

Easy Current Slope Detection for Low Cost Implementation of the Direct Adaptive Current Control for DC-DC-Converters

Andreas Liske, Fabian Stamer, Michael Braun

Karlsruhe Institute of Technology (KIT)

Elektrotechnisches Institut (ETI) - Electrical Drives and Power Electronics

Kaiserstr. 12, 76131 Karlsruhe, Germany

Phone: +49 (0) 721-608-46848

Andreas.Liske@kit.edu, www.eti.kit.edu

ABSTRACT

"Direct Adaptive Current Control" (DACC) is a completely adaptive current control scheme with excellent control quality and high dynamics. It is based on the detection of the current slopes during each switching state of the utilized inverter and identifies the system within one period. One method to measure the current slopes is fast oversampling of the current followed by a least squares estimator to filter the noise. This is well-proven but is difficult at high switching frequencies and implies the use of powerful hardware.

In this paper a new algorithm for the detection of the current slopes in DC/DC-Converters is presented that does not need oversampling or a complex filter. Its simplicity allows for easy implementation in low-cost microcontrollers and operation at high switching frequencies. The presented algorithm enables the implementation of the DACC as a low-cost, high dynamic and completely adaptive current control scheme for fast-switching DC/DC-Converters.

I. INTRODUCTION

In recent years many advanced current control strategies have been investigated and published. Most of them can be classified into linear, hysteresis, sliding mode, fuzzy and predictive control. In [1] the field of predictive control is further broken down into hysteresis based, trajectory based, dead beat and model predictive control. Especially model predictive control (MPC) [2] offers great possibilities since even complex physical side-effects that are often neglected can be controlled.

In [3], [4] the Direct Adaptive Current Control (DACC) scheme has first been presented for the control of a DC/DC-converter and for single phase applications such as the armature current of a DC machine [5]. Later it was adapted for the current control of magnetic isotropic 3-phase synchronous machines (with $L_d = L_q$) [6], magnetic anisotropic synchronous machines ($L_d \neq L_q$) [7] and even for 3-phase

induction motors [8]. It shows excellent control quality and high dynamics. At the same time it does not need any control path parameters, no complex model, no test pulses, no offline calculations or cost functions and the computational effort is comparatively little. The system is identified permanently in every period by measuring the slopes of the current in each of the applied switching states, making the DACC completely adaptive. It can be classified as a parameter-free dead-beat control scheme.

The major basis of this control scheme is the detection of the current slopes during each switching state of the utilized inverter. This can be done in different ways, such as measuring two samples, one shortly after switching on a certain switching state and then again taking one sample shortly before switching it off again. Linear interpolation between those two samples then gives the current slope. To avoid inaccurate measurement values, the sample time must not be too close to the switching time of the power semiconductors. This leads to difficulties at small duty cycles.

In [9] the current slopes are sensed with an analog op-amp differentiator. To avoid distortion that comes from the high gain amplification of noise of the current signal, the corner frequency of this differentiator is limited. Hence the rising and falling edges of the square waveform that represents the current slopes in active and freewheeling switching state are significantly smoothed out. This can be handled by sample-and-holding the current slope signal shortly before duty cycle transitions, because then the settling point is reached. The drawback of this method is that an additional external analog circuit is needed and that the smoothing is a limitation at high switching frequencies.

Another way is to use fast oversampling of the current and running a least-squares-estimator algorithm to filter the noise [5], [7], [8], [10]. This method is reliable but needs powerful hardware for the fast AD-Conversion and the least-squares-estimator algorithm. With the high switching frequencies that

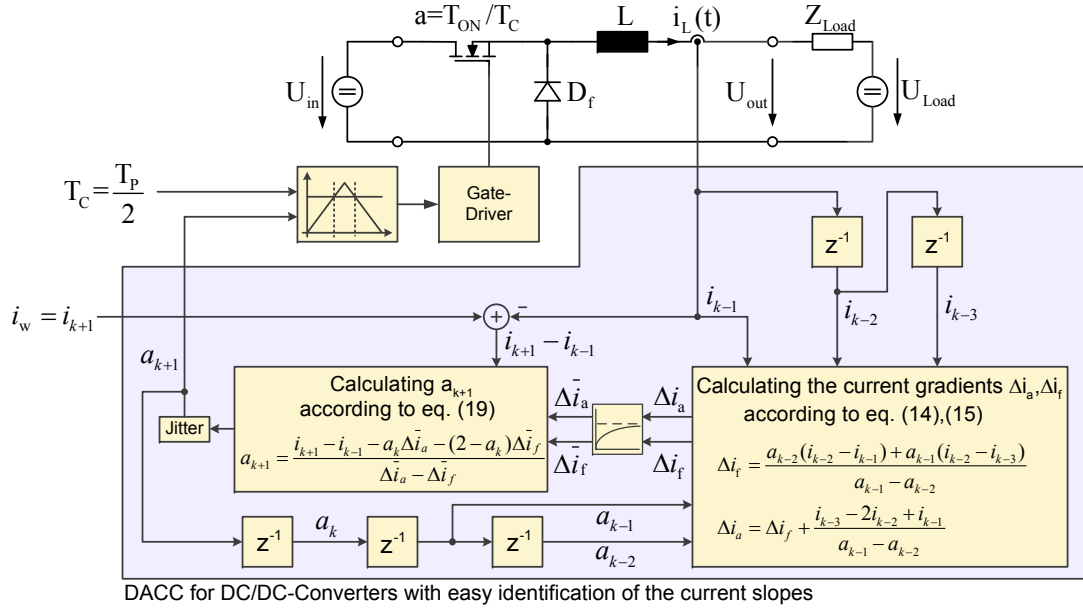


Figure 1. Block diagram of the presented DACC for DC/DC-Converters. In this example a buck converter is shown but the algorithm is suitable for many DC/DC-converter topologies.

are possible and desired in state-of-the-art switch mode power supplies (SMPS), a very fast AD-Conversion is necessary.

In this paper a comparatively simple algorithm is presented that allows for the detection of the current slopes without additional hardware, fast oversampling or a complex filter algorithm. Hence it was named "Easy Current Slope Detection" (ECSD). The measurement of the current is done like in standard pulse width modulation (PWM) applications with a triangle modulation signal. In each period just two samples are taken, one when the triangle signal is zero and the other, when it equals the maximum (see also Fig. 2). Those sample times are farthest from the duty cycle transitions and hence distortion by switching effects is minimal. Since only one sample is taken in each switching state this algorithm marks the minimum of what is possible in shortening the measurement time. That makes it most suitable for high switching frequency applications.

After a short introduction to the DACC control scheme in section II, the new algorithm to identify the current slopes is developed and presented in section III. Since the algorithm is based only on one measurement value per control period, noise and sensitivity is an issue that must be considered, which is done in section IV. Two possibilities to master the noise are presented in the subsections IV-A (DACC with filtered current gradients (DACC-F)) and IV-B (Self-parametrizing deadbeat control). In the following subsection IV-C some aspects concerning the implementation and the timing of the proposed algorithms are discussed. In the concluding sections V and VI simulation and measurement results show that despite noise and disturbance the new ECSD-Algorithm can be used to realize simple, adaptive current control algorithms

without the need of any control path parameters with high dynamics and very good steady state accuracy.

II. DIRECT ADAPTIVE CURRENT CONTROL (DACC) FOR DC/DC-CONVERTERS

The basic working principle of the DACC control scheme for DC/DC-converters is best explained with a buck-converter as depicted in Figure 1.

By operating the switch with the duty cycle $a = T_{ON}/T_C$, the voltage $U_{in} - U_{out}$ is applied to the inductance L. Given that the switching frequency is high enough, the current will rise and fall linear in straight line segments. The current slopes m_a and m_f for each switching state of period k can be assumed to be the same in the following period k+1 if the applied voltage, the inductance and the load are approximately constant for two consecutive periods. This usually is given due to the relatively high switching frequency.

$$\left(\frac{di}{dt}\right)_{a,k+1} \approx \left(\frac{di}{dt}\right)_{a,k} = \frac{U_{in} - U_{out}}{L} = m_a \quad (1)$$

$$\left(\frac{di}{dt}\right)_{f,k+1} \approx \left(\frac{di}{dt}\right)_{f,k} = \frac{-U_{out}}{L} = m_f \quad (2)$$

The index "a" stands for the "active" switching state, when the switch is turned on and the inductance is connected to the input voltage. The index "f" describes the "freewheeling" switching state, in which the switch is open and the inductance discharges into the output capacitor. For reasons explained in section III the control period T_C with period index "k" is set to half the PWM period T_P with period index "p" (see Fig. 2).

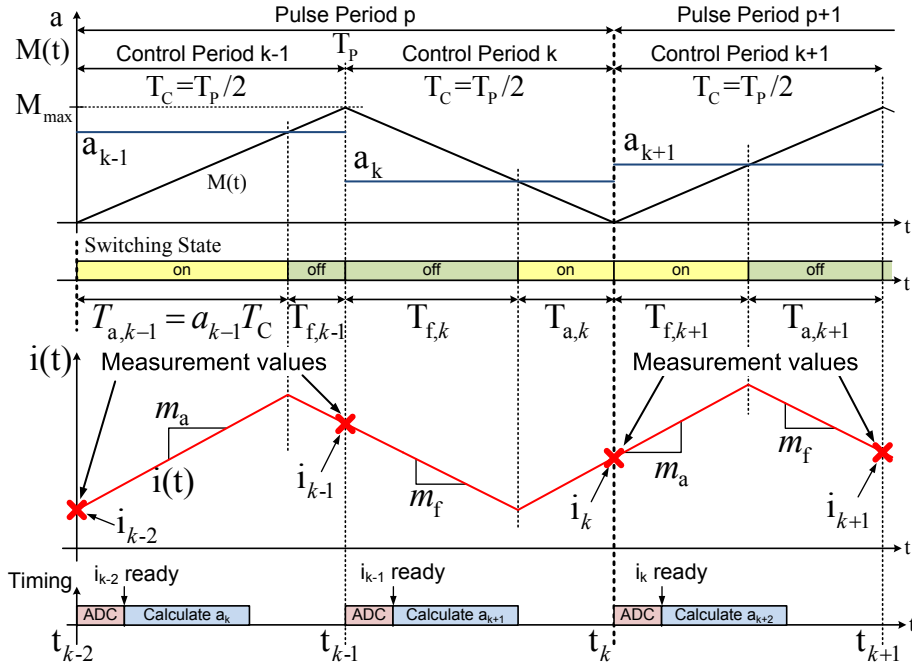


Figure 2. Pulse width modulation pattern with double update of the duty cycle for the "Easy Current Slope Detection"-Algorithm

A given current setpoint i_{k+1} that should be reached at the end of the next period $k+1$ can be calculated with this linear equation that describes the current curve (see Fig. 2):

$$i_{k+1} = i_k + a_{k+1} \cdot T_C \cdot m_a + (1 - a_{k+1}) \cdot T_C \cdot m_f \quad (3)$$

Once the current slopes m_a and m_f of period k are known, the necessary duty cycle to reach the given i_{k+1} can be calculated easily by solving (3) for a_{k+1} :

$$a_{k+1} = \frac{i_{k+1} - i_k - T_C \cdot m_f}{T_C \cdot (m_a - m_f)} \quad (4)$$

The products $T_C \cdot m_f$ and $T_C \cdot m_a$ correspond to the current ripple that occurs if for a complete control period only the freewheeling switching state, respectively the active switching state would be applied:

$$\Delta i_f = T_C \cdot \frac{-U_{out}}{L} = T_C \cdot m_f \quad (5)$$

$$\Delta i_a = T_C \cdot \frac{U_{in} - U_{out}}{L} = T_C \cdot m_a \quad (6)$$

They are further referred to as "current gradients". With this abbreviation, the computational effort in the controller can be reduced and (4) becomes

$$a_{k+1} = \frac{i_{k+1} - i_k - \Delta i_f}{\Delta i_a - \Delta i_f} \quad (7)$$

With this simple equation the new duty cycle for the following period can be calculated. No control parameters like with a PI-controller are necessary. The knowledge of control path parameters like the absolute values of the voltages U_{in} or U_{out} , the inductance L and the load is also not needed. Only the

last two current samples and the current gradients are used. If the current slopes are detected in every period, this control strategy is completely adaptive. Although in this paper the DACC is presented in the application of a buck converter, it is not limited to this topology. The requirements are just that the current between two switching states can be assumed to be a straight line and that the current gradients are approximately constant for two consecutive control periods.

III. EASY CURRENT SLOPE DETECTION (ECSD) - PROPOSED NEW ALGORITHM TO IDENTIFY THE CURRENT SLOPES

The very basis for the DACC is the detection of the current slopes during each switching state of the utilized power converter. The well-proven method of fast oversampling with subsequent least-squares-estimator [5], [7], [10] requires a sample frequency that is considerably higher than the basic switching frequency of the PWM to get good results especially at small duty cycles. This is not a big issue in powerful inverter-driven motor applications, because switching frequencies of up to some few 10kHz should still be manageable with powerful hardware. But with modern switch mode power supplies (SMPS) with switching frequencies of several 100kHz, this is not practicable any more. Another way to identify the current slopes is necessary to enable the implementation of the DACC control scheme in DC/DC-Converters.

The main contribution of this paper is a new algorithm, named "Easy Current Slope Detection" (ECSD) that identifies the current slopes without oversampling but with a comparatively simple algorithm.

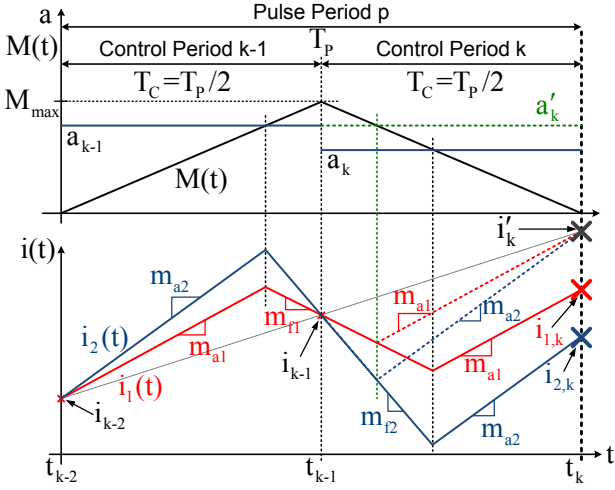


Figure 3. Illustration why the duty cycle must vary from control period to control period to be able to identify the current slopes

The basic idea of this new algorithm is depicted in Fig. 3. A triangle waveform is used as the modulation signal $M(t)$ for the PWM but the duty cycle as the reference signal is updated twice during the PWM period. This results in a control period with $T_C = T_P/2$. The blue and the red current curve $i_1(t)$ and $i_2(t)$ in Figure 3 each represent specific unknown current slopes, which might come from different inductances or different voltages.

If the duty cycle a_{k-1} that is valid during the rising edge of the PWM triangle signal differs from the duty cycle during the falling edge a_k , then the current values at the end of the PWM period $i_{1,k}$ and $i_{2,k}$ depend not only on the applied duty cycles, but also directly on the current slopes (see Fig. 3). If the duty cycle would be constant for a whole PWM period T_P as it is in standard PWM ($a_k = a_{k-1} = a'_k$), no information of the current slopes can be obtained. This is illustrated in Fig. 3 with the dotted green, blue and the red lines. The current value at the end of this period would then be i'_k for both curves.

To ensure that no consecutive duty cycles are equal, a simple logic checks the similarity of the new calculated duty cycle to the last one. In case that both are equal or even just relatively close, then the new duty cycle is jittered around the calculated value. The absolute value of the necessary jitter depends on the utilized hardware, mainly on the A/D-converters resolution and the noise in the measurement hardware. In the presented simulation and the hardware testbench the duty cycle is varied by 3%. The effect on steady state accuracy is minor compared to the current ripple amplitude and due to the high switching frequency of 100kHz, which implies a control frequency of 200kHz (see Fig. 4). In steady state operation the jitter can optionally be deactivated. The current slopes are not identified in such an operation mode, but can be assumed to be constant, as long as a constant duty cycle leads to the same current measurement value in the next period. If this isn't given anymore, the jitter can be reactivated and the current slopes are measured again.

In the lower part of Figure 2 the resulting current waveform for a buck converter as depicted in Fig. 1 is illustrated. The measurement of the current is done like in standard PWM applications with a double updated triangle modulation signal. In each PWM period just two samples are taken: one when the triangle signal is zero and the other, when it equals the maximum. Using a sawtooth signal instead of the double updated triangle waveform would lead to inaccurate measurement values, since then the samples would be taken directly during the duty cycle transitions. Those samples are regarded as the current values at the end of their corresponding control period.

The current value i_k at the end of control period k can be expressed by a linear equation similar to (3) with the two preceding current samples i_{k-1} , i_{k-2} and the current slopes m_a and m_f (see Fig. 2):

$$i_k = i_{k-1} + T_{f,k} \cdot m_f + T_{a,k} \cdot m_a \quad (8)$$

$$= i_{k-1} + (1 - a_k) \cdot T_C \cdot m_f + a_k \cdot T_C \cdot m_a \quad (9)$$

$$i_{k-1} = i_{k-2} + T_{a,k-1} \cdot m_a + T_{f,k-1} \cdot m_f \quad (9)$$

$$= i_{k-2} + a_{k-1} \cdot T_C \cdot m_a + (1 - a_{k-1}) \cdot T_C \cdot m_f$$

Again, the products $T_C \cdot m_f$ and $T_C \cdot m_a$ can be substituted by Δi_f and Δi_a (5)(6), leading to shorter equations:

$$i_k = i_{k-1} + (1 - a_k) \cdot \Delta i_f + a_k \cdot \Delta i_a \quad (10)$$

$$i_{k-1} = i_{k-2} + a_{k-1} \cdot \Delta i_a + (1 - a_{k-1}) \cdot \Delta i_f \quad (11)$$

Solving this set of two linear equations for the current gradients Δi_f and Δi_a leads to the following equations that are the new ECSD-Algorithm:

$$\Delta i_f = \frac{a_{k-1}(i_{k-1} - i_k) + a_k(i_{k-1} - i_{k-2})}{a_k - a_{k-1}} \quad (12)$$

$$\Delta i_a = \frac{a_{k-1}(i_{k-1} - i_k) + a_k(i_{k-1} - i_{k-2})}{a_k - a_{k-1}} \quad (13)$$

$$+ \frac{i_{k-2} - 2i_{k-1} + i_k}{a_k - a_{k-1}} \\ = \Delta i_f + \frac{i_{k-2} - 2i_{k-1} + i_k}{a_k - a_{k-1}} \quad (14)$$

All parameters of those two equations are known and the current gradients Δi_f and Δi_a based on period k and $k-1$ can be calculated.

IV. IMPLEMENTATION OF THE ECSD-ALGORITHM FOR DIRECT ADAPTIVE CURRENT CONTROL

As mentioned before, the ECSD-algorithm needs different duty cycles in subsequent control periods to be able to identify the current slopes, because otherwise a division by zero would occur (see (12)(13)). The necessary jitter to guarantee this is held little to improve accuracy. So in steady-state operation mode, two subsequent duty cycles will differ only slightly. This leads to almost identical current samples and the difference of two subsequent current samples is very small compared to the whole measuring range. Depending

on the resolution of the utilized AD-converter and the overall measuring range, the differences in i_k and i_{k-1} might just be a few bit. In addition to that, noise of the sensed current signal and of the AD-converter can lead to very noisy calculated current gradients Δi_f and Δi_a .

Due to these influences, the original idea of controlling the current with the DACC-control scheme based on each individual current slope that is calculated from period to period only works with a good hardware setup that guarantees accurate current samples and calculations.

But even with a low-cost hardware setup that outputs noisy calculated current gradients, the ECSD-Algorithm can still be implemented to control dc-dc-converters with excellent control quality. The control algorithm of the DACC has to be modified so that not only one period is used to identify the system. In the following two subsections, two control techniques are described, that make use of the calculated current gradients from the ECSD-Algorithm.

A. DACC with filtered current gradients (DACC-F)

In the original DACC control scheme noisy, inaccurate current gradients directly lead to false duty cycles (see (7)). To still maintain a completely adaptive control-scheme without the need for initial control path parameters or additional measurement, the current gradients Δi_f and Δi_a can be filtered with a simple first-order low-pass-filter. The filtered current gradients $\bar{\Delta i}_f$ and $\bar{\Delta i}_a$ are then input to the DACC control algorithm. This algorithm is further referred to as "DACC-F":

$$a_{k+1} = \frac{i_{k+1} - i_k - \bar{\Delta i}_f}{\bar{\Delta i}_a - \bar{\Delta i}_f} \quad (15)$$

After initial settling time, the smoothed $\bar{\Delta i}_f$ and $\bar{\Delta i}_a$ lead to current setpoint step responses that show deadbeat characteristics, as long as the voltages are constant (see Fig. 4). Without any voltage feed-forward, a load step though can only be compensated with the time-constant of the low-pass-filter, since the new $\bar{\Delta i}_f$ and $\bar{\Delta i}_a$ have to reach their new settling point again. Simulation results of this algorithm with highly noisy calculated current gradients are presented in section V. Though the time constant of the low-pass-filter had to be set relatively high due to the simulated noise, the control quality is pretty good after initial settling time. Disturbance response of the DACC-F control scheme could drastically be improved by implementing voltage feed forward of U_{in} and U_{out} .

B. Self-parametrizing deadbeat control

If the inductance is roughly known in advance, a self-parametrizing model-based deadbeat-controller can be realized. In many applications the approximate value of the used inductance is known, but for a good deadbeat controller the exact value is essential. A deviation of 30% is not unusual with inductances but not good enough for a good deadbeat controller. One way to realize such a self-parametrizing control strategy is to start with the approximate value and evaluate

the measured Δi_f and Δi_a together with the also measured voltages U_{in} and U_{out} according to the following equation:

$$L = T_C \cdot \frac{-U_{out}}{\Delta i_f} \quad (16)$$

$$L = T_C \cdot \frac{U_{in} - U_{out}}{\Delta i_a} \quad (17)$$

In the realized hardware testbench such a self-parametrizing deadbeat-control was implemented and showed very good results (see section VI).

C. Timing aspects

Setting of the new duty cycle in the same period If the AD-

Conversion of i_k and the calculation of (7) is fast enough, the calculated duty cycle a_{k+1} can be applied within the same period k+1, shortly after the sample i_k was taken without serious restrictions to the possible duty cycle. This is given at lower switching frequencies (e.g. 8kHz) together with fast AD-Converters and realization of the control algorithm in an FPGA. Namely in [4], [11] the DACC was implemented in a similar way and the calculation time was only a few microseconds. If it is possible to implement it like that the control deadtime can be minimized significantly and the adaptivity is maximized. Together with the setpoint value i_{k+1} and the actual value i_k , the two current gradients Δi_f and Δi_a calculated according to (12)(13) are then input to (7) which delivers the duty cycle for the next control period a_{k+1} directly.

Setting of the new duty cycle in the next period By increasing

the PWM switching frequency and at the same time trying to avoid powerful but expensive hardware, the AD-Conversion and the calculation time of the control algorithm take too long to be able to apply the new duty cycle within the same period. As a result, the new duty cycle a_{k+1} for period k+1 has to be calculated in the previous period k, based on the measurement value i_{k-1} and the current duty cycle a_k . This timing is illustrated in Fig. 2 at the bottom. If the actual time is shortly after t_{k-1} , the actual control period is k, the newest measurement value is i_{k-1} and the algorithm calculates the new duty cycle a_{k+1} which is applied in the next control period k+1. Since the actual time is shortly after t_{k-1} , the necessary current value i_k to solve (7) can't be measured yet, and has to be calculated in advance. Substitution of i_k in (7) by (10) leads to the following equation to calculate the necessary duty cycle for this timing scheme to reach i_{k+1} at the end of control period k+1:

$$a_{k+1} = \frac{i_{k+1} - i_{k-1} - a_k \cdot \Delta i_a - (2 - a_k) \cdot \Delta i_f}{\Delta i_a - \Delta i_f} \quad (18)$$

A block diagram of this algorithm is shown in Figure 1.

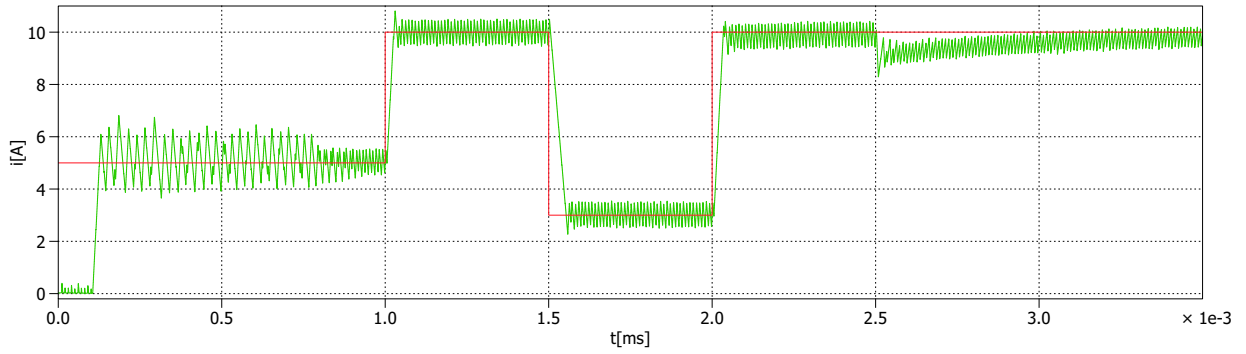


Figure 4. Simulation results of the proposed "Filtered DACC" algorithm. Green: Inductor current i_L , Red: setpoint value i_{k+1} . Simulation setup like in Figure 1 with $U_{in} = 40V$, $U_{out} = 15V$ from start, step up to 30V at 2.5ms, $L = 100\mu H$, $f_{PWM} = 100kHz$. During the first 10 PWM-periods minimal duty-cycles are applied to feed the algorithm with some identification data (Initializing). After that, the control algorithm is released and shows stable but not optimal control behaviour, since the filtered current gradients did not yet reach their settling point. Once this is reached (after approx. 0.8ms) the setpoint-step response is excellent. At 2.5ms a hard output voltage step-up is applied to show the distortion response. Because no voltage feed forward is implemented, the control algorithm doesn't have adjusted current gradients yet. After the time constant of the low-pass-filter, the control is optimal again.

V. SIMULATION RESULTS

The presented control scheme and the new Easy Current Slope Detection (ECS D) algorithm have been implemented in a Matlab/Simulink-simulation to develop and proof the theory. A buck converter together with the presented control scheme as depicted in Fig. 1 was simulated with $U_{in} = 40V$, $f_{PWM} = 100kHz$, $L = 100\mu H$.

The simulation results are shown in Figure 4. The inductor current $i_L(t)$ (green) is shown together with its setpoint value $i_w(t)$ (red). As can be seen in Fig. 1, there are no inputs to the control algorithm except for the setpoint value i_{k+1} and the measured value i_{k-1} . After power up or a reset, the control algorithm does not yet have any values for the current slopes and hence the startup sequence must be self-sufficient. This is done by presetting a minimal duty cycle for the first 10 PWM periods. By doing that, the algorithm can collect valid measurement values and the current slopes can be calculated. After these self-initializing periods, $\Delta \bar{i}_a$ and $\Delta \bar{i}_f$ have not yet reached their settling point, due to the low-pass-filter. The current value is not in steady state condition until 0.8ms, but stable. In fact, the worst that could happen is that due to too little assumed current gradients, the duty cycle would be 100% leading to an overshoot for one period. After this period, the pure P-controller-behaviour of the DACC would correct this again with 100% duty cycle in the other direction. That means that the current ripple would be twice as high as in normal operating mode, but the current controller is inherent stable (see the time from 0.2ms to 0.8ms in Fig. 4). In ordinary designed SMPS, this should not lead to bigger problems. After this time, the filtered $\Delta \bar{i}_a$ and $\Delta \bar{i}_f$ enable an excellent setpoint step response which can be seen at 1ms, 1.5ms and 2ms. At 2.5ms a hard output voltage step from 15V to 30V is applied and acts as output disturbance. Since there is no voltage feed-forward at all, the low-pass-filter takes a little while to adjust the changed $\Delta \bar{i}_a$ and $\Delta \bar{i}_f$.

The influence of the duty cycle jitter can be seen slightly, but is of minor impact. For setpoint steps the control algorithm shows dead-beat behavior and very good steady state accuracy as soon as the $\Delta \bar{i}_a$ and $\Delta \bar{i}_f$ have reached their settling point.

VI. EXPERIMENTAL RESULTS

A hardware testbench was designed and set up to implement the presented control algorithm (see Fig. 6). It was designed as a Half-Bridge with two single SiC-Mosfets from Cree (C2M0080120D) and was operated as a buck-converter with a PWM frequency of 100kHz at $U_{in} = 400V$ and an output current of up to 15A. The current measurement was done with a current sensor from Sensitec (CMS3015) with a primary nominal current of 15A and that has a bandwidth of up to 2MHz. The gate drivers were self-designed and applied +15/-4V to the SiC-Mosfets. The AD-Conversion was done with a 12-bit AD-Converter (THS1206) that was triggered to sample-and-hold only one sample at the described times (see Fig. 2). The control algorithm was implemented in an Altera Cyclone IV FPGA. The inductor was designed to have an inductance of $L=330\mu H$. Since noise was too heavy to implement the DACC as originally intended with no filtering, the ECS D-Algorithm was used to identify the exact value of the inductance L as described in IV.



Figure 6. Hardware testbench to prove the Easy Current Slope Detection Algorithm and its implementation with the presented DACC-F and MPC control algorithms.

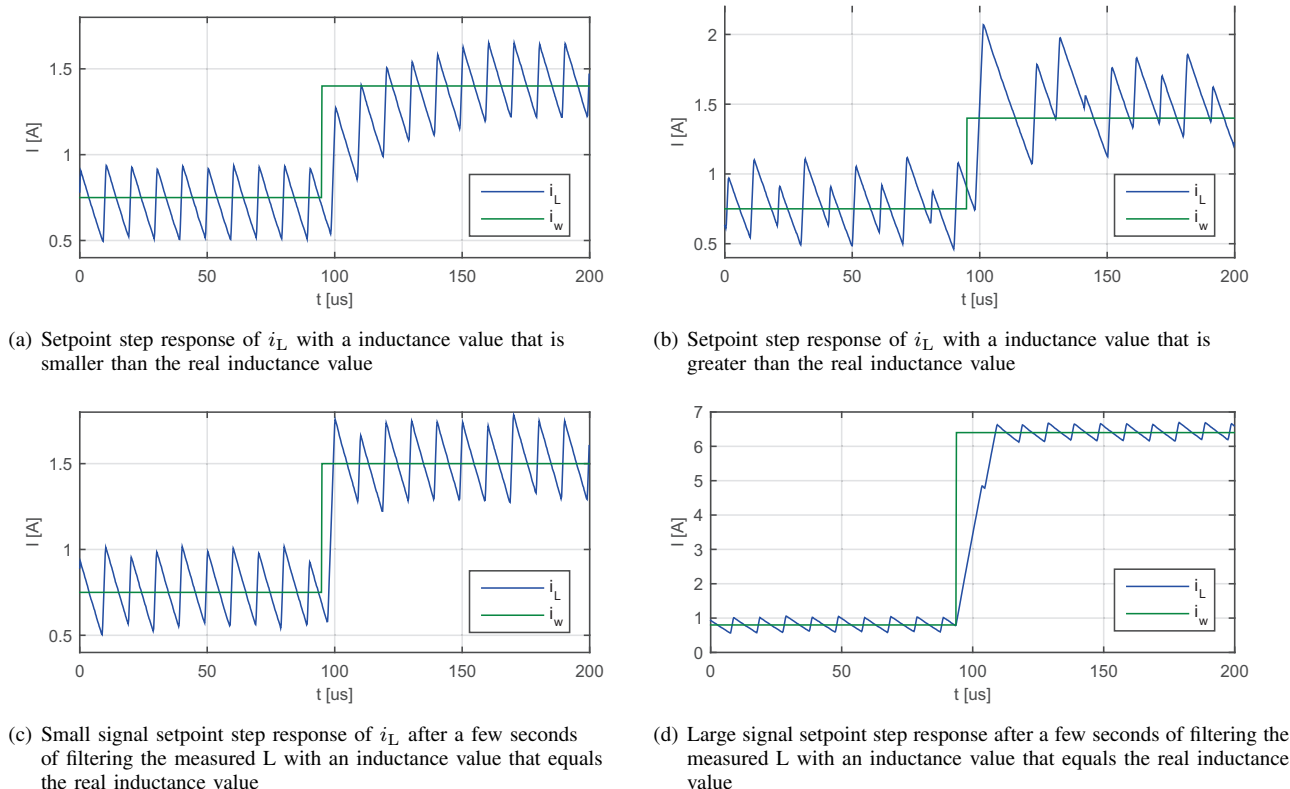


Figure 5. Measurement results of the proposed "Self-parametrizing deadbeat control"

The ECSD-Algorithm itself was implemented as derived in III. The so gained current slopes were used together with the measurement values of U_{in} and U_{out} to calculate the inductance L. Due to the noisy current slopes, the calculated L was very noisy too, but with filtering the real value of $330\mu\text{H}$ could be identified. The initial L that was given to the algorithm was set to a very small value of $200\mu\text{H}$ to avoid heavy overshooting of the current. The duty cycle calculation then was done based on the filtered L and the measured voltages, instead of the measured current slopes. Hence this implemented control scheme is not the DACC any more but a model predictive deadbeat control which is based on the

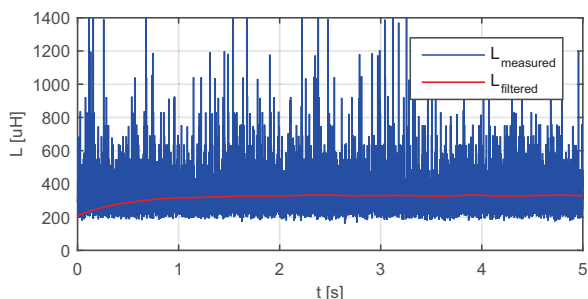


Figure 7. Measurement result: Identification of the inductance L by filtering the cycle-by-cycle calculated L (blue) with the results from the proposed ECSD-Algorithm (red: filtered L)

ECSD-Algorithm to auto-tune its parameter L. In Fig. 7 the measured and the filtered L is depicted. It is obvious that with the cycle-by-cycle calculated L the control algorithm would not work. But once the filtered L is settled, it leads to excellent dynamics (see Fig. 5(c) and Fig. 5(d)).

In Fig. 5(a), 5(b) and 5(c) the setpoint step response of i_L with different inductance values is measured. If the inductance is smaller than the real value, the controller applies smaller duty cycles and the current rises slowly. This is a safe operation mode because no overshoot is likely. If the inductance is assumed to be greater than the real one, the controller applies larger duty cycles and overshooting is the consequence. After a few seconds of filtering the measured L, the filtered L equals the real inductance value and the small signal setpoint step response in Fig. 5(c) as well as the large signal setpoint step response in Fig. 5(d) show excellent deadbeat behaviour. Note that the jitter is still implemented in Fig. 5(c) as well as in Fig. 5(d) but is of minor impact compared to the current ripple.

VII. CONCLUSION

This paper describes a new algorithm (ECSD) for the identification of the current slopes in DC/DC-converters. It is comparatively simple and easy to implement and does not need any external hardware, fast AD-Conversion or complex filter algorithms. Just two samples are taken in each PWM period, at times where disturbance from the switching is minimal. The equations can be programmed in a low-cost microcontroller.

Depending on the hardware noise makes filtering of the gained current slope values necessary. This can be done with a simple first-order low-pass-filter.

The combination of the presented ECSD-Algorithm and the Direct Adaptive Current Control is a low-cost but high-quality control scheme, most suitable for high switching frequencies that does not need any control parameters or external hardware components. It is self-initializing and most SMPS-topologies can be controlled without any changes.

The ECSD-Algorithm is derived from the modulation pattern and different adaptive control schemes that make use of the measured current slopes are presented. The control quality and the dynamics of the DACC with current slope values from the ECSD-Algorithm are demonstrated with simulation results. The implementation of a model predictive deadbeat controller that tunes its parameters with the measured current slopes from the ECSD-Algorithm is described and the measurement results are shown.

REFERENCES

- [1] P. Cortes, M. Kazmierkowski, R. Kennel, D. Quevedo, and J. Rodriguez, "Predictive control in power electronics and drives," *Industrial Electronics, IEEE Transactions on*, vol. 55, no. 12, pp. 4312–4324, dec. 2008.
- [2] A. Linder and R. Kennel, "Model predictive control for electrical drives," in *Power Electronics Specialists Conference, 2005. PESC '05. IEEE 36th*, june 2005, pp. 1793–1799.
- [3] J. Weigold and M. Braun, "Robust predictive dead-beat controller for buck converters," in *Power Electronics and Motion Control Conference, 2006. EPE-PEMC 2006. 12th International*, Aug 2006, pp. 951–956.
- [4] F. Becker, T. Lannert, and M. Braun, "An improved adaptive current control for d.c.-d.c. converters based on current slope detection," in *Industrial Technology (ICIT), 2010 IEEE International Conference on*, 2010, pp. 70–75.
- [5] F. Becker, H. Ennadifi, and M. Braun, "Straightforward current control - one step controller based on current slope detection," in *EPE ECCE-EUROPE 2011 Birmingham*, 2011.
- [6] F. Becker, "Ein neues adaptives verfahren zur hochdynamischen stromregelung," Ph.D. dissertation, Elektrotechnisches Institut, Karlsruher Institut für Technologie (KIT), 2011.
- [7] A. Liske, P. Hofmeier, and M. Braun, "Extended straightforward current control for permanent magnet synchronous machines," in *Power Electronics and Applications (EPE), 2013 15th European Conference on*, 2013, pp. 1–10.
- [8] A. Liske and M. Braun, "Direct adaptive current control - a universal current control scheme for electrical machines," in *The 40th Annual Conference of the IEEE Industrial Electronics Society (IECON), 2014*, 2014, pp. 1–7.
- [9] P.-H. Liu, Y. Yan, F. Lee, and P. Mattavelli, "External ramp autotuning for current mode control of switching converters," in *Applied Power Electronics Conference and Exposition (APEC), 2013 Twenty-Eighth Annual IEEE*, March 2013, pp. 276–280.
- [10] J. Weigold and M. Braun, "Predictive current control using identification of current ripple," *IEEE Transactions on Industrial Electronics*, vol. 55, no. 2, pp. 4346–4353, 2008.
- [11] F. Becker, H. Ennadifi, and M. Braun, "Straightforward current control - one step controller based on current slope detection," *EPE Journal Volume 22-2*, 2012.