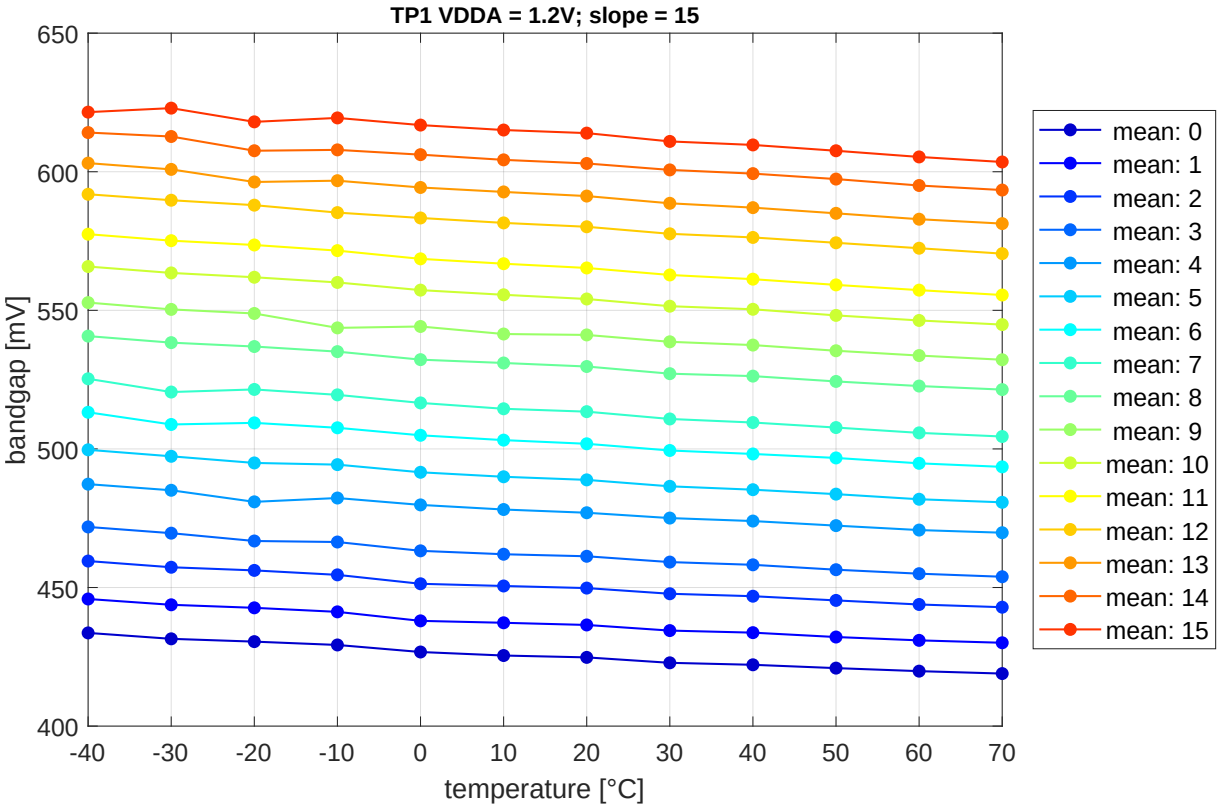


Figure 62: BGR voltage over temperature of TP1 for slope setting 15. The mean settings [0:15] are represented in different colors. The step widths are approximately the same between all mean settings.

H Linearity

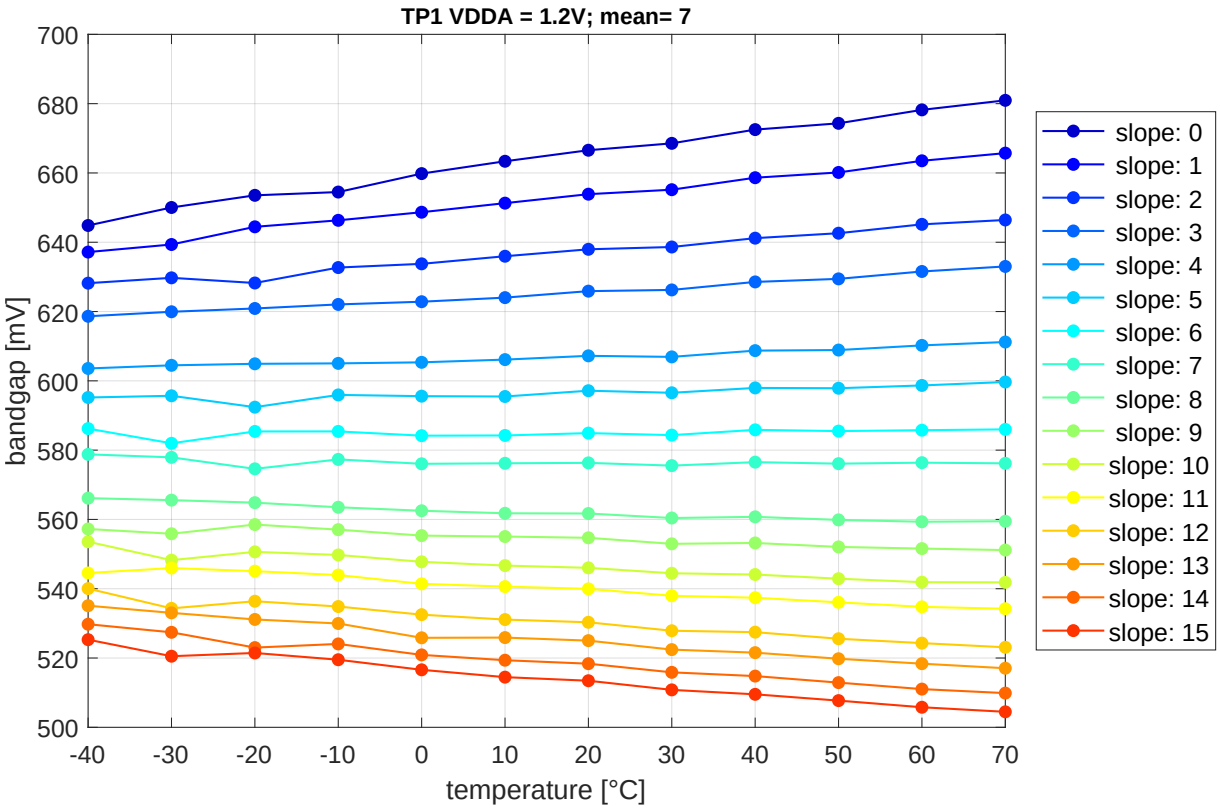


Figure 63: BGR voltage over temperature TP1 including at configuration: mean = 7.

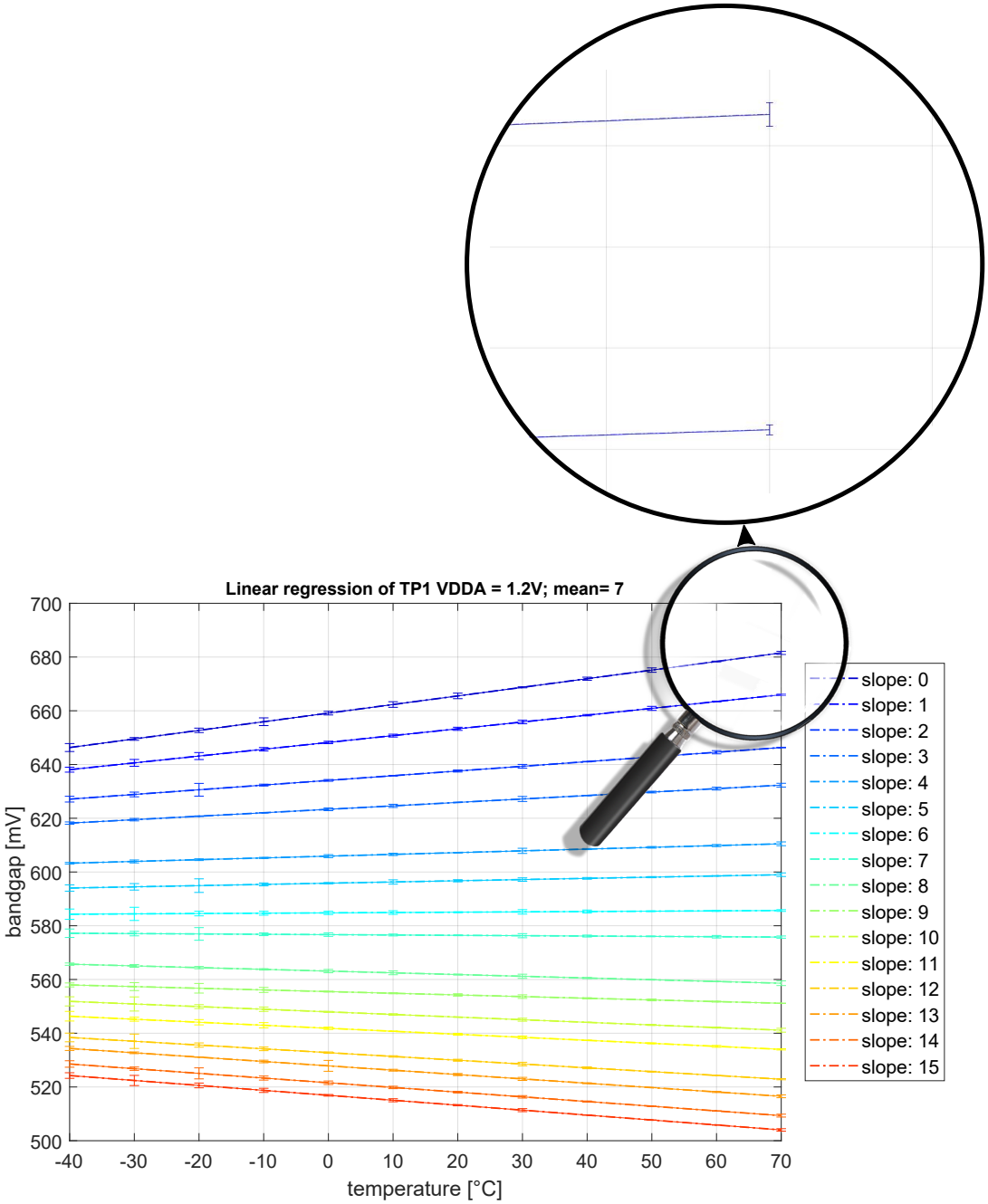


Figure 64: BGR voltage over temperature linear regression of TP1 at configuration: mean = 7 including error bars. Small *y-errors* mean high degree of linearity. Since the *y-errors* are so small, only the errors were shown in the evaluation.

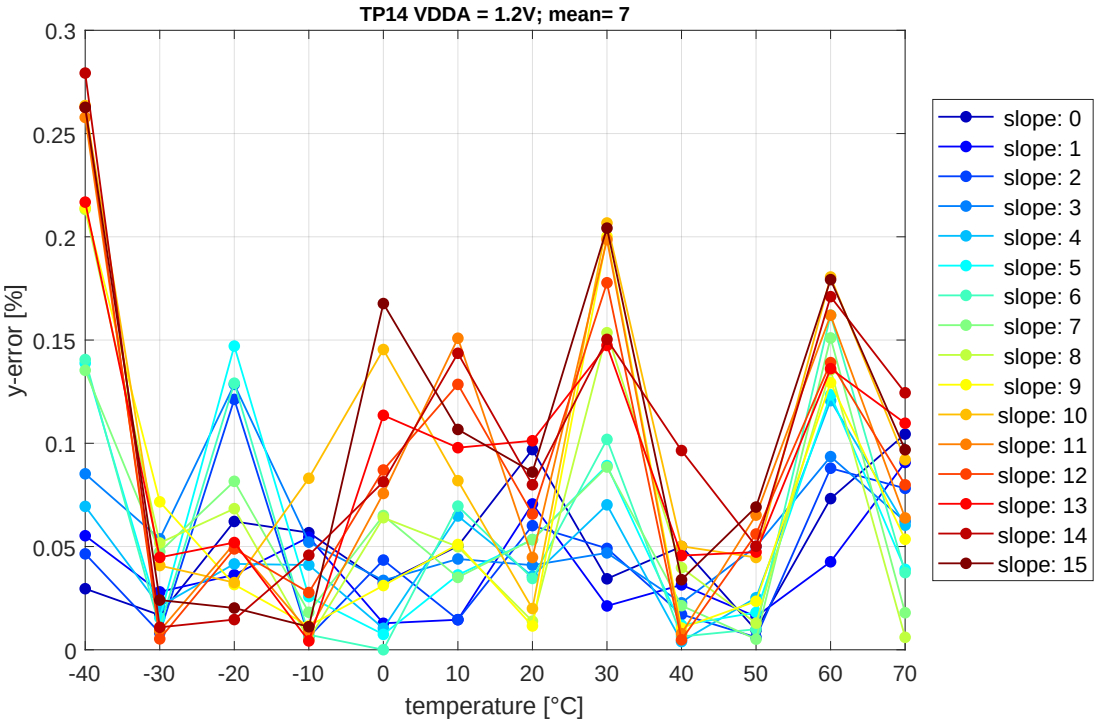


Figure 65: Distribution of y-errors in percent between measurement and linear regression of TP14 at configuration: mean = 7.

I Dependence of the output voltage on the input voltage

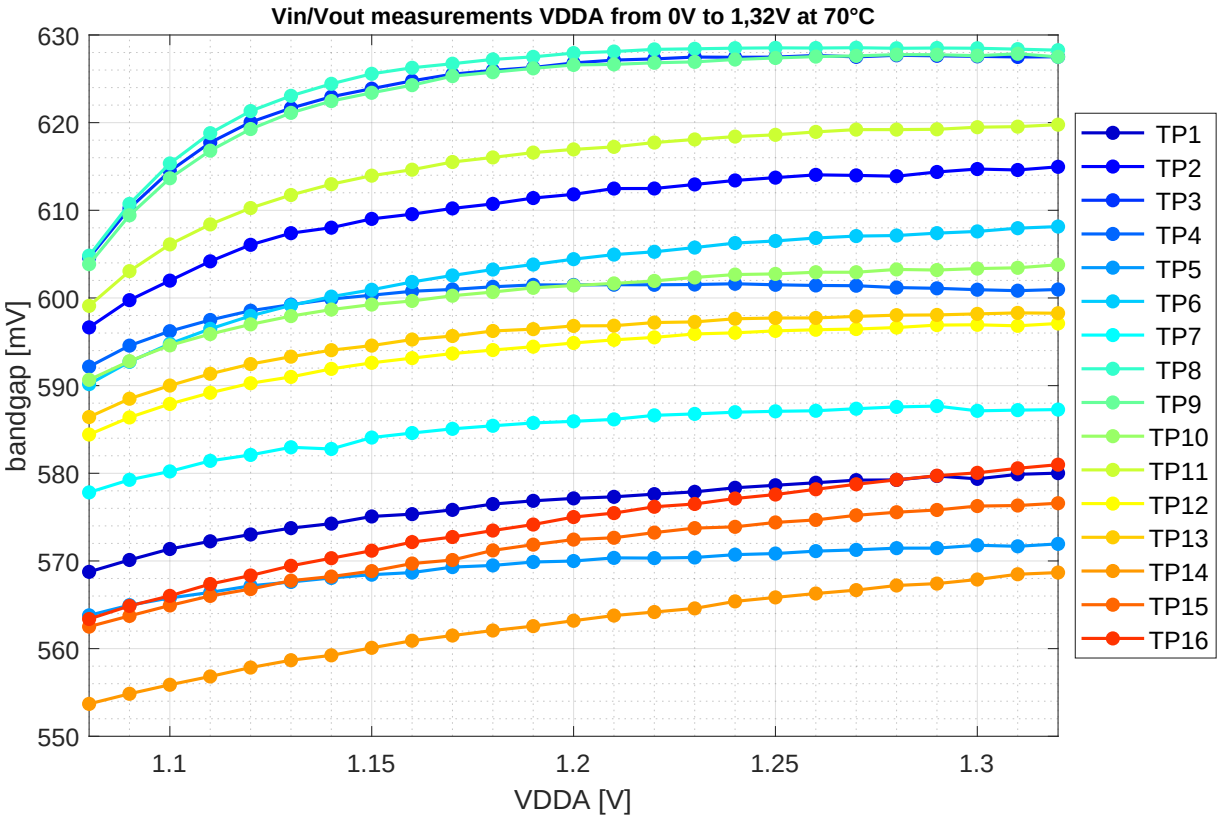


Figure 66: V_{in}/V_{out} curve of all TPs at 70 °C. Input voltage was increased from 0 V to 1.32 V in steps of 10 mV.

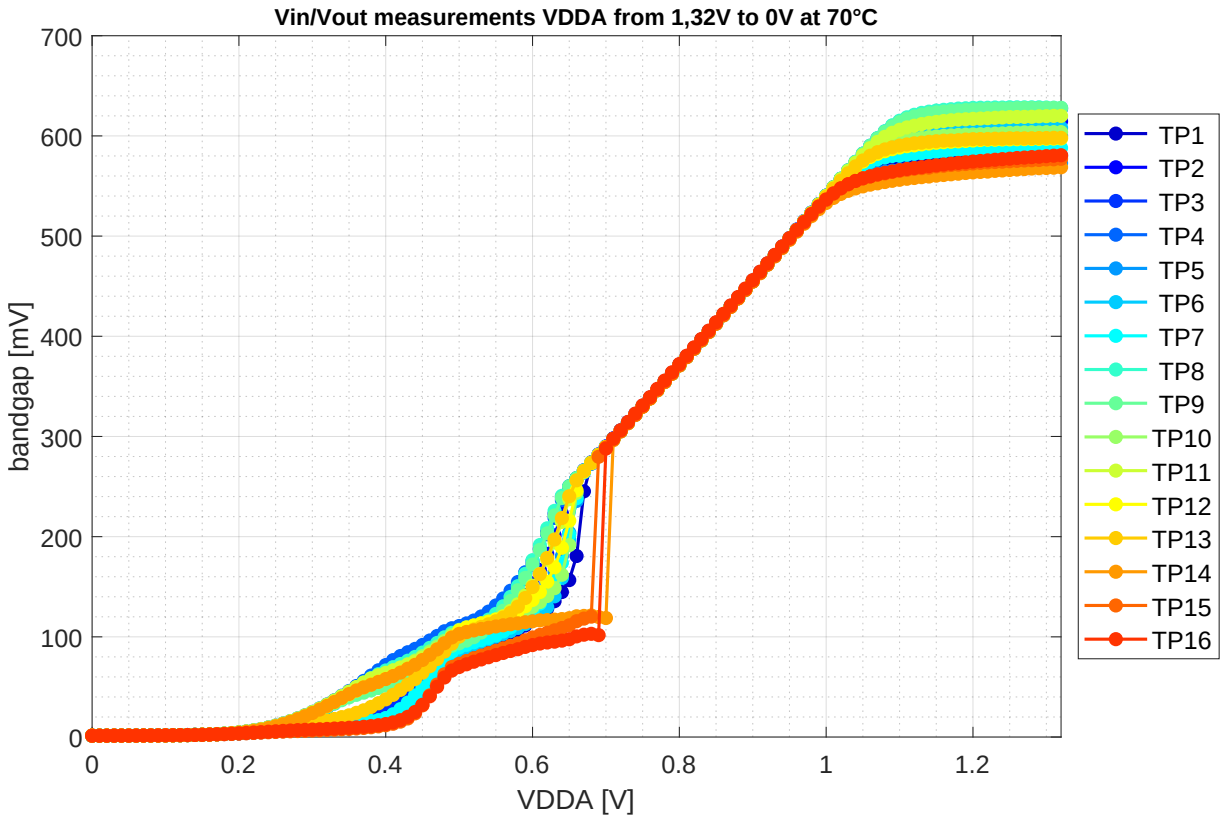


Figure 67: Vin/Vout curve of all TPs.