Abstract.— The Asymmetrical Half Bridge converter (AHBC) has proven to be a promising candidate for LED lighting applications. It provides high efficiency, galvanic isolation and it can be easily built without electrolytic capacitor. On the other hand, its main drawback is its poor attainable bandwidth. In any two-stage ac-dc LED driver based on the AHBC, the first stage is a Power Factor Corrector (PFC) converter which has to be also implemented without electrolytic capacitor. As a consequence, its output voltage (input voltage of the AHBC) presents a low-frequency ripple. Due to the poor bandwidth of the AHBC, this voltage ripple will be transferred to the converter output voltage, leading to flickering. Due to the complex and non-linear transfer function of the AHBC, any analog feedforward loop has to be tuned for a given operating point, leading to a poor performance when the AHBC moves away from that point. In this paper, a digital feedforward loop is proposed in order to solve this problem. The digital implementation allows the feedforward loop to perfectly cancel the ripple under any condition (e.g., output voltage variation due to dimming). Besides, this digital feedforward loop has been designed and simplified considering the specific purpose of cancelling flickering in the emitted light. In this way, it can be easily implemented in small-size microcontrollers. Experimental results with a 40-W prototype prove the usefulness of the proposed feedforward loop.

Keywords: Asymmetrical Half Bridge, Complementary Control, Feedforward loop, LED Lighting, Flickering.

I. INTRODUCTION

The advantages of LEDs in lighting applications are well-known [1], [2]. Their luminous efficacy is very high. Moreover, their theoretical efficacy (not reached yet) is the highest one. Regarding reliability, the lifetime of LEDs is as long as 50,000 h or even higher. Besides, LEDs also have other advantages that make them the growing trend in lighting. For example, they are environmentally friendly, they are shock resistant [3], precise control of the color and temperature of light [4], etc.

Nevertheless, LED strings cannot be directly connected to the grid [5]. Hence, one of the main issues regarding LEDs is developing power supplies with the same features that LEDs have so that the couple LEDs plus converter keeps all the advantages of those. As a consequence, many topologies with high efficiency and high reliability (i.e., long lifespan [6]) have been proposed in order to supply different kind of LED lamps (retrofit, street lighting, commercial or industrial, etc.). Besides, other common requirements imposed to these power supplies are Power Factor (PF) correction [7], [8] and galvanic isolation [9], [10]. Any lighting device connected to the grid has to comply with EnergyStar or IEC 61000-3-2 Regulations. The galvanic isolation may be a customer’s requirement or something established by local regulations. Obviously, cost is a feature that cannot be set aside. As a consequence, it is possible to classify all the LED drivers depending on the number of stages.

One-stage and integrated two-stage topologies [11]-[17] are used when the cost of the lamp is of primary concern, rather than efficiency or lifetime. On the other hand, two-stage [18], [19] and three-stage [20], [21] topologies (optimized for reaching very high efficiency in spite of the number of power conversions) are preferred when the saving that represents a higher efficiency and a longer lifespan is more important than the saving that represents the cost per unit.

In [22], the use of a two-stage topology is proposed as LED driver for a street-lighting application. The first stage is a PFC boost converter with an efficiency as high as 97%-98%. As it is implemented without electrolytic capacitor, its output voltage is affected by a 15% low-frequency ripple. The second stage, as a consequence, is in charge of avoiding that this ripple reaches the load. The topology chosen for this second stage is the Asymmetrical Half Bridge converter (AHBC) [23]-[25]. This topology is a promising candidate for LED lighting applications, overcoming the disadvantages of the Flyback topology as postregulator. The voltage withstand by its primary switches is clamped to the input voltage. Its output filter is very small, so it can be easily implemented without electrolytic capacitor. Besides, it provides galvanic isolation and its efficiency is very high (it reaches 94%) because Zero Voltage Switching (ZVS) can be easily obtained. Its main problem is that the feedback loop of the AHBC cannot be fast enough to cancel the low-frequency ripple of the PFC boost converter. In [22], a feedforward technique is proposed in order to overcome this problem. This technique effectively reduces the output voltage low-frequency ripple without affecting stability, but it presents two problems:

- The feedforward loop generates a sinusoidal variation of the control voltage (i.e., sinusoidal variation of the duty cycle) for canceling the sinusoidal low-frequency ripple. Due to the non-linear static transfer function of the converter, this is not always the most suitable waveform for this purpose.
- The feedforward loop is tuned for a given operating point. Again, due to the non-linearity, the ripple cancelation is far from being the optimum one when dimming is applied and the AHBC is not in the aforementioned operating point.

In this paper, a digital feedforward loop is presented. The digital implementation is the most suitable one in order to deal with this kind of non-linear systems if they change their operating point (for instance, due to dimming). Nevertheless, the use of high-performance digital systems (such as FPGAs or DSPs) is not always a valid option in
The proposed digital feedforward loop is simplified as it fits into a small-size, low-cost microcontroller (μC). This simplification is based on placing the output-voltage ripple of the LED driver in a frequency range high enough so that it does not involve flickering problems (i.e., it is not perceived by human eyes and, consequently, does not affect human’s health). As the ripple is not cancelled, but only moved, the algorithm is strongly simplified.

There is an additional issue regarding digital control of the AHBC that should be considered. The delays (i.e., dead times) that have to be introduced in the control signals of the AHBC’s MOSFETs in order to achieve Zero-Voltage Switching (ZVS) are not equal and change depending on the operating point of the converter [29]. Therefore, digital control is highly recommendable in this converter as it can be used for dynamically calculating the necessary delays for each situation (increasing its efficiency). The feedforward loop, due to the aforementioned simplification, can be included in the system used to implement these dynamic calculations of the necessary dead times (microcontroller, FPGA or DSP) without a significant increase in the needed resources.

Finally, the proposed loop is valid not only for the AHBC, but for any converter with a non-linear relation between the control variable and the output voltage or current (e.g., the boost converter).

This paper is organized as follows. A brief description of the AHBC topology is provided in section II, including the analysis of the analog feedforward loop and its limitations. The description of the proposed digital feedforward loop is presented in section III while its optimization for minimizing the flickering is explained in section IV. Experimental results are provided in section V. Finally, conclusions will be presented in section VI.

II. REVIEW OF THE AHBC AND THE ANALOG FEEDFORWARD LOOP

This section only pretends to give a brief description of the AHBC, presenting the equations that will be used later in the analysis of the feedforward loop. A deep analysis can be easily found in literature [23]-[25], [29]-[31]. The main issue regarding the AHBC (see Fig. 1) is that the driving signals of the primary-side switches are complementary (the AHBC is also known as HB with complementary control [30], [31]). Due to this, the static transfer function of the AHBC can be expressed as:

\[ V_o = V_{in} (n_1 + n_2) D (1-D), \]  

where \( V_o \) is the output voltage, \( n_1 \) and \( n_2 \) are the turn ratios of the center-tapped transformer and \( D \) is the duty cycle. As can be seen, the relation between the input and the output voltage is linear (for a constant duty cycle). Nevertheless, the relation between the output voltage and the duty cycle is not linear. This has a great impact on the implementation and range of usefulness of any analog feedforward loop.

The input voltage of the AHBC is provided by a PFC converter (in this case a boost converter). Therefore, the average value of this input voltage can be considered regulated under any condition and equal to \( V_{in,nom} \). As the PFC converter is implemented without electrolytic capacitor in order to increase its lifespan, its output voltage (input voltage of the AHBC) can be expressed as:

\[ v_{in}(t) = V_{in,nom} \left[1 + r \sin(2\pi f t)\right], \]  

where \( r \) is the relative peak value of the ripple and \( f \) is the line frequency. If the desired output voltage of the AHBC has to be constant and the input voltage is affected by a low frequency ripple, (1) can be rewritten as:

\[ v_o(t) = V_{in,nom} \left[1 + r \sin(2\pi f t)\right] (n_1 + n_2) d(t) (1-d(t)) \]  

where \( d(t) \) is the time-varying duty cycle, whose value changes in order to counteract the input-voltage ripple and keep the output voltage constant. Hence, considering (1), (2) and that \( n_1 \) and \( n_2 \) are designed according to nominal conditions \( (D_{nom}, V_{O,nom} \text{ and } V_{in,nom},) \), (3) can be used for obtaining the required duty cycle that keeps the output voltage constant:

\[ d(t) = 2 \frac{\sqrt{1 - 4 V_o D_{nom} / (1 - D_{nom})}}{1 + r \sin(2\pi f t) V_{O,nom}} \]  

As can be seen in Fig. 1, the control voltage \( v_c(t) \) satisfies:

\[ v_c(t) = v_{c,p}(t) + v_{c,b}(t), \]  

where \( v_{c,p}(t) \) is the control voltage coming from the output of the feedforward loop and \( v_{c,b}(t) \) is the control voltage...
defined by the feedback loop. Consequently, the converter duty cycle will verify:

\[ d(t) = d_{ff}(t) + d_{gb}(t), \]  

(6)

where \( d_{gb}(t) \) is the component of the duty cycle determined by the feedback loop and \( d_{ff}(t) \) is the variation in the duty cycle introduced by the feedforward loop. It should be taken into account that the feedback loop will determine the value of the duty cycle for obtaining a given average output voltage defined by \( v_{ref} \) (see Fig. 1), whereas the feedforward loop will introduce a variation in the duty cycle in order to attenuate the effect of the input-voltage ripple. The variation of the parameters of LEDs (dynamic resistance, knee voltage, light-current ratio) is defined by their warming-up and, therefore, it is a slow process. Besides, the variation of the output voltage due to dimming (i.e., \( v_{ref} \)) has slight requirements regarding dynamic response. LED lamps can be then considered as very slow load. Therefore, the duty cycle defined by the feedback loop can be considered constant in each ripple period although it may change along time in a very slow way.

The value of the \( d_{gb}(t) \) can be easily obtained from (4) and (6) cancelling the effect of the input voltage ripple, as follows:

\[ d_{gb}(t) = \frac{1}{2} \sqrt{\frac{4 V_0 D_{nom} (1-D_{nom})}{V_{ref,nom}}}. \]  

(7)

The duty cycle variation that should be introduced by the feedforward loop can be obtained from (4), (6) and (7):

\[ d_{ff}(t, V_0, r, f) = \frac{1}{2} \sqrt{\frac{4 V_0 D_{nom} (1-D_{nom})}{V_{ref,nom}}}, \]  

\[ \frac{1}{2} \sqrt{\frac{1 + \frac{4 V_0 D_{nom} (1-D_{nom})}{V_{ref,nom}}}{1 + r \sin(2\pi 2f t)}} \]  

(8)

which represents \( d_{ff} \) as a function of the input voltage ripple, the output voltage and the line frequency, which are the main variables that define it.

Fig. 2 shows the optimum duty cycle variation that should be introduced by any feedforward loop (equation (8)) and also the duty cycle variation generated by the analog feedforward loop proposed in [22]. The transfer function of this loop is linear and it uses the sinusoidal input-voltage ripple as input variable (see Fig. 1). As a consequence, its output signal \( d_{ff} \) is also sinusoidal. Nevertheless, as can be seen in Fig. 2, the optimum waveform is not sinusoidal. Therefore, it is easy to see the first problem of the analog feedforward loop. Even when the gain of this loop is different for each half-period (see [22]), there is going to be a mismatch with the optimum waveform of \( d_{ff}(t) \), as shown in Fig. 2. This mismatch leads to just a partial cancelation of the ripple and, as a consequence, to some flickering (see Fig. 3).

Besides, (8) has some parameters that are not going to change (\( D_{nom}, V_{ref,nom} \)). Nevertheless, \( V_0 \) may change due to dimming, something very common in LED lighting applications, while \( r \) may change due to the PFC converter operation, depending on the demanded power or the capacitor degradation, for instance. Besides, \( f \) is defined by the grid and the corresponding tolerances. The analog feedforward loop may partially compensate variations in \( r \) and \( f \) since it uses \( V_{ref} \) as input variable (see Fig. 1). Nevertheless, its transfer function is optimized for a given value of \( V_0 \). As a consequence, its effect for different dimming states (i.e., different values of \( V_0 \)) is nothing but optimal (see Fig. 3).

Digital control is a perfect candidate in this application in order to reach the maximum ripple rejection. Due to the complex expression presented in (8), the most suitable devices for implementing the feedforward loop are FPGAs or DSPs. Nevertheless, including this kind of devices in LED drivers may be excessively expensive due to the cost limitation in many lighting applications. On the other hand, the inclusion of a \( \mu C \) in LED drivers is a more extended practice due to the new range of functionalities that are being included in them (communication under DALI standard, dimming depending on the amount of light, hour and/or presence of people, etc.). Therefore, the proposed digital feedforward loop will be specifically designed for its implementation in a pre-existing \( \mu C \). The main problems that the design will have to face are mainly two:

- The calculation of (8) in a reasonable period of time is impossible for a low-cost \( \mu C \). Therefore, look-up tables will be the most suitable option.
- Memory is limited in this kind of devices. Therefore, the number and size of the look-up tables will have to be thoughtfully analyzed.

It should be taken into account that the objective of implementing the feedforward loop in a low-cost \( \mu C \) does not prevent it from being used with FPGAs or DSPs.

In the next section, the proposed feedforward loop will be described. Nevertheless, the optimization of its design, trying to minimize the output-voltage low-frequency ripple will be presented in section IV.

III. LOW-FREQUENCY RIPPLE ATTENUATION TECHNIQUE

Fig. 4 shows the schematic of the proposed control (section VII presents a simplified block diagram of the code implemented in the \( \mu C \)). For the sake of simplicity, the feedback loop is not included in the \( \mu C \) because the duty cycle \( d_{gb} \) defined by this loop can be considered constant in each period of the ripple and, consequently, does not affect...
the feedforward loop analysis. Moreover, the feedback loop is not fast enough to be affected by the changes introduced by the feedforward loop (this is represented by the low-pass filter that senses the output voltage in Fig. 4). Therefore, the stability of the system (provided by the feedback loop) is not affected by the digital feedforward one. In any real implementation, both loops can be easily included in the µC code.

In broad outline, the µC converts both the average output voltage (set by the feedback loop) and the input-voltage ripple (sensed with a band-pass filter and a peak voltage detector) to digital format (\(V_{O,ADC}\) and \(r_{ADC}\)). Both conversions are carried out by the standard analog-to-digital converter (ADC) that most of the µCs include. The µC also senses \(v_{c_{ff}}\) and \(v_{O_{ADC}}\) to generate a feedforward signal synchronized with the converter (ADC) in order to construct the required \(v_{c_{ff}}\) waveform during a ripple period. This detection helps the µC to generate a feedforward signal synchronized with the ripple. Depending on the value of these three variables, the feedforward loop will choose one of the look-up tables. Each of these tables stores the information needed to compute \(v_{c_{ff}}\) for different grid angle intervals (i.e., for obtaining the \(v_{c_{ff}}\) waveform during a ripple period). Obviously, the values stored in each table have been pre-calculated according to given conditions \((\tau_{O_{ADC}}, \tau_{V})\).

Each look-up table stores \(N_v\) values (from \(\tau_1\) to \(\tau_{N_v}\)). These values are used in combination with the PWM module of the µC (different from the PWM module of the converter) in order to construct the required \(v_{c_{ff}}\) signal. Each value \(\tau_v\) of the look-up table defines a certain duty cycle of the µC PWM module. The output signal of this module, \(v_{c_{ff},PWM}\), is filtered and its average value is \(v_{C_{ff}}\) in the feedforward loop (see Fig. 4):

\[
v_{c_{ff}}(t) = \frac{\tau_v}{T_{PWM,\mu C}} v_{PWM,\mu C},
\]

where \(v_{PWM,\mu C}\) and \(T_{PWM,\mu C}\) are the amplitude and the period of the PWM signal generated by the µC. Obviously, the frequency of the PWM module of the µC has to be considerably higher than the ripple frequency.

From Fig. 4, it can be easily seen that:

\[
d_{ff} = \frac{v_{c_{ff}}}{V_m},
\]

where \(V_m\) is the amplitude of the sawtooth waveform of the converter PWM (different from the µC PWM module).

According to (8), \(d_{ff}\) depends on the output voltage \(V_o\), the line frequency \(f\) and the input voltage ripple \(r\). Therefore, \(v_{c_{ff}}\) also depends on these variables. As a consequence, it is mandatory to have different look-up tables for different values of \(V_o\), \(f\) and \(r\). As the number of possible look-up tables \(N_f\) is limited by the size of the µC memory, a reduced number of look-up tables can be stored in it:

\[
N_f = N_v, N_r, N_f,
\]

\(N_v\) being the number of tables calculated for different values of \(V_o\) and constant values of \(r\) and \(f\). \(N_r\) being the number of tables calculated for different values of \(r\) and constant values of \(V_o\) and \(f\). \(N_f\) being the number of tables calculated for different values of \(f\) and given values of \(r\) and \(V_o\). Actually, the variations of the line frequency, and consequently, of the ripple frequency, are limited by the corresponding regulations. Therefore, they are very small in practice and its influence in the whole analysis is very low. As a consequence and for the sake of simplicity, \(N_f\) can be considered equal to one. Therefore, (11) becomes:

\[
N_f = N_v, N_r, N_f,
\]

The memory of the µC is then organized as an array of look-up tables, as shown in Fig. 4. This array has \(N_v\) columns and \(N_r\) rows. Each column (e.g., column \(j\)) corresponds to look-up tables whose values (\(\tau_v\)) are calculated from (8), (9) and (10) for a given value of \(V_o\) (e.g., \(V_{O_{ADC}}\)). In the same way, each row (e.g., row \(i\)) corresponds to look-up tables whose values are calculated for a given value of \(r\) (e.g., \(r_{ADC}\)).

Finally, the available memory size \(M\) must verify:

\[
M = N_v, N_r, N_f, N_f.
\]

**IV. OPTIMIZED DESIGN OF THE FEEDFORWARD LOOP FOR LIGHTING APPLICATIONS**

In this section, the optimization of the digital feedforward loop will be presented for the particular case of lighting applications. This optimization is focused on reducing only the ripple that produces harmful flickering, moving it to a frequency range that does not affect human’s health.

It should be taken into account that the values of \(V_o\) and \(r\) have to be sensed by the µC. This means that two analog-to-digital conversions have to be carried out (as has been said, the resulting values of the ADC that correspond to \(V_o\) and \(r\) are \(V_{O_{ADC}}\) and \(r_{ADC}\), as presented in Fig. 4). Due to the slow variations of these variables, each conversion can be carried out once in each period of the ripple (half the line period) and the resulting values can be used in the calculation and construction of the whole \(d_{ff}\) waveform that is going to be used in the next ripple period.

Every conversion involves an error due to the limited
resolution of the ADC (i.e., quantization noise). As will be explained, this error can be neglected in this particular case due to its small magnitude in comparison to other sources of error derived from the simplified design of the proposed feedforward loop.

The first important error is due to the fact that the number of look-up tables that can be stored in the µC memory is limited to \( N_f N_V \). That means that only \( N_f \) values of input-voltage ripple and \( N_V \) values of output voltage can be discriminated. The mathematical representation of this error can be expressed as (see Fig. 5):

\[
V_{µC} = \frac{V_{O\_max}}{2N_V} + \frac{V_{O\_max}}{2N_V} \cdot \text{round}_{\_low} \left( \frac{V_{O\_ADC}}{V_{O\_max}} \cdot N_V \right),
\]

\[
r_{µC} = \frac{r_{max}}{2N_f} + \frac{r_{max}}{N_f} \cdot \text{round}_{\_low} \left( \frac{r_{ADC}}{r_{max}} \cdot N_f \right),
\]

where \( V_{µC} \) and \( r_{µC} \) are the values discriminated by the µC according to the look-up tables defined in the system and to the digital variables \( V_{O\_ADC} \) and \( r_{ADC} \). Round_{low} represents a function that provides the greatest integer lower than the argument. The maximum error in each variable is (see \( P_1 \) in Fig. 5):

\[
e_{V_{µC}} = \frac{V_{O\_max}}{2N_V},
\]

\[
e_{r_{µC}} = \frac{r_{max}}{2N_f}.
\]

Considering realistic values of \( M \) and taking into account (13), the quantization noise due to the ADC can be disregarded and only the error due to the limited value of \( N_f \) and \( N_V \) \( (e_{V_{µC}} \text{ and } e_{r_{µC}}) \) is going to be considered.

In order to take into account this first source of error, \( V_{O\_nom} \) and \( r \) in (8) must be replaced with \( V_{µC} \) and \( r_{µC} \):

\[
d_{ff}(t, V_{µC}, r_{µC}) = \sqrt{1 - \frac{4V_{µC}D_{nom}(1-D_{nom})}{V_{O\_nom}}} - \sqrt{1 - \frac{4V_{µC}D_{nom}(1-D_{nom})}{[1 + r_{µC} \sin(2\pi f_{T_{ADC}})]V_{O\_nom}}}.
\]

The value of \( d_{ff}(t, V_{O\_f}) \) according to (8) (i.e., the desired evolution of \( d_{ff} \) to ideally cancel the ripple) and the value of \( d_{ff}(t, V_{µC}, r_{µC}) \) according to (18) (i.e., the evolution of \( d_{ff} \) obtained so far with the proposed method) have been represented in Fig. 6 (\( d_{ff}(t_{µC}, V_{µC}, r_{µC}) \) will be explained later). This figure shows the discrepancies between \( d_{ff}(t, V_{O\_f}) \) and \( d_{ff}(t, V_{µC}, r_{µC}) \) due to the finite number of look-up tables. In this example, the amplitude of \( d_{ff}(t, V_{O\_f}) \) is higher than the amplitude of \( d_{ff}(t, V_{µC}, r_{µC}) \) because \( r_{µC} \) is lower than \( r \) (see \( P_2 \) in Fig. 5).

As has been already explained, the µC will choose one of the look-up tables according to the value of \( V_{µC} \) and \( r_{µC} \) in order to build the signal \( d_{ff} \) during the grid period. The way the signal is built has also been explained and is based on the use of the PWM module of the µC. Each value \( c_{ff} \) of the look-up table will define the value of \( v_{c\_ff} \) during a certain period of time \( T_{µC} \). It can be easily seen that \( T_{µC}=1/(2\cdot f_{N_f}) \).

As a consequence, \( T_{µC} \) is made of a series of echelons or steps (see \( d_{ff}(t_{µC}, V_{µC}, r_{µC}) \) in Fig. 6) whose length depends on the size of the look-up table (\( N_f \)). A discrete time variable \( r_{µC} \) can be defined according to this explanation:

\[
r_{µC} = nT_{µC},
\]

where \( n \) changes from 1 to \( N_f \). From (18) and (19):

\[
d_{ff}(t_{µC}, V_{µC}, r_{µC}) = \sqrt{1 - \frac{4V_{µC}D_{nom}(1-D_{nom})}{V_{O\_nom}}} - \sqrt{1 - \frac{4V_{µC}D_{nom}(1-D_{nom})}{[1 + r_{µC} \sin(2\pi f_{T_{µC}})]V_{O\_nom}}}.
\]

Obviously, the higher \( N_f \), the shorter each echelon and the higher the accuracy. Besides, the error due to the limited number of look-up tables is already included in (14) and (15) and, therefore, in (20). Hence, it mathematically describes the two sources of error that have to be taken into account (see Fig. 6).

From (3), (6), (7) and (20), the effect of the digital feedforward loop in the output-voltage ripple can be expressed as:

\[
v_{\text{c}}(t, t_{µC}, V_{µC}, r_{µC}) = v_{\text{c}_{\_nom}}[1 + r \sin(2\pi f_{T_{µC}})](1 + n_f).
\]

\[
\left[ d_{ff}(t, V_{µC}, r_{µC}) \right] \left[ 1 - d_{ff}(t, V_{µC}, r_{µC}) \right] = \left[ d_{ff}(t, V_{µC}, r_{µC}) \right]
\]

Fig. 7 shows the output voltage when no feedforward loop is implemented, when only the first source of error is considered (\( N_f=∞ \)) and when both sources of error are taken into account (\( N_f=12 \)). As can be seen, the second and the third waveforms are equal in average value, the difference are the echelons that the third one presents.

Before calculating the optimum number of \( N_f, N_V \) and \( N_N \) for a given value of \( M \), it would be interesting to compare the proposed digital control and the analog control proposed in [22]. In Fig. 8a, the resulting output voltage for both methods is shown when dimming is applied and the output voltage is reduced to 80% of its nominal value. As can be seen, the ripple in the case of using the digital feedforward loop is due to the two aforementioned reasons.
Nevertheless, this ripple is considerably lower than the ripple in the case of using the analog one. In this last case, the ripple is due to the difference between the actual average value of $V_o$ used when tuning the loop and the actual average value of $V_o$. Nevertheless, when the value of $V_o$ is the nominal one, results are not so easy to analyze. As can be seen in Fig. 8b, when the value of $V_o$ is the one used in the design of the analog feedforward loop ($V_{o, nom}$), this loop seems to offer better results, mainly due to the lack of echelons in its $d_o$ signal. Therefore, at first sight, the analog feedforward loop is better in nominal conditions. Nevertheless, dimming is a very common feature in any LED driver. Besides, it should be taken into account that the purpose of any feedforward loop in this kind of applications is not exactly reducing the output voltage/current ripple, but to reduce the light flickering that may affect humans. Hence, it is acceptable to have high-frequency ripple (i.e., high-frequency flickering) in the emitted light as long as it does not affect human’s health. In fact, this is the idea beneath the widely-used PWM-mode dimming used in LED lighting [3]. Therefore, the purpose of the proposed digital feedforward loop is not providing a constant output voltage/current to the LED lamp, but to translate all the flickering to a frequency range that does not affect people. With this straightforward approach to the flickering problem, it is possible to simplify the digital feedforward loop. As a consequence, this loop can be implemented in a small-size µC while obtaining better results than with the analog one for the whole duty cycle range (i.e., dimming range). This idea is valid not only for the AHBC proposed in this paper, but also for any other converter used in LED lighting applications and with a non-linear transfer function.

As has been said, only the harmonics that may affect human’s health should be taken into account (i.e., relevant flickering). Therefore, only the harmonics below a limit frequency $f_{lim}$ will be considered in the ripple/flickering analysis. Nevertheless, human eye does not behave as an ideal low-pass filter with an infinite slope when the cut-off frequency (i.e., $f_{lim}$) is reached. Human eye is a complex system and, consequently, an analysis in which different weights are given to different harmonic components should be carried out. This special treatment to flickering in LEDs is the result of the fast response in converting electrical energy into light. As presented in [32] and [33], the use of LEDs in lighting makes the redefinition of flickering analysis something mandatory. Unfortunately, although the first steps have been taken in this direction, there is not a precise regulation regarding flickering in LED lighting yet. Besides, the purpose of this paper is not providing a value for these weights, but to compare the results obtained with the analog loop and the results obtained with the proposed digital one. Therefore, to ease this comparison, all the weights will be equal to one for frequencies lower than $f_{lim}$. Although this is considering the human eye as an ideal low-pass filter, it is also the worst situation for the digital loop analysis.

The situation depicted in Fig. 8 turns into the one shown in Fig. 9 when only the harmonic components lower than $f_{lim}$ are taken into account. As can be seen, now the results obtained with the digital feedforward loop are considerably better than with the analog one, even when dimming is not applied. The reason is that with the digital loop all the strong harmonic components are placed in a frequency range with no associate flickering problems. Nevertheless, this can only be achieved when the variables $N_e$, $V_o$ and $N_i$ are wisely chosen.

Once the digital loop has proven to be more effective
than the analog one, the proposed design guideline of the
digital feedforward loop will be presented by means of an
example. In this example, the AHBC has an input voltage of
385 V with a maximum ripple of 20% (peak to peak), an
output voltage of 20 V and a nominal power of 40 W. The
turn ratios of the transformer are 0.177 and 0.07 (n1 and n2
respectively). The nominal duty cycle is 33% and the
memory size available for storing the look-up tables is
1kword. The frequency limit, f_{limit} is set at 400 Hz. This is a
common frequency for PWM dimming because it is
assumed to be far from the frequency range that implies
harmful flickering.

The final purpose of the design guideline is optimizing
the value of the three variables involved in the definition of
\( d_{ff} (N_v, N_c, N_t) \) for a given value of \( M \). For doing that, (13),
(14), (15), (20) and (21) must be taken into account. The
design should be based on the analysis and reduction of the
relevant output-voltage ripple (i.e., only the harmonics
below \( f_{limit} \)) for the worst possible situation. That means
considering always the greatest possible errors in \( V_{dc} \) and
\( f_{cfc} \) when selecting the most suitable look-up table (see Fig.
5 and equations (15) and (16)).

The optimum value of \( N_t \) can be obtained considering
that it has to assure that the first strong harmonic component
of the resulting output-voltage ripple has a frequency higher
than \( f_{limit} \). Besides, as can be seen in Fig. 6, when the zero-
crossing of the input voltage ripple (positive or negative) is
centered in one of the steps of \( d_{ff} \), the value that should be
introduced by the feedforward loop is zero. By making the
first and last steps half the length of the rest steps (see Fig.
6), the corresponding zero-crossing will be always centered
in the step that both half-steps conform. As a consequence,
this value of the look-up table is always zero and that implies
that, actually, there is no need of storing it. This reduces the size of all
the look-up tables stored in the available memory, something very important due to its
small size.

Due to the proposed digitalization and implementation of
\( V_{c_{ff}}, \) the harmonic analysis of \( v_o \) shows that the frequency
(corresponding to its first strong harmonic component
\( f_{strong,1} \)) satisfies:

\[
f_{strong,1} = (N_t - 1) \cdot 2f
\]  

(22)

For instance, when \( N_t = 6 \) and \( f = 50 \) Hz, then \( f_{strong,1} \) is 500
Hz and, consequently, higher than \( f_{limit} = 400 \) Hz.

From (22) and taking into account that \( N_t \) is an integer
value, the optimum value of \( N_t \) that guarantees that
\( f_{strong,1} > f_{limit} \) is:

\[
N_{t, opt} = \frac{f_{limit} + 2f}{2f} + 1
\]  

(23)

The relative value of the output voltage ripple (peak-to-
peak value) when only the relevant harmonics are
considered can be obtained from (21). This value has been
labeled as \( f_{v_{o, relevant}} \) (different from \( r \), which represents the
input voltage ripple) and it has been represented in Fig. 10
as a function of \( k_{\phi} \) (defined as \( k_{\phi} = N_v / N_t \)) for the
aforementioned worst situation. As can be seen, this figure
validates the previous optimization of \( N_v \), as the minimum value of \( r_{v_{o, relevant}} \) is obtained when \( N_t \) satisfies (23) (i.e., in
the proposed example, \( N_t = 6 \) because \( f_{limit} = 400 \) Hz). Besides,
this figure shows that the optimum value of \( k_{\phi} \) is around 4
for this particular example. It should be taken into account
that this optimum ratio varies depending on the value of the
variables that define the equations used in the design
process. For instance, a reduction in the nominal input-

Fig. 10 also clearly shows that the influence of \( N_v \) is
considerably higher than the influence of \( k_{\phi} \). This will be
depicted in section V.

To sum up, this example shows the way the proposed
equations should be used for optimizing the digital
feedforward loop of any design. The resulting figures will
allow us to decide which the most suitable values of \( N_v, \)
and \( k_{\phi} \) are. Once they are determined, \( N_{v, opt} \) and \( N_{t, opt} \)
can be easily calculated with (13).

V. EXPERIMENTAL RESULTS

A prototype has been built in order to verify the
mathematical model (see Fig. 11). The nominal input and
output voltage, as well as the nominal power are \( V_{dc} = 385 \) V,
\( V_o = 20 \) V and \( P = 40 \) W. The transformer is built with an
ETD34 core and the output inductor with an E30 core. Their
inductances are 3.1 mH (magnetizing inductance) and 80
\( \mu \)H, respectively. The turns ratios of the transformer are
0.177 and 0.07. The output capacitor is implemented with 6
MKP capacitors of 3.7 \( \mu \)F each. The input capacitors are
270-nF MKP ones. The digital feedforward loop is
implemented in a PIC12F683 \( \mu \)C from Microchip. It has 8
pins and 2 kwords of program memory (Harvard structure).
One kword is left apart for housekeeping code and the
feedforward code (PWM, ADC, etc.) and the other 1
kword is used for storing the look-up tables.

In Fig. 12, the influence of \( N_t \) is analyzed. Therefore, for
the sake of clarity \( N_v \) and \( N_t \) are forced to be equal. The
three figures include the results obtained with the
mathematical model (both, considering all the harmonic
components and only those located at a frequency lower
than \( f_{limit} \)). As can be seen, the predicted results
(mathematical model) are close to the real ones. The only
significant differences occur when the signal \( d_{ff} \) generated

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{fig10.png}
\caption{Fig. 10. Peak to peak value of the relevant ripple for different values of \( k_{\phi} \) when the available memory is 1 kword.}
\end{figure}
by the feedforward loop changes from one value to the next one. The mathematical model does not take into account the dynamic of the output filter and, as a consequence, the variations of the output voltage in the model are a little bit faster than in the prototype. Nevertheless, this difference is very small. Therefore, the mathematical model is validated and can be used for optimizing the digital feedforward loop as has been explained in previous sections. As can be seen in Fig. 12c, when the number of different values \( N \_\text{lmt} \) used in the construction of \( d_{\text{ff}} \) is lower than \( f_{\text{num}}/2 + 2 \) (\( N = 5 \) in this case), at least one strong harmonic component is lower than (or equal to) \( f_{\text{num}} \) and, consequently, the amplitude of the relevant ripple is higher than in Fig. 12b or Fig. 12a. On the other hand, the amplitude of the relevant ripple in Fig. 12a is slightly higher than in Fig. 12b. In this case, the reason can be found in the lower value of \( N \_v \) and \( N \_l \) (both equal to 12), which makes the error when sensing the input variables greater. Besides, the advantage of a higher value of \( N \_v \) (\( N = 7 \)) when \( N \_v = N \_l = 12 \) is not a real advantage as the relevant harmonic components are already higher than \( f_{\text{num}} \) in Fig. 12b, when \( N \_v \) is equal to 6 (and \( N \_l = N \_v = 13 \)). Nevertheless, it should be mentioned that this difference is very small. Therefore, a very important issue when optimizing the feedforward loop is always taking into account (23) so that the strong harmonic components are always higher than \( f_{\text{num}} \) and, consequently, outside the harmful-flickering range.

In Fig. 13, the influence of \( N \_v \) and \( N \_l \) is studied. Consequently, the value of \( N \_v \) is the optimum one. In Fig. 13a, the values of \( N \_v \) and \( N \_l \) are the optimum ones while in Fig. 13b they are not. The relevance of optimizing \( N \_v \) and \( N \_l \) can be seen when comparing both figures. In the latter, in which the values of \( N \_v \) and \( N \_l \) are not the optimum ones, the ripple is 50% higher than when these two variables are optimized.

In Fig. 14, the output voltage ripple is shown when dimming is applied and the optimum values of \( N \_v \), \( N \_l \), and \( N \_l \) are chosen (i.e., \( N \_v = 28 \), \( N \_l = 6 \) and \( N \_l = 6 \)). As can be seen, the proposed digital feedforward loop reduces the ripple under any dimming condition as it adapts its output signal \( (d_{\text{ff}}) \) to the operating conditions. Moreover, the effect of the non-linear transfer function has no effect as the calculation of the values stored in the look-up tables of the digital feedforward loop always takes this into account. This represents the main advantage of the proposed loop in comparison to the standard analog one (even with different gains for each half period of the ripple, as explained in [22]).

Finally, Fig. 15 presents three different output voltages for three different input-voltage ripples. The output voltage is shown including the dc value in order to clearly verify that the proposed digital feedforward loop effectively reduces the output-voltage ripple. As can be seen, the output voltage is nearly constant, even considering all harmonic components and not only those located at a frequency lower than \( f_{\text{num}} \).

VI. CONCLUSIONS

The AHBC has proven to be a promising option as second stage of any two-stage ac-dc LED driver (in comparison to other topologies, such as the Flyback converter). Its only drawback is its poor dynamic response due to the resonance between the magnetic inductance of the transformer and the input capacitors. This is a problem when the feedback loop has to compensate the effect of the low-frequency ripple that appears in its input voltage. The solution proposed in this paper is the implementation of a digital feedforward loop in a small-size \( \mu \)C. Due to the limited size of the available memory and the limitations in the mathematical operations that the \( \mu \)C can carry out, the proposed feedforward loop is based on cancelling not all the ripple, but only the one that may lead to harmful flickering. With this specific design the feedforward loop can be strongly simplified, although it has to focus on issues different from the traditional ones (e.g., the quantization noise can be disregarded). The proposed feedforward loop is based on look-up tables. The first key point in the design is calculating the number of different values stored in these tables. This number should assure that the resulting harmonic components are not going to generate flickering in a frequency range that may affect humans (e.g., when the
harmful flickering is limited to frequencies lower than 400 Hz, the optimum number is six). The digital loop takes, as input variables, the input voltage ripple and the average output voltage. The second key point is then defining the number of different values that the loop can discriminate in these two variables. Experimental results show that the proposed digital feedforward loop presents a lower ripple than the analog one. This improvement is more remarkable when dimming is applied; situation in which the digital feedforward loop still reduces the ripple nearly to zero while the analog one, rather than reducing it, increases it.

VII. ANNEX

A simplified block diagram of the code implemented in the µC is shown in Fig. 16. As can be seen, it corresponds to the explanation given in section III.

After the configuration of all the pins and peripherals that will be used, the code of the digital feedforward loop waits for a zero-crossing in the input-voltage ripple to start. At the beginning of each ripple period, the LUT that will be used during this period is selected according to the values of i and j. These values are defined according to the results obtained in the previous ripple period from the AD conversion of the input voltage ripple and output voltage. Also, the variables x and Count are set to zero.

The selected LUT, along with variable x (actually, x is the index of the LUT), is used to obtain the value of $\tau_x$ (a full description of this concept is provided in section III). This value is used to define the duty cycle of the PWM module of the µC, which will define the feedforward-loop signal once it is filtered (see Fig. 4b).

Every time a period of the µC PWM module is finished (different from the PWM module of the converter), variable Count is incremented and compared to Count_limit. This allows the system to identify when the duty cycle of the µC PWM module has to be updated with a new $\tau_x$ (according to x, which is also incremented); in other words, when the signal of the feedforward loop has to change.

When the input voltage ripple reaches its peak value, the system launches an AD conversion of this parameter. It should be taken into account that a peak detector is implemented in the hardware, so the conversion does not need to be precisely launched when the peak of the ripple takes place. As can be seen, the ADC interrupt allows the system to update the value of i and j, which will be used at the beginning of the next ripple period in order to select the LUT that will be used.

Finally, it should be said that the system also detects the end of a ripple period and restarts the generation of the feedforward signal.

REFERENCES

Choose LUT according to task

Launch ADC (output voltage)

Update i value

Update j value

Launch ADC (input voltage ripple)

Update i value

Update j value

End Interrupt (ADC)

ADC of output voltage on input ripple?

PWM module with ripple

µC

PWM module period finished?

Count

Count=0

Yes

No

Yes

No

Counter limit reached?

No

Yes

No

Ripple period finished?

x

x+*

Counter=0

Fig. 16. Simplified block diagram of the feedforward loop code

Applied Power Electronics Conference and Exposition (APEC), Fort Worth, 2011.


