# Digital control of switching mode power supplies

**Simone Buso**

**University of Padova – ITALY**

**Dept. of Information Engineering – DEI**

# About the instructor

## Simone Buso

**Associate Professor of Electronics**

**University of Padova**
**Dept. of Information Engineering – DEI**
**Via G. Gradenigo, 6/a**
**35131 Padova**
**ITALY**

**phone: +39 049 8277525**
**fax: +39 049 8277699**

**E-mail: simone.buso@dei.unipd.it**

# Lesson 1

## Digital current mode control of power converters

Control of power converters by PWM modulation
Analog PWM: naturally sampled implementation
Digital PWM: uniformly sampled implementation
Single update and double update modes
Minimization of the modulator delay

## Space Vector Modulation

Modulation in three phase systems
$\alpha\beta$ Transformation
Space Vector Modulation

## Basic motivations

Digital control offers the possibility to implement sophisticated control laws, taking care of system non linearities, parameter variations or construction tolerances by means of self-analysis and auto tuning strategies, very difficult or impossible to implement analogically.

Software based digital controllers are inherently flexible, which allows the designer to modify the control strategy, or even to totally re-program it, without the need for significant hardware modifications. Also very important are the higher tolerance to signal noise and the complete absence of ageing effects or thermal drifts.

A large variety of electronic devices, from home appliances to industrial instrumentation, require the presence of some form of man to machine interface (MMI). Its implementation is almost impossible without having some kind of embedded microprocessor. The utilization of the computational power, that thus becomes available, also for lower level control tasks is often very convenient.

## Basic motivations

The application of digital controllers has been increasingly spreading and has become the only effective solution for a lot of industrial power supply production areas. Adjustable speed drives (ASDs) and uninterruptible power supplies (UPSs) are nowadays fully controlled by digital means.

The increasing availability of low cost, high performance microcontrollers and digital signal processors stimulates the diffusion of digital controllers in areas where the cost of the control circuitry is a critical issue, e.g. in power supplies for portable equipment, battery chargers, electronic welders ...

A significant increase of digital control applications in these very competing markets is not likely to take place until new implementation methods, different from the traditional microcontroller or DSP unit application, prove their viability. From this standpoint, the research efforts need to be focused on the design of custom integrated circuits, more than on algorithm design and implementation. Issues like occupied area minimization, scalability, power consumption minimization, limit cycle containment play a key role in this context.

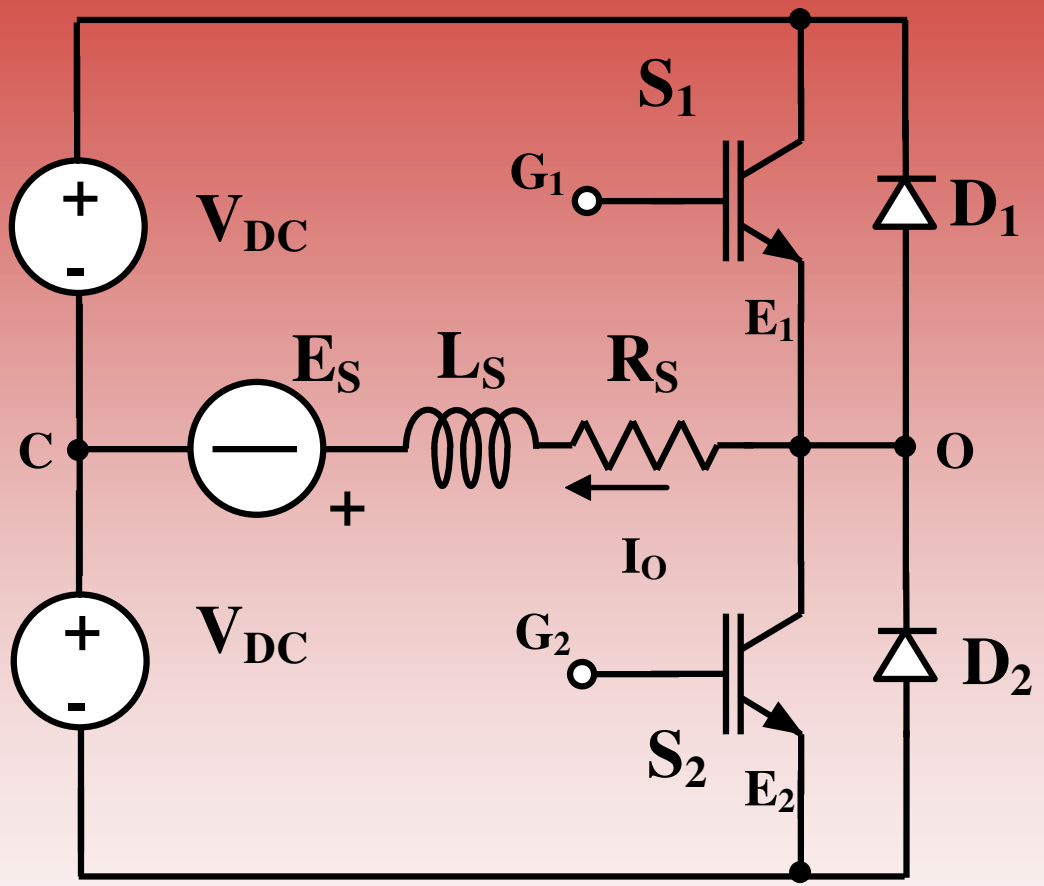
## Case study

Several different circuit topologies and related digital controllers could be considered: what we are going to do, instead, is to consider just a single, simple application example, i.e. the half bridge voltage source inverter.

The principles of its more commonly adopted low level control strategy, namely Pulse Width Modulation (PWM), will be explained, at first in the continuous time domain, successively in the discrete time domain.

The issues related with PWM control modelling are fundamental for the correct formulation of a Switch Mode Power Supply (SMPS) digital, or even analog, control problem.

## Case study: a Voltage Source Inverter




Half bridge voltage source inverter

## Case study: a Voltage Source Inverter

The VSI represented can be described in the state space by the following equations:

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx + Du \end{cases}$$

where $x = [I_O]$ is the state vector, $u = [V_{OC}, E_S]^T$ is the input vector and $y = [I_O]$ is the output variable.

Direct circuit inspection yields:

$A = [-R_S/L_S]$, $B = [1/L_S, -1/L_S]$, $C = [1]$, $D = [0, 0]$

Based on this model and using Laplace transformation, the transfer function between the inverter voltage $V_{OC}$ and the output current $I_O$, $G_{I_O V_{OC}}$ can be found to be:

$$G_{I_O V_{OC}}(s) = \frac{\tilde{I}_O}{\tilde{V}_{OC}} = C \cdot (sI - A)^{-1} \cdot B_{11} = \frac{1}{R_S} \cdot \frac{1}{1 + s\frac{L_S}{R_S}}$$

## Case study

The VSI controller is organized hierarchically. In the lowest level a controller determines the state of each of the two switches, and in doing this, the average load voltage. This level is called the modulator level.

The strategy according to which the state of the switches is changed along time is called the modulation law. The input to the modulator is the set point for the load average voltage, normally provided by a higher level control loop.

A direct control of the average load voltage is possible: in this case the VSI is said to operate in open loop conditions. However, this is not a commonly adopted mode of operation, since no control of load current is provided in the presence of load parameter variations.

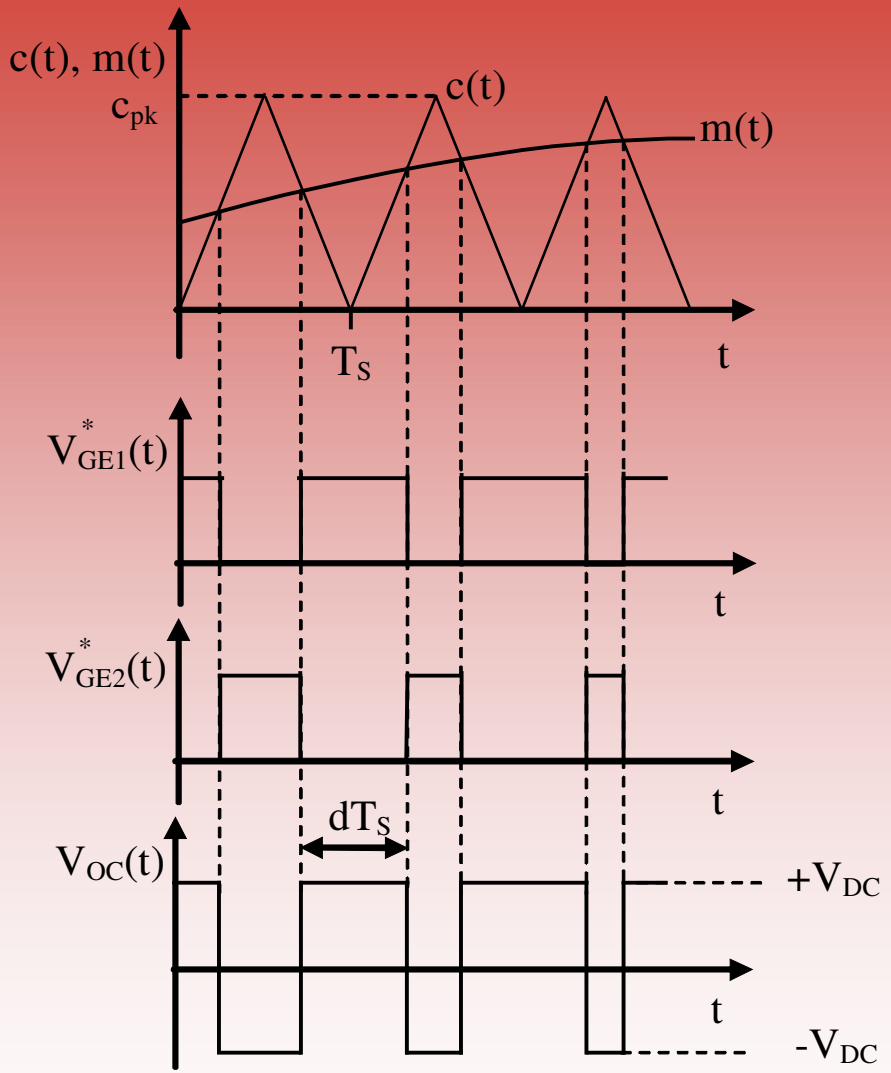
## Case study

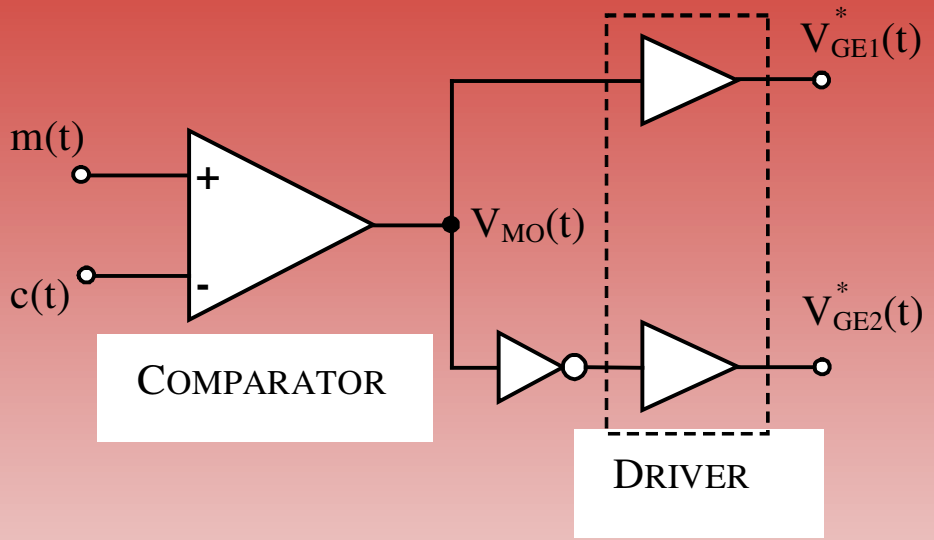
Because of that, in the large majority of cases, a current controller can be found immediately above the modulator level. This is responsible for providing the set-point to the modulator.

Similarly, the current controller set-point can be provided by a further external control loop or directly by the user.

In the latter case, the VSI is said to operate in current mode, meaning that the control circuit has turned a voltage source topology into a controlled current source.

# PWM modulator: analog implementation




Naturally sampled implementation of a PWM modulator.

## PWM modulator: principles of operation

A square wave voltage $V_{OC}$ is applied to the load, with constant frequency $f_S = 1/T_S$, $T_S$ being the period of the carrier signal $c(t)$, and variable duty cycle $d$. This is implicitly defined as the ratio between the time duration of the $+V_{DC}$ voltage application period and the duration of the whole modulation period, $T_S$.

We can explicitly relate signal $m(t)$ to the resulting PWM duty-cycle. Simple calculations show that, in each modulation period, where a constant $m$ is assumed, the following equation holds:

$$\frac{m}{dT_S} = \frac{c_{pk}}{T_S} \quad \Leftrightarrow \quad d = \frac{m}{c_{pk}}$$

In addition, we can compute the relationship between the duty-cycle and the average inverter voltage. This turns out to be:

$$\overline{V}_{OC}(t) = \frac{1}{T_S}\int_{t-T_S}^{t} V_{OC}(\tau)d\tau = \frac{1}{T_S}\left(T_S \cdot V_{DC} \cdot d(t) - V_{DC}\left(1-d(t)\right)\cdot T_S\right) = V_{DC}\left(2d(t)-1\right)$$

## PWM modulator: principles of operation

If we assume that the modulating signal changes slowly along time, with respect to the carrier signal, i.e. the upper limit of *m(t)* bandwidth is well below $1/T_S$, we can still consider the above result correct.

This means that, in the hypothesis of a limited bandwidth *m(t)*, the information carried by this signal is transferred, by the PWM process, to the duty-cycle, that will change slowly along time following the *m(t)* evolution. Based on the previous relation, this means that

$$\frac{\partial d}{\partial m} = \frac{1}{c_{pk}}$$

The duty-cycle, in turn, is transferred to the load voltage waveform by the power converter. The slow variations of the load voltage average value will therefore copy those of signal *m(t).* Therefore, the modulator transfer function, including the inverter gain will be given by:

$$\frac{\partial \overline{V}_{OC}}{\partial d}\frac{\partial d}{\partial m} = \frac{2V_{DC}}{c_{pk}}$$

## PWM modulator: principles of operation

Combining the above results with the previously calculated transfer function between inverter voltage and inductor current $G_{I_O V_{OC}}$, we can now find the modulating signal to inductor current transfer function G(s), that will be used in the design of the current loop compensator.
This is given by:

$$G(s) = \frac{\tilde{d}}{\tilde{m}} \frac{\tilde{V}_{OC}}{\tilde{d}} \frac{\tilde{I}_O}{\tilde{V}_{OC}}(s) = \frac{2V_{DC}}{c_{pk}} \frac{1}{R_S} \cdot \frac{1}{1 + s\frac{L_S}{R_S}}$$

and represents the dynamic relationship between small perturbations of the modulating signal (around its steady state value) and the corresponding variations of the average inductor current value.

## PWM modulator: principles of operation

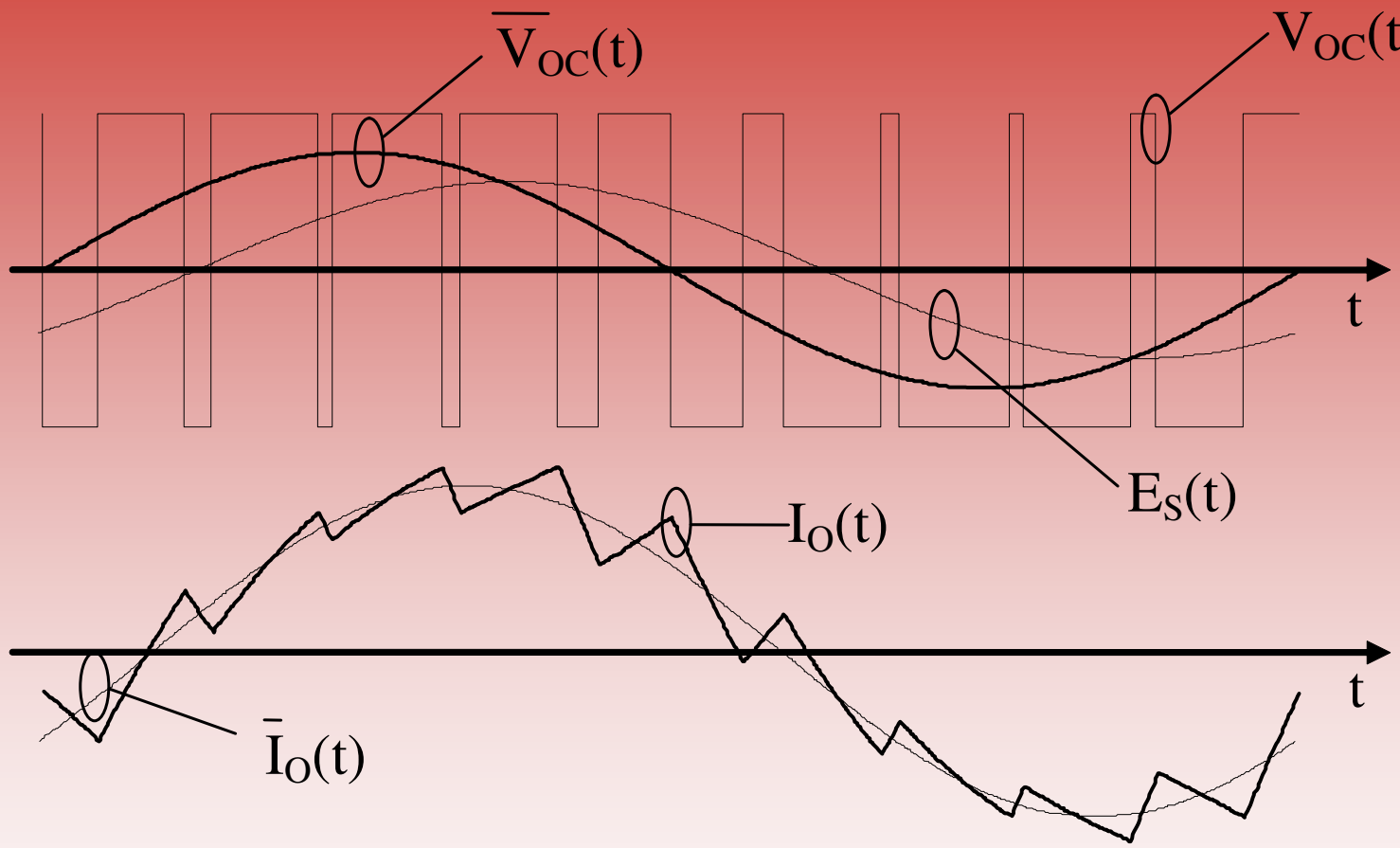
A more mathematically sound approach, would basically show that the frequency content, i.e. the spectrum, of the modulating signal *m(t)* is shifted along frequency by the PWM process, and is replicated around all integer multiples of the carrier frequency.

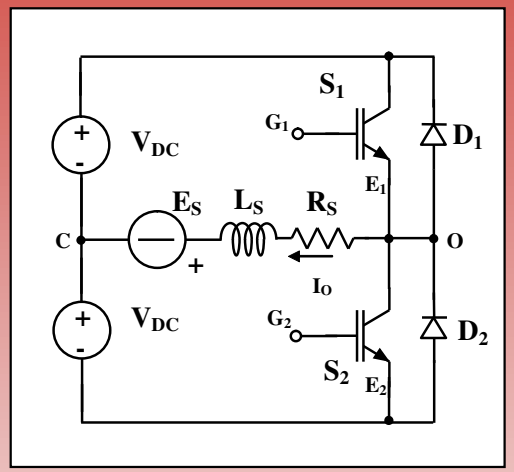
This implies that, as long as the spectrum of signal *m(t)* has a limited bandwidth with a upper limit well below the carrier frequency, signal demodulation, i.e. the reconstruction of signal *m(t)* spectrum from the signal $V_{OC}(t)$, with associated power amplification, can be easily achieved by low pass filtering $V_{OC}(t)$.

In the case of power converters, like the one we are considering here, the low pass filter is actually represented by the load itself.

Again, this implies that the previously found transfer function is, in a first approximation (i.e. neglecting the residual ripple), correct. Please note that, from now on, the modulating signal *m(t)* will always be assumed to be limited in bandwidth.

# PWM modulator: principles of operation




Example of PWM operation

## PWM modulator: dynamic response

The previous analysis assumes the following relationship exists between small variations of the duty-cycle, $\tilde{d}$, and the corresponding variations of the modulating signal, $\tilde{m}$.

$$\frac{\partial d}{\partial m} = \frac{1}{c_{pk}}$$

The purely proportional relationship implies an instantaneous response (i.e. exhibiting no delay whatsoever) of the modulator to changes in the modulating signal. A fundamental question arises:

is the assumption **correct?**

The answer to this question has been found 30 years ago by R.D. Middlebrook, and it is absolutely **affirmative.**

## PWM modulator: dynamic response

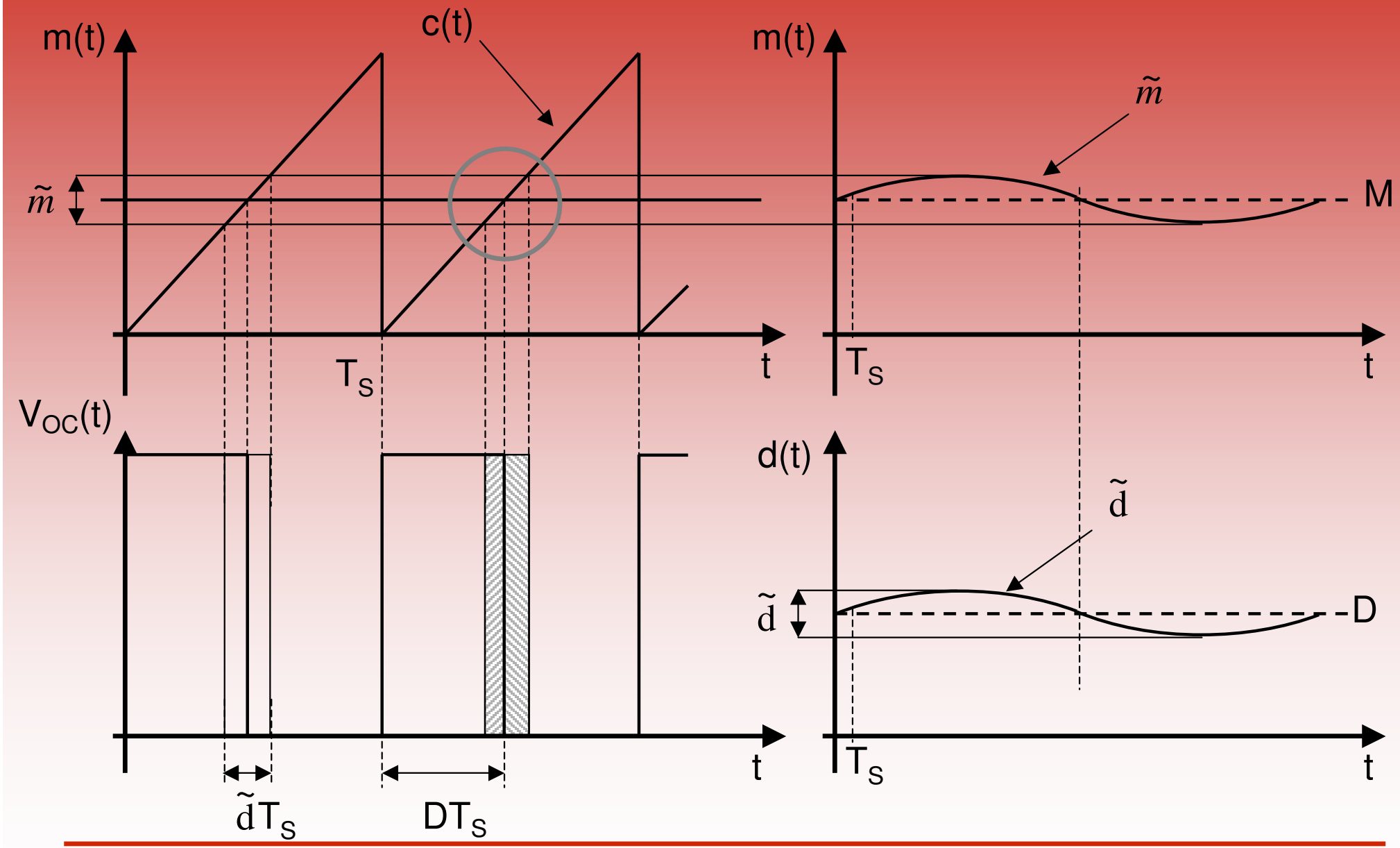
Indeed, it is possible to see that any change in the modulating signal's amplitude, provided that its *bandwidth limitation* is maintained, implies an "immediate", i.e. in phase, adjustment of the resulting duty-cycle.

This means that the analog implementation of PWM guarantees the minimum delay between modulating signal and duty-cycle. Therefore, the intuitive representation of the modulator operation can be actually corroborated by a more formal, mathematical analysis.

The formal derivation of an equivalent modulator transfer function, in magnitude and phase, has been studied and obtained since the early 80's. The modulator transfer function has been determined using small signal approximations [1], where the modulating signal *m(t)* is decomposed in a *dc* component *M* and a small signal perturbation $\tilde{m}$ (i.e. $m(t) = M + \tilde{m}$). The corresponding duty-cycle has been found, whose small signal component is called $\tilde{d}$.

[1] R.D. Middlebrook; "Predicting modulator phase lag in PWM converter feedback loops", Advances in switched-mode power conversion, vol I, pp. 245-250, 1981.

## PWM modulator: dynamic response



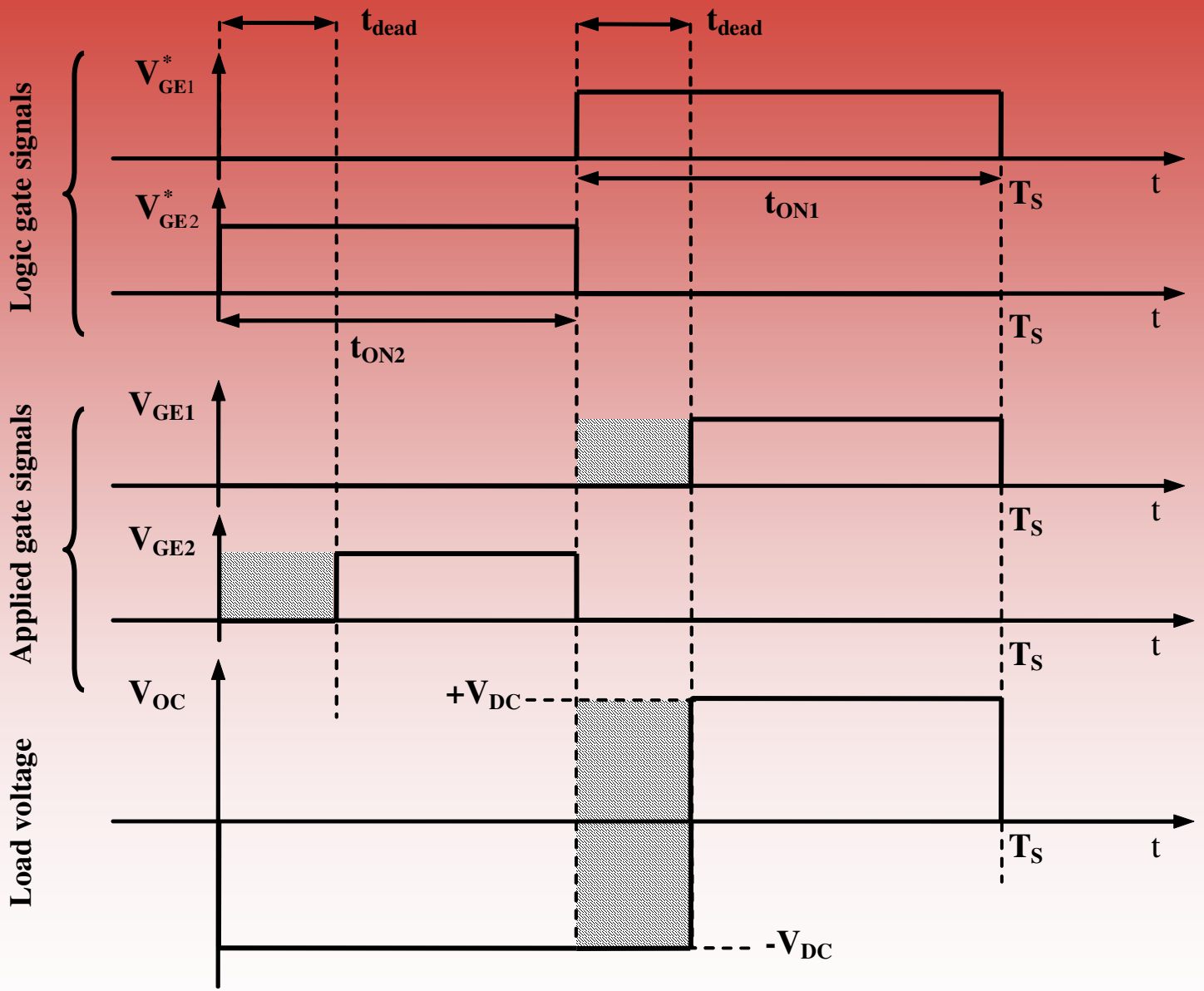
## PWM modulator: dynamic response

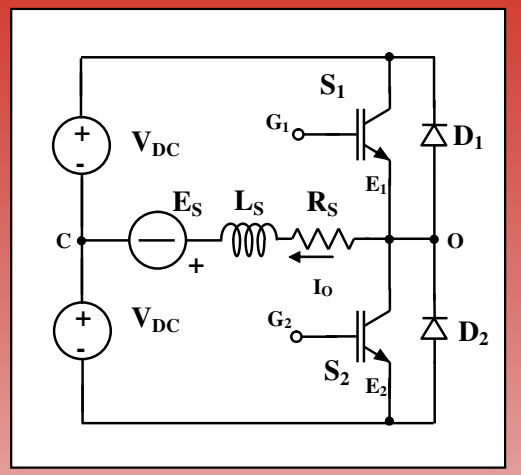
Under these assumptions, in [1], the author demonstrates that the phase lag of the naturally sampled modulator is actually zero, i.e. $\tilde{m}$ and $\tilde{d}$ are in phase, concluding that the analog PWM modulator delay can always be considered negligible. Therefore, the transfer function we already computed can be considered as well a reasonable model of the inverter dynamic behaviour.

Quite differently, we will see in the following how the discrete time or digital implementations of the pulse width modulator, that necessarily imply the introduction of sample-and-hold effects, often determine a significant response delay [2].

[2] D.M. Van de Sype, K. De Gusseme, A.P. Van den Bossche, J.A. Melkebeek, “Small-signal Laplace-domain analysis of uniformly-sampled pulse-width modulators”; 2004 Power Electronics Specialists Conference (PESC), 20-25 June, pp. 4292 - 4298

# PWM modulator: dead times




Dead times effect for $I_o > 0$

## PWM modulator: dead times

To avoid cross conduction, the modulator delays $S_1$ turn-on by a time $t_{dead}$, applying the $V_{GE1}$ and $V_{GE2}$ command signals to the switches. The duration of $t_{dead}$ is long enough to allow the safe turn off of switch $S_2$ before switch $S_1$ is commanded to turn on, considering propagation delays through the driving circuitry, inherent switch turn off delays and suitable safety margins.

The typically required dead time duration for 600 V, 40 A IGBT is currently well below 1 µs. Of course, the dead time required duration is a direct function of the switch power rating.

It is important to notice that the effect of the dead time application is the creation of a time interval where both switches are in the off state and the load current flows through the free-wheeling diodes.

Because of that, an undesired difference is created between the duration of the $S_1$ switch on-time and the actual one, that turns into an error in the voltage applied to the load.

## PWM modulator: dead times

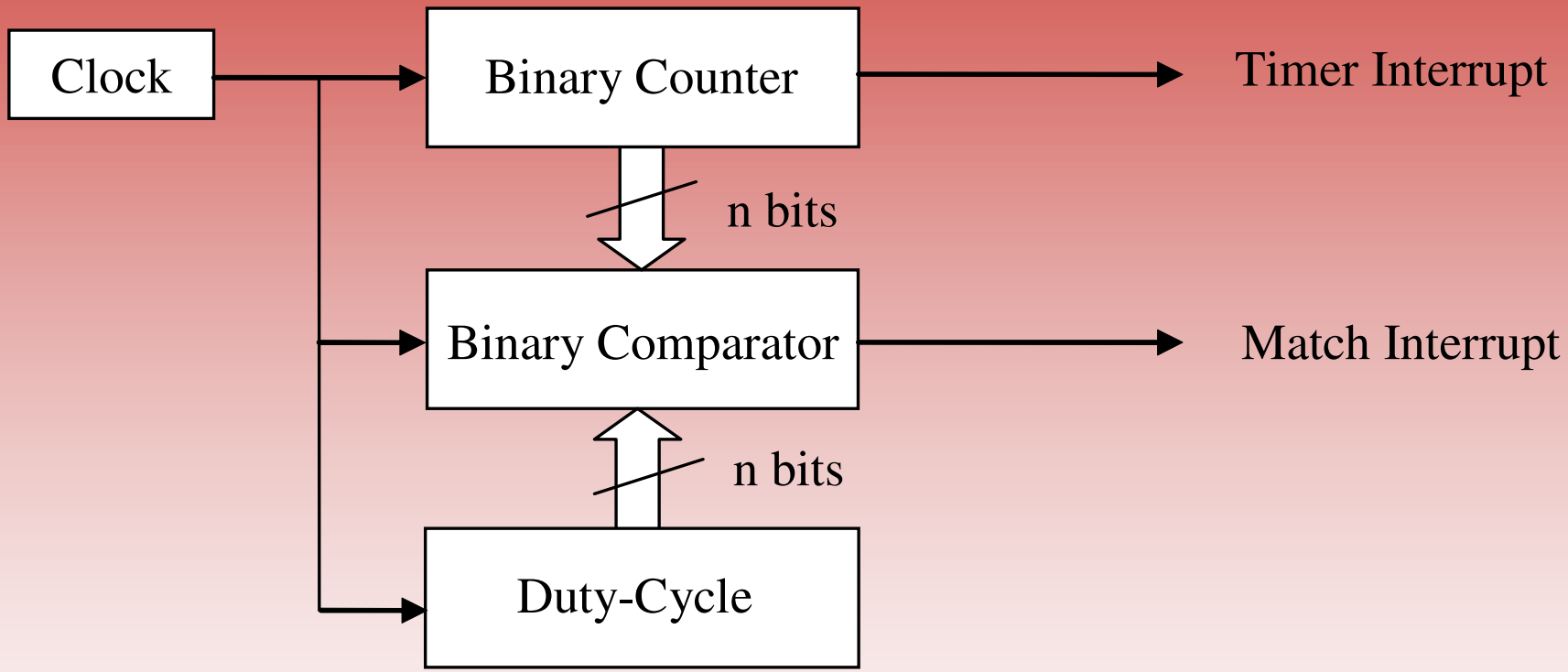
It is important to notice that the opposite commutation, i.e. where $S_1$ is turned off and $S_2$ is turned on, does not determine any voltage error. However, we must point out that, if the load current polarity were reversed, the dead time induced load voltage error would take place exactly during this commutation.

The above discussion reveals that, because of dead times, no matter what the modulator implementation, an error on the load voltage will always be generated. This error, $\Delta V_{OC}$, whose entity is a direct function of dead time duration and whose polarity depends on the load current sign, according to the following relation

$$\Delta V_{OC} = -2V_{DC}\frac{t_{dead}}{T_S}sign(I_O)$$

will have to be compensated by the current controller. Failure to do so will unavoidably determine a tracking error on the trajectory the load current has to follow (i.e. current waveform distortion).

## PWM modulator: digital implementation




Digital PWM modulator typical structure

## PWM modulator: digital implementation

The counter is incremented at every clock pulse; any time the binary counter value is equal to the programmed duty-cycle (match condition), the binary comparator triggers an interrupt to the microprocessor and, at the same time, sets the gate signal low.

The gate signal is set high at the beginning of each counting (i.e. modulation) period, where another interrupt is typically generated for synchronization purposes.

The counter and comparator have a given number of bits, *n,* which is often 16, but can be as low as 8, in the case a very simple microcontroller is used.

Depending on the ratio between the durations of the modulation period and the counter clock period, a lower number of *effective* bits, $N_e$, could be available to represent the duty-cycle. The $N_e$ parameter is important to determine the duty-cycle *quantization step.*

## PWM modulator: digital implementation

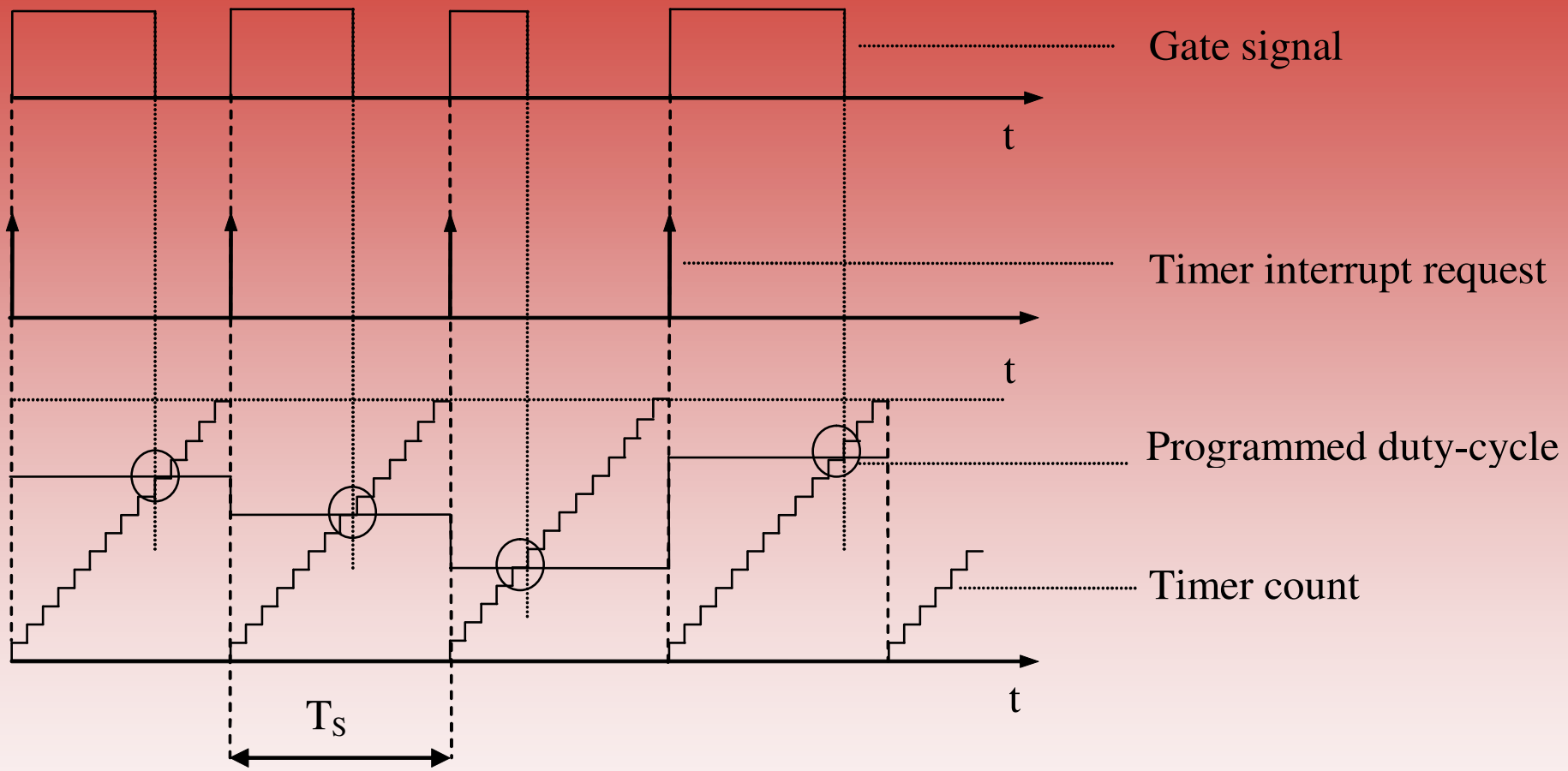
The number $N_e$ of effective bits, used to represent the duty-cycle, is given by the following relation

$$N_e = floor\left[\frac{\log_{10}\left(\frac{f_{clock}}{f_S}\right)}{\log_{10} 2}\right] + 1$$

where $f_{clock}$ is the modulator clock frequency, $f_S=1/T_S$ is the desired modulation frequency and the floor function calculates the integer part of its argument. Typical values for $f_{clock}$ are in the few tens of MHz range, while modulation frequencies can be as high as a few hundreds of kHz.

When the desired modulation period is short, the number of effective bits, $N_e$, will be much lower than the number of hardware bits, $n$, available in the comparator and counter circuits, unless a very high clock frequency is possible.

## PWM modulator: digital implementation



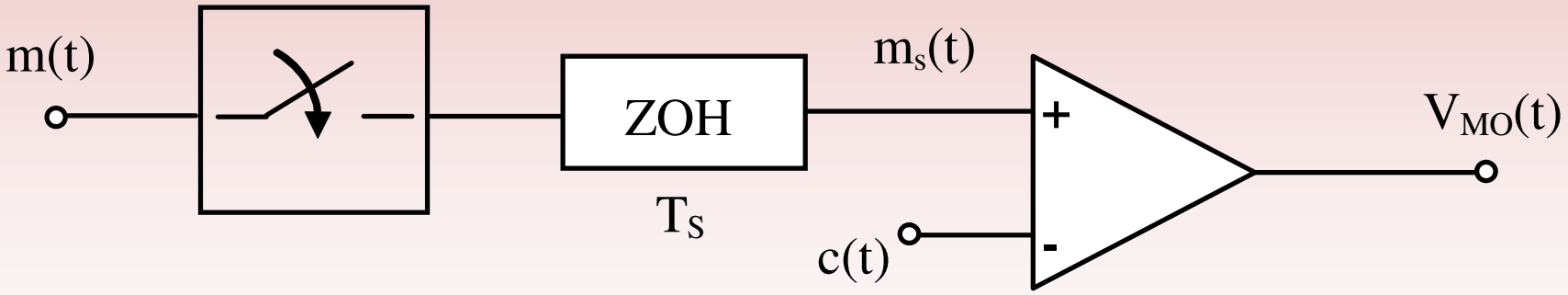

Digital PWM operation principle

## Digital PWM modulator: dynamic response

It is immediate to see that the modulating signal update is performed only at the beginning of each modulation period.

We can model this mode of operation using a sample and hold equivalent.

Indeed, we can observe that, neglecting the digital counter and binary comparator effects (i.e. assuming infinite resolution for both), the digital modulator works exactly as an analog one, where the modulating signal *m(t)* is sampled at the beginning of each modulation period and the sampled value held constant for the whole period.



## Digital PWM modulator: dynamic response

It is now evident that, because of the sample and hold effect, the response of the modulator to any disturbance, e.g. to one requiring a rapid change in the programmed duty-cycle value, can take place only during the modulation period following the one where the disturbance actually takes place.

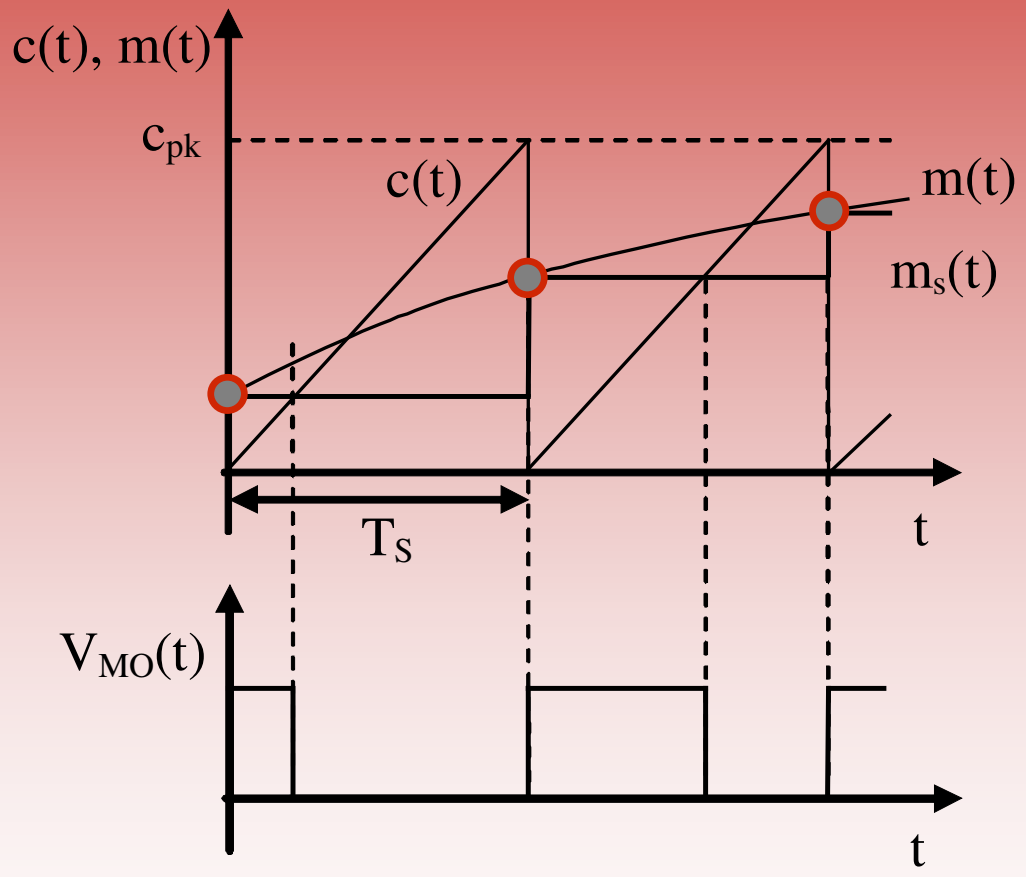
This delay amounts to a dramatic difference with respect to the analog modulator implementation, where the response could take place already during the current modulation period, i.e. with negligible delay.

Even if our signal processing were fully analog, without any calculation or sampling delay, passing from an analog to a digital PWM implementation would imply, by itself, an increase in the system's response delay.

We can now mathematically analyze the simplest implementation of the digital modulator, so as to determine its exact dynamic model.

# Digital PWM modulator: dynamic response

Digital PWM: trailing edge implementation

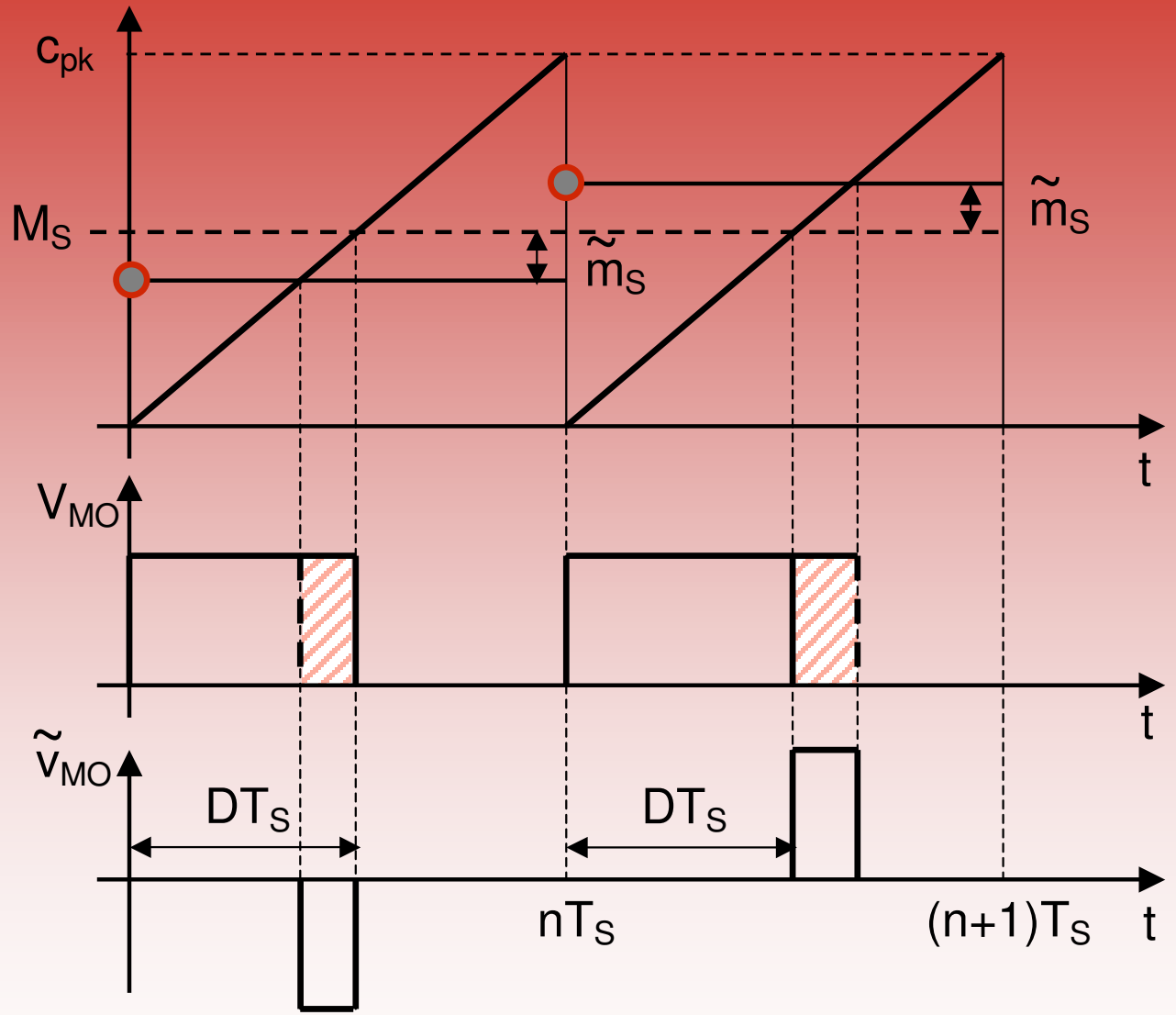



**Objective: we will now prove that**

$$PWM(s) = \frac{V_{MO}(s)}{M(s)} = \frac{e^{-sDT_S}}{c_{pk}}$$

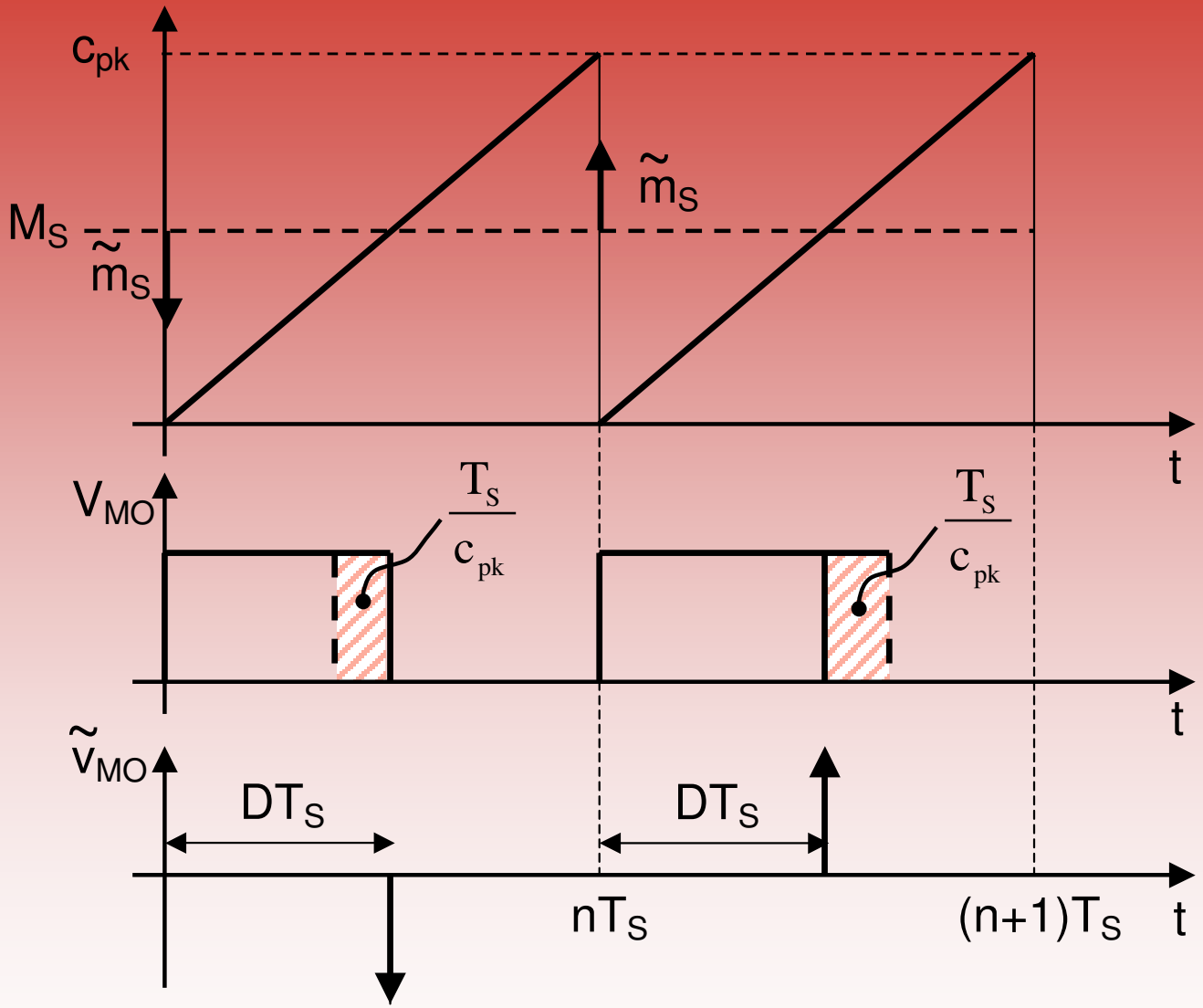
● : **sampling instants**

# Digital PWM modulator: dynamic response




Digital PWM: small signal analysis

## Digital PWM modulator: dynamic response




***Unity area Dirac impulse* perturbations**

⇩

***Correction pulses***

⇩

***Dirac impulse approximation of correction pulses***

Digital PWM: small signal analysis

## Digital PWM modulator: dynamic response

Considering small signal perturbations, $\tilde{m}_S$, of the steady state output of the sampler, $M_S$, we can see how these are turned into small correction pulses, appearing at the modulator output, $\tilde{v}_{MO}$.

The correction pulses can be approximated as ideal, zero duration impulses, with equal area, and located at the steady state pulse's edge.

The input perturbations can be, in particular, *unity area Dirac impulses* applied at the modulator input. Considering one of these impulses to be applied at time zero, we can immediately find that, in the above approximation, it generates a time translated impulse at the output:

$$\tilde{v}_{MO} = 1 \cdot \frac{T_S}{c_{pk}} \delta(t - DT_S)$$

whose area is equal to the modulator small signal gain (i.e. the inverse of the saw-tooth slope).

## Digital PWM modulator: dynamic response

Any generic discrete time sampled signal can be expressed as a sum of weighted Dirac pulses, such as:

$$\tilde{m}_S(t) = \sum_{n=-\infty}^{+\infty} \tilde{m}(nT_S) \cdot \delta(t - nT_S)$$

therefore, it is now possible to express the Laplace transform of the generic modulator output as a function of the sampled input signal's one. Since any input pulse is translated into a time shifted, scaled area, correction impulse we can write:

$$\tilde{v}_{MO}(t) = \sum_{n=-\infty}^{+\infty} \tilde{m}(nT_S) \cdot \frac{T_S}{c_{pk}} \cdot \delta(t - nT_S - DT_S)$$

We can now compute the Laplace transform of both sides of the above expression, exploiting the rule for time translation and the basic property of the Dirac pulse to have a unity Laplace transform.

## Digital PWM modulator: dynamic response

Consequently, we find the following relation:

$$V_{MO}(s) = \frac{T_S}{c_{pk}} e^{-sDT_S} M_S(s)$$

where

$$M_S(s) = \sum_{n=-\infty}^{+\infty} \tilde{m}_S(nT_S) \cdot e^{-snT_S}$$

which, by the way, happens to be the equivalent to the Z-transform of the sequence $\tilde{m}_S(nT_S)$. It is now possible to relate the Laplace transform of the sampled data sequence, $M_S(s)$, with the original signal's one, $M(s)$. We can write:

$$M_S(s) = \frac{1}{T_S} \sum_{k=-\infty}^{+\infty} M\left(s - jk\frac{2\pi}{T_S}\right)$$

## Digital PWM modulator: dynamic response

If we assume, as usual, that the input signal spectrum is limited in bandwidth below the Nyquist frequency, and if we neglect the output signal frequency content above the same frequency, then we can say:

$$M_S(s) \cong \frac{1}{T_S} M(s)$$

And, consequently,

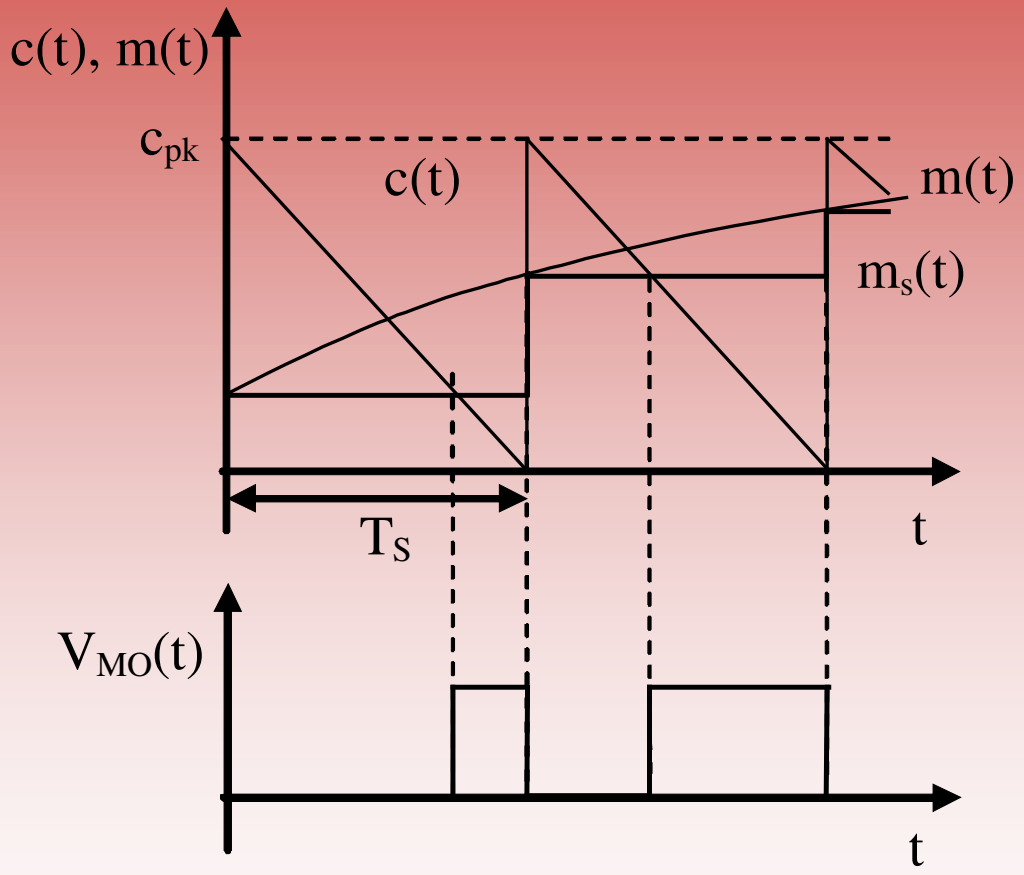
$$PWM(s) = \frac{V_{MO}(s)}{M(s)} = \frac{e^{-sDT_S}}{c_{pk}}$$

that represents the transfer function between the modulator input and output signals. A similar procedure can be applied to other, more complex, modulator organizations. Another useful relation, that we will use later on, is the following:

$$T_S \cdot PWM(s) = \frac{V_{MO}(s)}{M_S(s)}$$

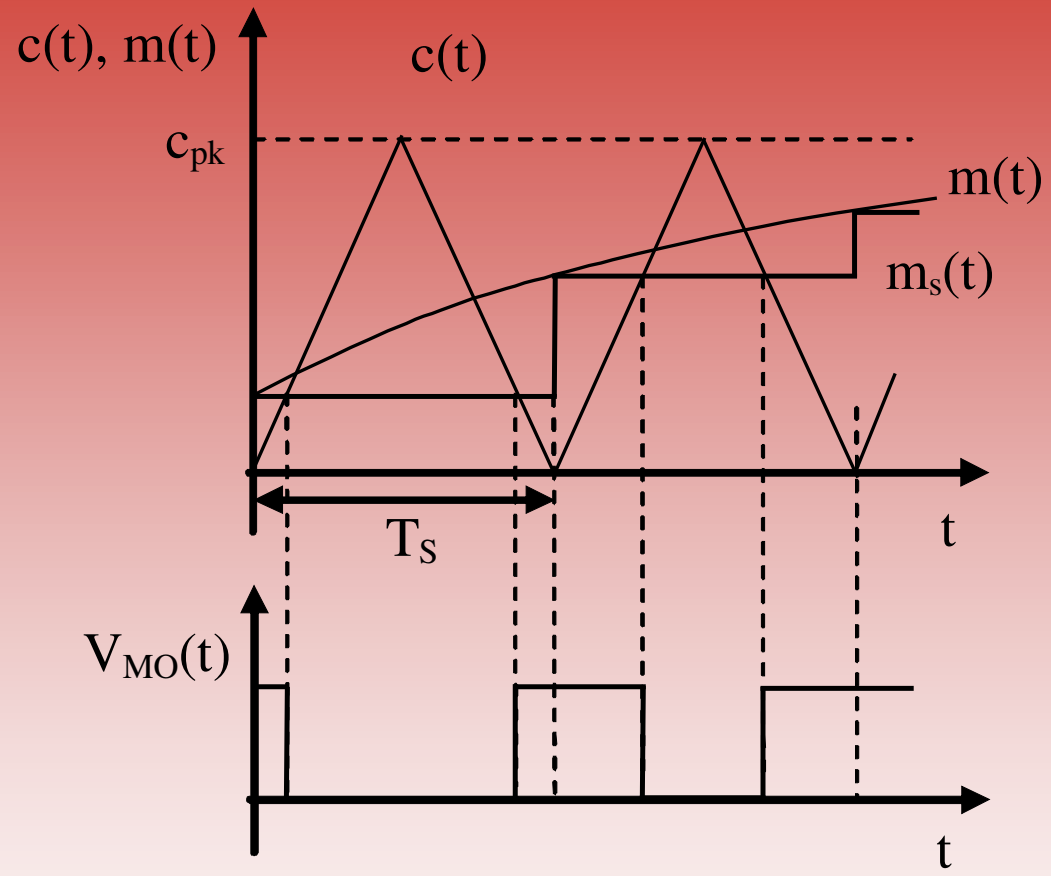
# Digital PWM modulator: dynamic response

Digital PWM: leading edge implementation




$$PWM(s) = \frac{V_{MO}(s)}{M(s)} = \frac{e^{-s(1-D)T_S}}{c_{pk}}$$

## Digital PWM modulator: dynamic response




Digital PWM: symmetric pulse implementation

$$PWM(s) = \frac{V_{MO}(s)}{M(s)} = \frac{1}{2c_{pk}} \left( e^{-s(1-D)\frac{T_S}{2}} + e^{-s(1+D)\frac{T_S}{2}} \right)$$

## Digital PWM modulator: dynamic response

The transfer functions we just found correspond to a non instantaneous behavior of the digital modulator. As can be seen by computing arg(PWM(jω)) there will always be a phase shift between the input and output signal, whose entity is, in general, a function of the steady state duty-cycle value. For example, in the case of the single update, trailing edge implementation we can find:

$$\arg(PWM(j\omega)) = \arg\left(\frac{e^{-j\omega DT_S}}{c_{pk}}\right) = -\omega DT_S$$

Similarly, for the symmetric pulse implementation we find:

$$\arg(PWM(j\omega)) = \arg\left(\frac{e^{-j\omega(1-D)\frac{T_S}{2}} + e^{-j\omega(1+D)\frac{T_S}{2}}}{2c_{pk}}\right) = -\omega\frac{T_S}{2}$$

which is a remarkable result, as it does not depend on the particular steady-state value of the duty-cycle, D.

## Digital PWM modulator: dynamic response

To partially compensate for the increased delay of the uniformly sampled PWM, the double update mode of operation is often available in several microcontrollers and DSPs.

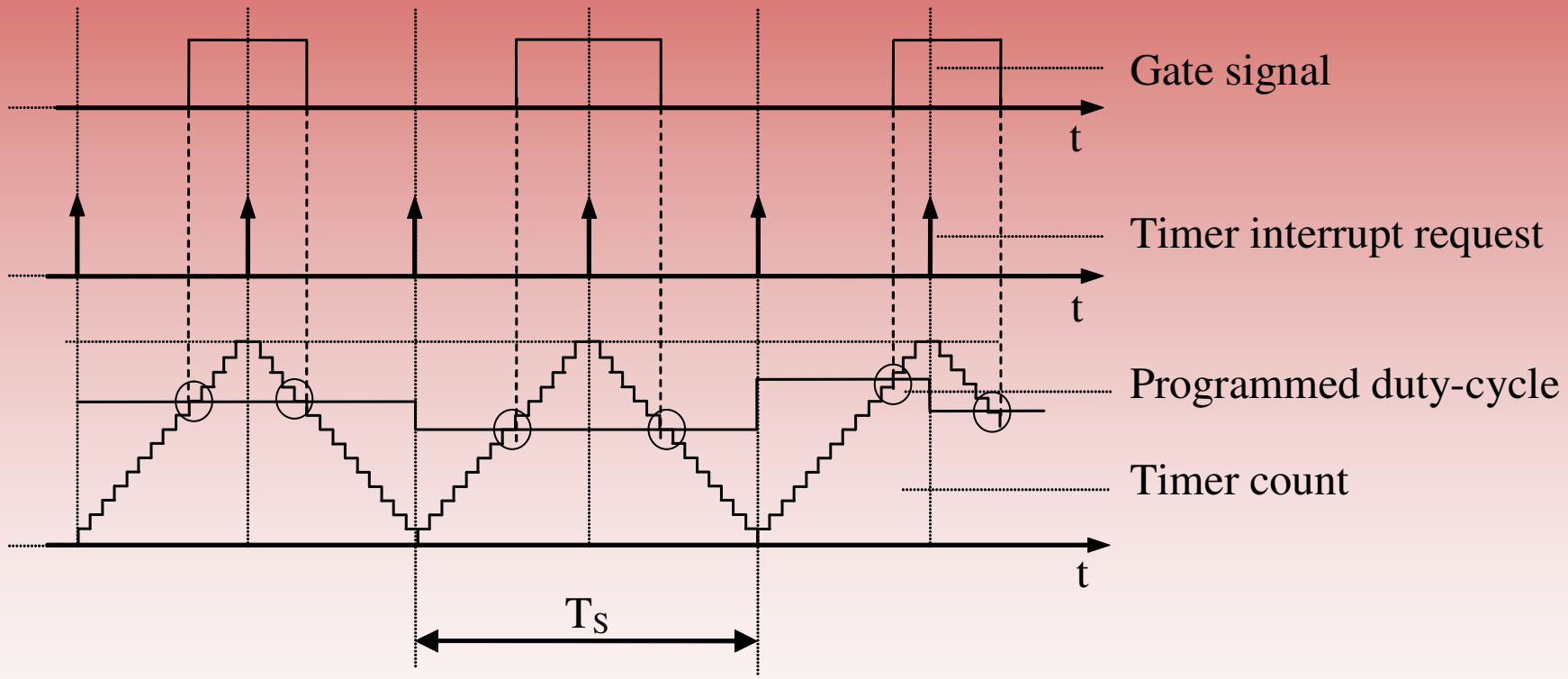
In this mode, the duty-cycle update is allowed at the beginning and at the half of the modulation period. Consequently, in each modulation period, the match condition between counter and duty-cycle registers is checked twice, at first during the run up phase, then during the run down phase. In the occurrence of a match, the state of the gate signal is toggled.

The result of this mode of operation is a stream of gate pulses that are symmetrically allocated within the modulation period, at least in the absence of any perturbation.

Interrupt requests are generated by the timer at the beginning and at the half of the modulation period, to allow proper synchronization with other control functions, e.g. with the sampling process.

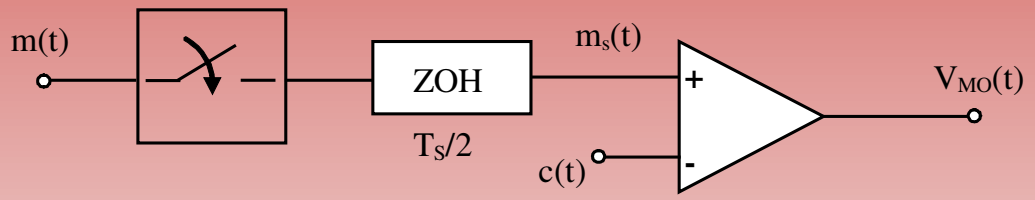
# Digital PWM modulator: dynamic response

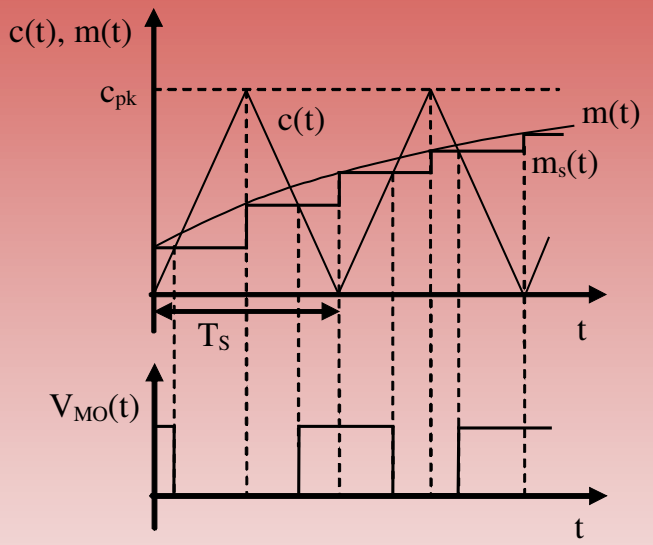
*Digital PWM: double update implementation*



# Digital PWM modulator: dynamic response

*Digital PWM: double update implementation*




$$PWM(s) = \frac{V_{MO}(s)}{M(s)} = \frac{1}{2c_{pk}}\left( e^{-sD\frac{T_s}{2}} + e^{-s(1-D)\frac{T_s}{2}} \right)$$

## Digital PWM modulator: dynamic response

This modulator implementation can be analyzed as well using a sample and hold equivalent. In this case, the sampling frequency is set to the double of the modulation frequency. The analysis proceeds following the same approach we have used for the basic modulator implementation.

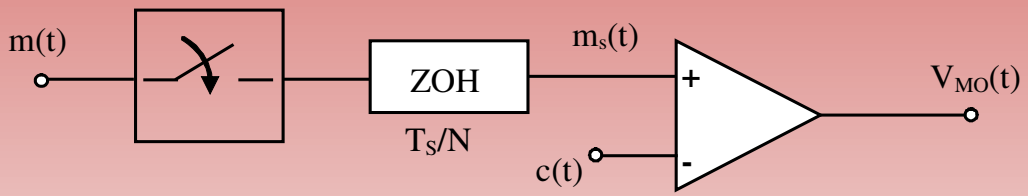
Interestingly, the transfer function we can derive in this case presents a similar structure with respect to the symmetric pulse modulator's one. However, the modulator's phase lag in this case turns out to be equal to:

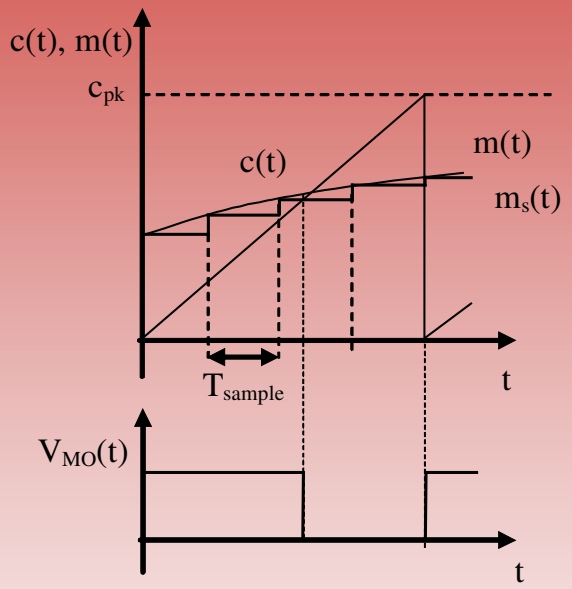
$$\arg(PWM(j\omega)) = \arg\left(\frac{e^{-j\omega D\frac{T_S}{2}} + e^{-j\omega(1-D)\frac{T_S}{2}}}{2c_{pk}}\right) = -\omega\frac{T_S}{4}$$

which is exactly ½ of the previously obtained one. This suggests the generalization of the technique, leading to the so-called multi-sampling PWM implementations.

# Digital PWM modulator: dynamic response

*Digital PWM: multi-sampled implementation*




$$PWM(s) = \frac{1}{c_{pk}} e^{-st_d}$$

where $t_d = DT_S - \frac{\text{floor}(ND)}{N} T_S$

Trailing edge delay

Multi-sampling effect

## Digital PWM modulator: dynamic response

*Digital PWM: multi-sampled implementation*

The equivalent delay is equal to the one found for the conventional trailing edge implementation, reduced by the so-called multi-sampling effect.
It is interesting to observe that, as N tends to infinity, the equivalent delay tends to zero, which is consistent with a continuous time, naturally sampled implementation of the modulator, where the sample and hold effect is not present.

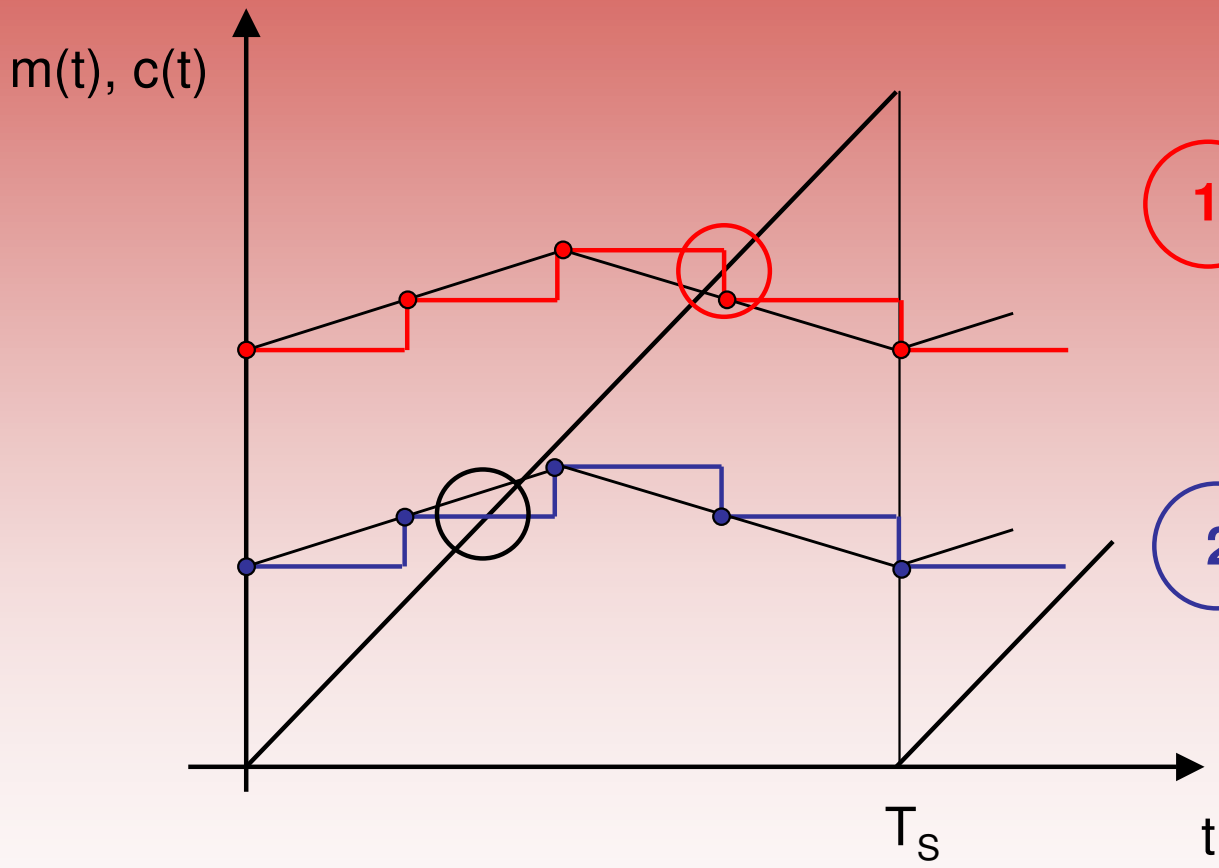
Multi-sampling presents some limitations as well, namely:

- need for proper filtering of the switching noise;
- need for non conventional hardware;
- generation of dead bands.

Research investigates possible means to overcome the limitations and fully exploit the advantages of multi-sampling.

## Digital PWM modulator: dynamic response

*Digital PWM: multi-sampled implementation*




1 Vertical intersection: the modulator gain is zero.

2 Horizontal intersection: the modulator gain is $1/c_{pk}$.

Generation of dead bands.

## Digital PWM modulator: dynamic response

*Digital PWM: multi-sampled implementation*

The presence of zero gain regions in the multi-sampled modulator trans-characteristic increases the settling time during transients and generates sub-harmonic oscillations in the steady state.

One possible way to compensate for these undesired effects consists in suitably synchronizing the sampling process and the modulator (i.e. the carrier wave) so that only horizontal intersections are allowed to take place [3].

[3] L. Corradini, P.Mattavelli, “Modeling of Multisampled Pulse Width Modulators for Digitally Controlled DC–DC Converters”, IEEE Trans. on Power Electronics, Vol. 23, No. 4, July 2008, page(s) 1839-1847.

## Three phase systems

What we have just seen for single phase converters can be almost identically applied to three phase systems. When the three phase converter is characterized by four wires, i.e. three phases plus neutral, the application is straightforward, since a four wire three phase system is totally equivalent to three independent single phase systems. Of course, this particular situation does not deserve any further discussion. On the contrary, we need to apply a little more caution when we are dealing with a three phase system with insulated neutral, i.e. with a three-wire, three-phase system.

The $\alpha\beta$ ***transformation*** represents a very useful tool for the analysis and the modelling of three phase electrical systems. In general, a three phase linear electric system can be properly described in mathematical terms only by writing a set of tri-dimensional dynamic equations (integral and/or differential), providing a self consistent mathematical model for each phase. In some cases though, the existence of physical constraints makes the three models not independent from each other. In these circumstances the order of the mathematical model can be reduced without any loss of information. We will see a remarkable example of this in the following.

## Three phase systems

Supposing that it is physically meaningful to reduce the order of the mathematical model from three to two dimensions, $\alpha\beta$ transformation represents the most commonly used relation to perform the reduction of order.

To explain how it works we can consider a tri-dimensional vector $[x_a, x_b, x_c]$ that can represent any triplet of system's electrical variables (voltages or currents). We can now consider the following linear transformation, ,

$$\begin{bmatrix} x_\alpha \\ x_\beta \\ x_\gamma \end{bmatrix} = T_{\alpha\beta\gamma} \begin{bmatrix} x_a \\ x_b \\ x_c \end{bmatrix} = \sqrt{\frac{2}{3}} \begin{bmatrix} 1 & -1/2 & -1/2 \\ 0 & \sqrt{3}/2 & -\sqrt{3}/2 \\ 1/\sqrt{2} & 1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix} \begin{bmatrix} x_a \\ x_b \\ x_c \end{bmatrix}$$

that, in geometrical terms, represents a change from the set of reference axes denoted as *abc* to the equivalent one indicated as $\alpha\beta\gamma$.

## αβ Transformation

This change of reference axes takes place because the standard $R^3$ orthonormal base $B_{abc}$

$$B_{abc} = \{ \begin{bmatrix}1 & 0 & 0\end{bmatrix}^T, \begin{bmatrix}0 & 1 & 0\end{bmatrix}^T, \begin{bmatrix}0 & 0 & 1\end{bmatrix}^T \}$$

is replaced by the new base $B_{\alpha\beta\gamma}$

$$B_{\alpha\beta\gamma} = \sqrt{2/3}\{ \begin{bmatrix}1 & -1/2 & -1/2\end{bmatrix}^T, \begin{bmatrix}0 & \sqrt{3}/2 & -\sqrt{3}/2\end{bmatrix}^T, \begin{bmatrix}1/\sqrt{2} & 1/\sqrt{2} & 1/\sqrt{2}\end{bmatrix}^T \}$$

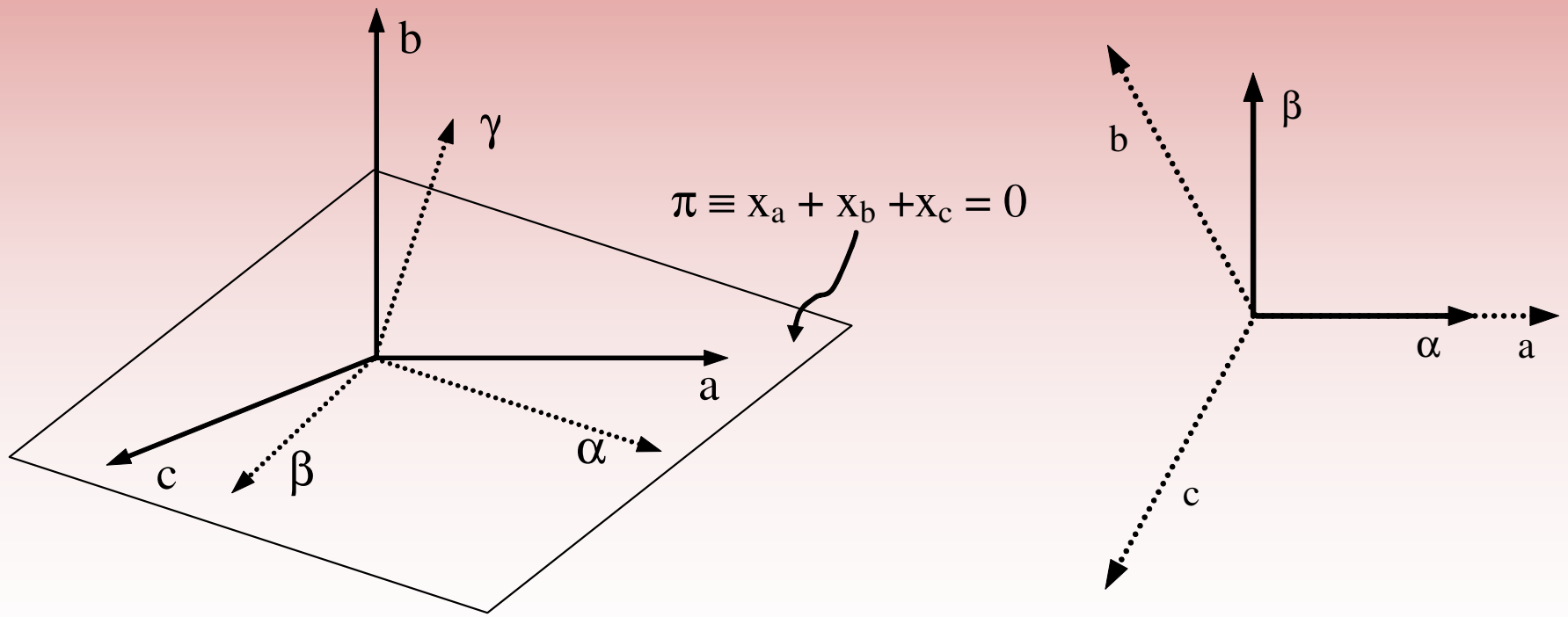
The $B_{\alpha\beta\gamma}$ base is once again orthonormal, i.e. its vectors have unity norm and are orthogonal to one another, thanks to the presence of the coefficient $\sqrt{2/3}$. Orthonormality implies that: *i)* the inverse of the transformation is equal to the matrix transposed and *ii)* the computation of electrical powers is independent from the transformation of coordinates.

## αβ Transformation

The transformation has an additional, interesting property, that becomes clear when we take into account the following condition

$$x_a + x_b + x_c = 0 \qquad \Rightarrow \qquad x_\gamma = 0$$

whose meaning is to operate the restriction of the tri-dimensional space to a plane $\pi$ (Fig. 4.1.1.a).



## αβ Transformation

We can therefore define the so called $\alpha\beta$ transformation as follows :

$$\begin{bmatrix} x_\alpha \\ x_\beta \end{bmatrix} = T_{\alpha\beta} \begin{bmatrix} x_a \\ x_b \\ x_c \end{bmatrix} = \sqrt{\frac{2}{3}} \begin{bmatrix} 1 & -1/2 & -1/2 \\ 0 & \sqrt{3}/2 & -\sqrt{3}/2 \end{bmatrix} \begin{bmatrix} x_a \\ x_b \\ x_c \end{bmatrix}$$

and its inverse as

$$\begin{bmatrix} x_a \\ x_b \\ x_c \end{bmatrix} = T^T_{\alpha\beta\gamma} \begin{bmatrix} x_\alpha \\ x_\beta \\ 0 \end{bmatrix} = \sqrt{\frac{2}{3}} \begin{bmatrix} 1 & 0 \\ -1/2 & \sqrt{3}/2 \\ -1/2 & -\sqrt{3}/2 \end{bmatrix} \begin{bmatrix} x_\alpha \\ x_\beta \end{bmatrix} = T^T_{\alpha\beta} \begin{bmatrix} x_\alpha \\ x_\beta \end{bmatrix}$$

## αβ Transformation

Considering the following example :

$$e_a = U_M \sin(\omega t)$$
$$e_b = U_M \sin(\omega t - 2\pi/3)$$
$$e_c = U_M \sin(\omega t + 2\pi/3)$$

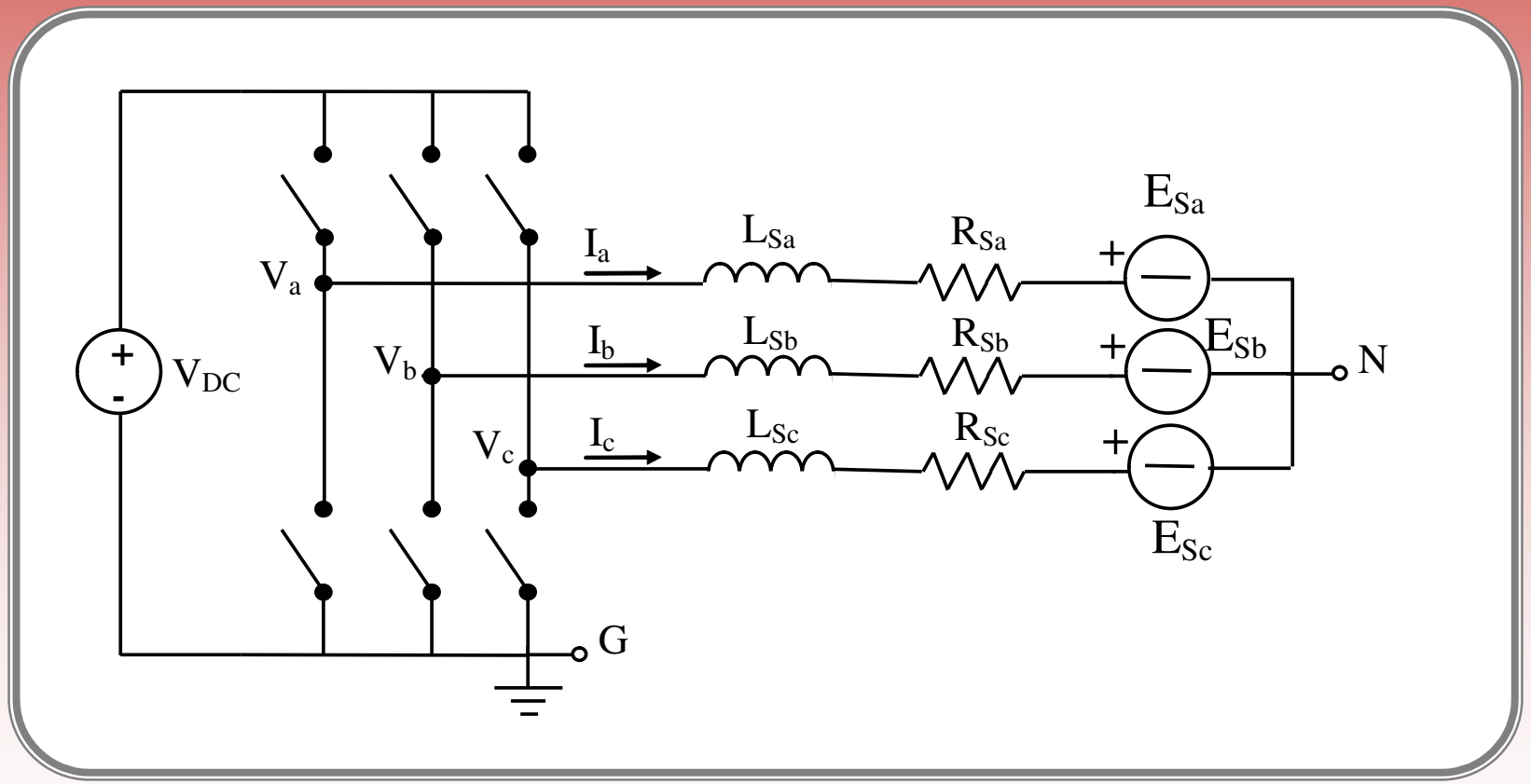
We get:

$$e_\alpha = \sqrt{\frac{3}{2}}\, U_M \sin(\omega t)$$

$$e_\beta = -\sqrt{\frac{3}{2}}\, U_M \cos(\omega t)$$

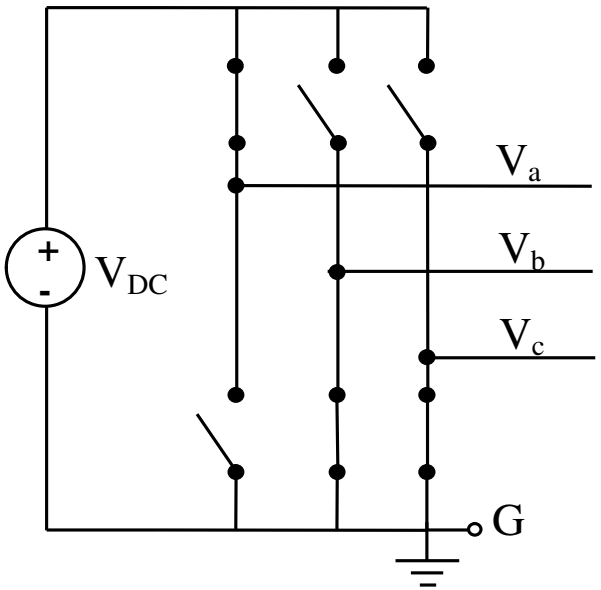
## Space Vector Modulation - SVM

We can consider a typical three phase voltage source inverter and represent the possible output voltage configurations as *vectors* on the αβ plane π.



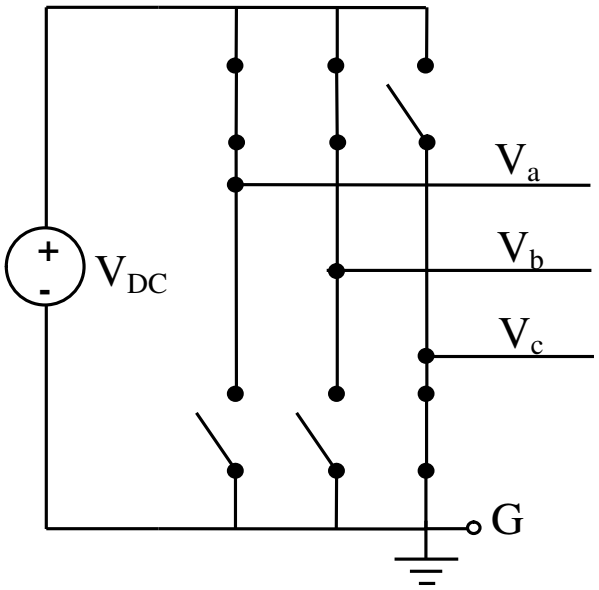
# Space Vector Modulation - SVM

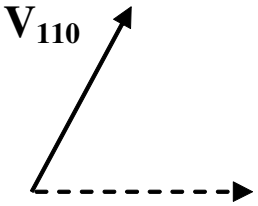
Vector 100: $V_a = V_{DC}$ $V_b=0$ $V_c=0$




# Space Vector Modulation - SVM

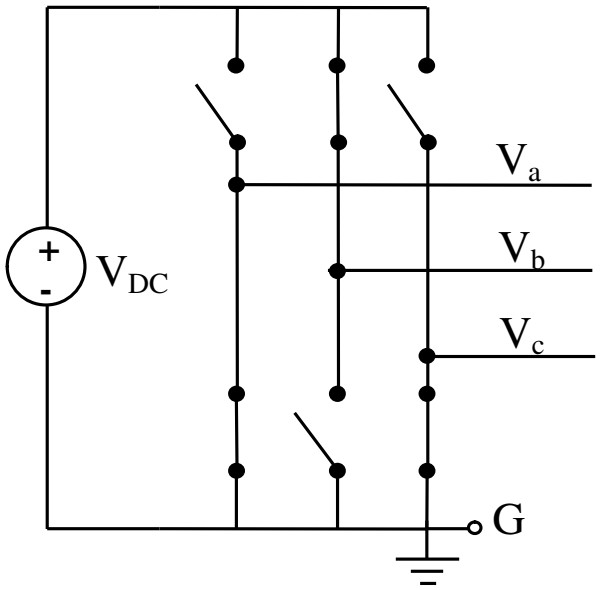
Vector 110: $V_a = V_{DC}$ $V_b = V_{DC}$ $V_c = 0$

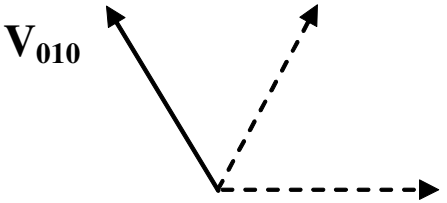


# Space Vector Modulation - SVM

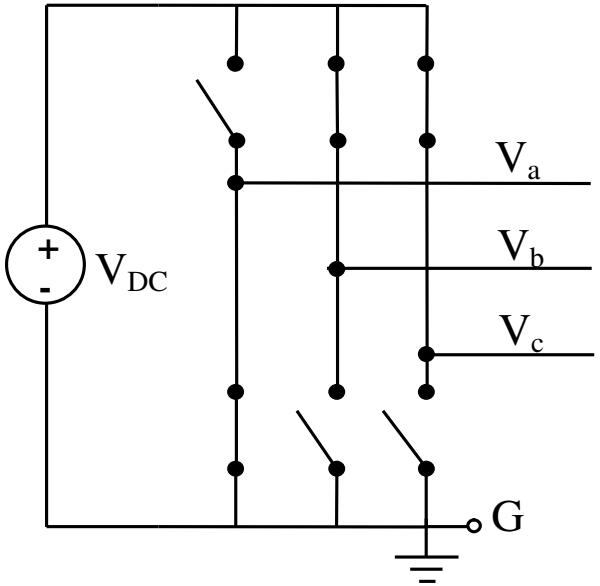
Vector 010: $V_a = 0$ $V_b = V_{DC}$ $V_c = 0$

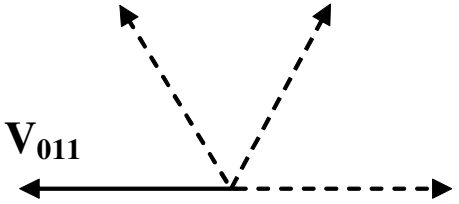


# Space Vector Modulation - SVM

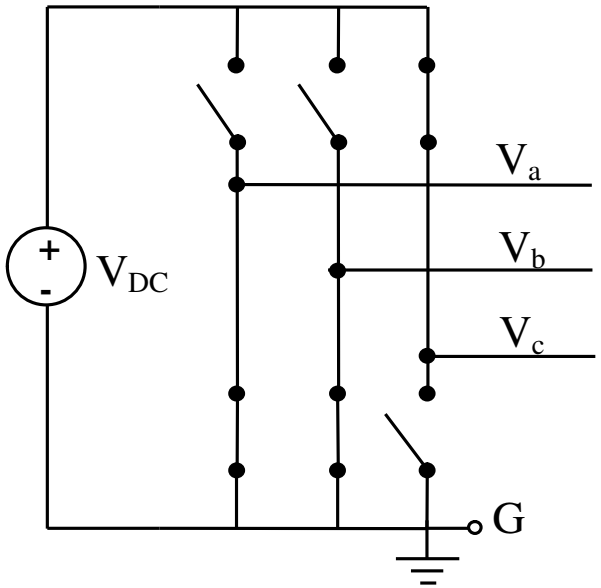
Vector 011: $V_a = 0$ $V_b = V_{DC}$ $V_c = V_{DC}$

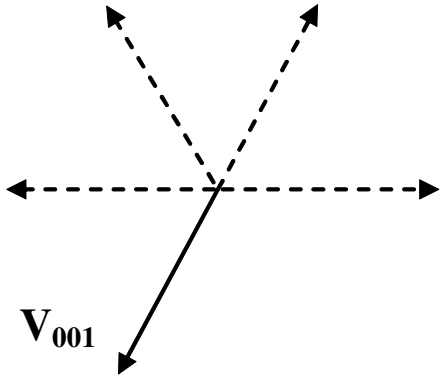


# Space Vector Modulation - SVM

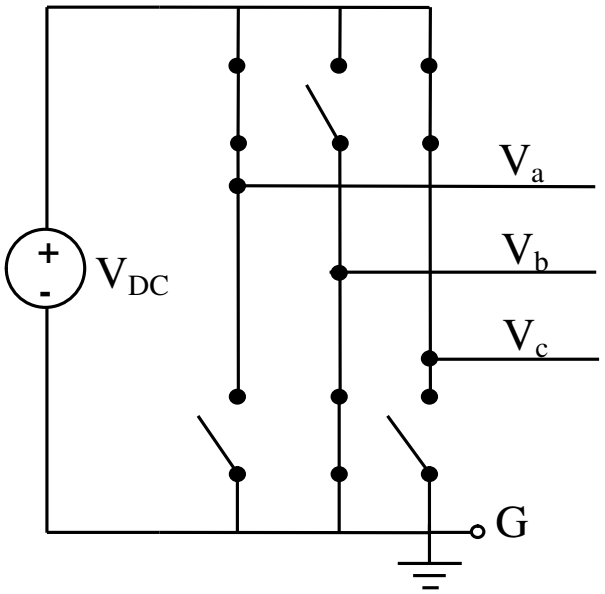
Vector 001: $V_a = 0$ $V_b = 0$ $V_c = V_{DC}$

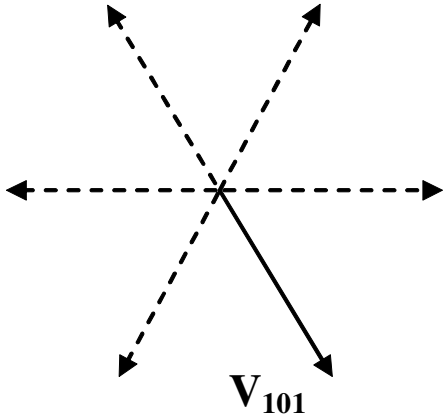


# Space Vector Modulation - SVM

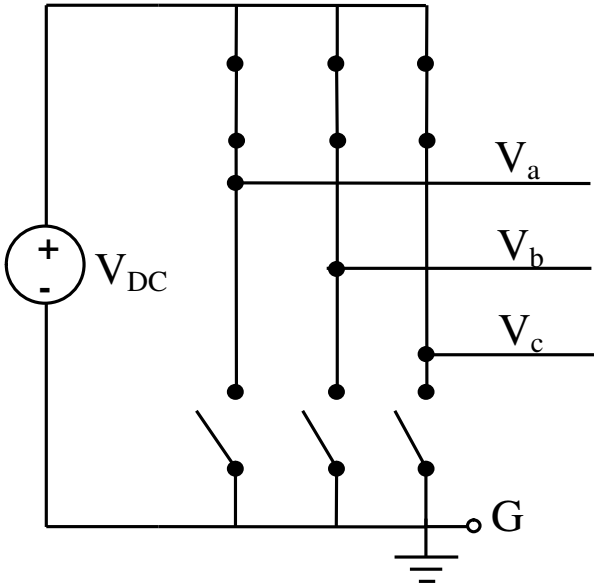
Vector 101: $V_a = V_{DC}$ $V_b = 0$ $V_c = V_{DC}$

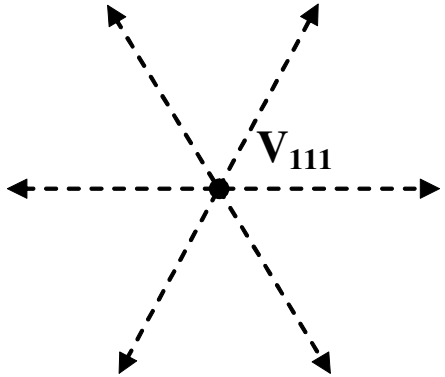


# Space Vector Modulation - SVM

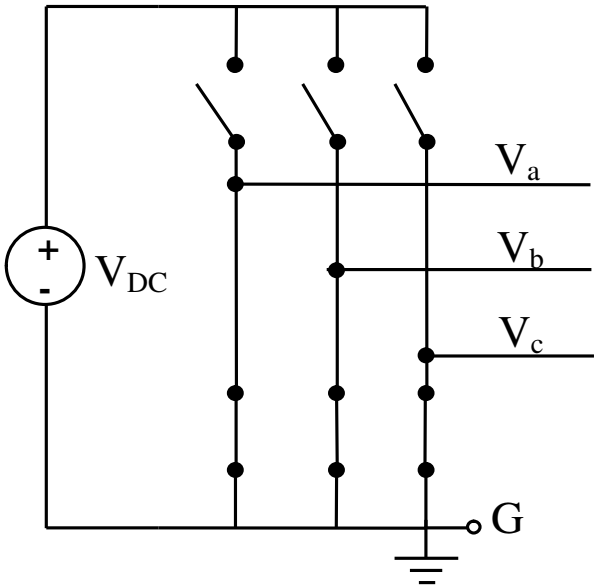
Vector 111: $V_a = V_{DC}$ $V_b = V_{DC}$ $V_c = V_{DC}$

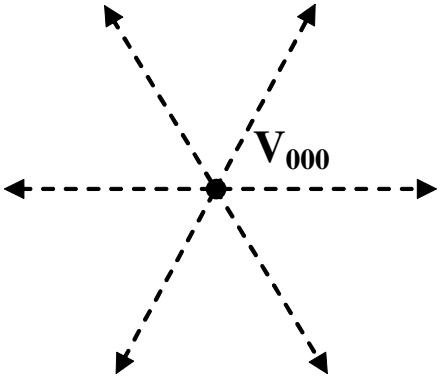


# Space Vector Modulation - SVM

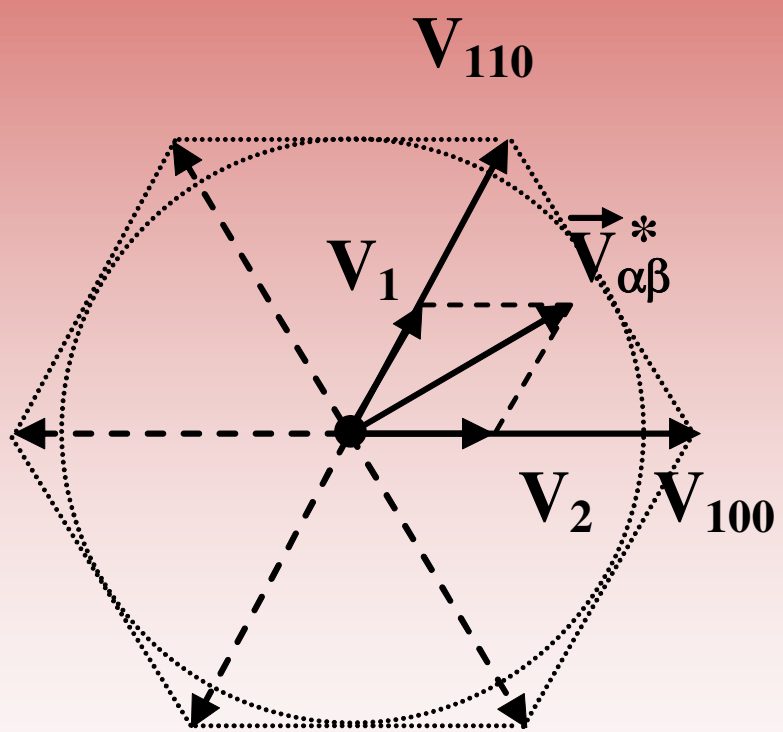
Vector 000: $V_a = 0$ $V_b = 0$ $V_c = 0$

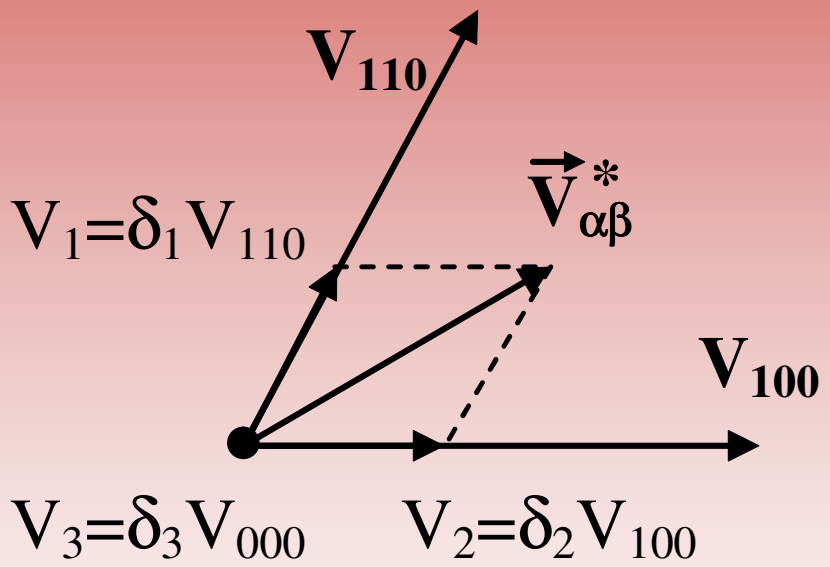


## Space Vector Modulation - SVM

The procedure of Space Vector Modulation can be explained referring to the following figure:



## Space Vector Modulation - SVM

The basic relations, used to compute the vector duty-cycles are the following:

$$\delta_1 = \frac{|V_1|}{|\vec{V}_{100}|} \qquad \delta_2 = \frac{|V_2|}{|\vec{V}_{110}|}$$

Considering that the sum of the three duty-cycles has to be 1, i.e. the whole modulation period must be occupied, we can derive the third of them, referred to the zero vector:

$$\delta_1 + \delta_2 + \delta_3 = 1$$

The average voltage vector generated by the inverter is therefore:

$$\overline{V}_o = \delta_1 V_{100} + \delta_2 V_{110} + \delta_3 V_{111} = V_1 + V_2 = \vec{V}^*_{\alpha\beta}$$

## Space Vector Modulation - SVM

It can be interesting to identify the locus of the constant amplitude rotating reference vectors that can be generated by the inverter without distortion.

This is represented by the circle *inscribed* in the vector hexagon. It is easy to verify that every vector that lays inside the circle generates a valid $\delta_1$, $\delta_2$, $\delta_3$, triplet. Instead, a vector that lays partially outside the circle cannot be generated by the inverter, because the sum of the corresponding $\delta_1$, $\delta_2$, $\delta_3$ becomes greater than unity.

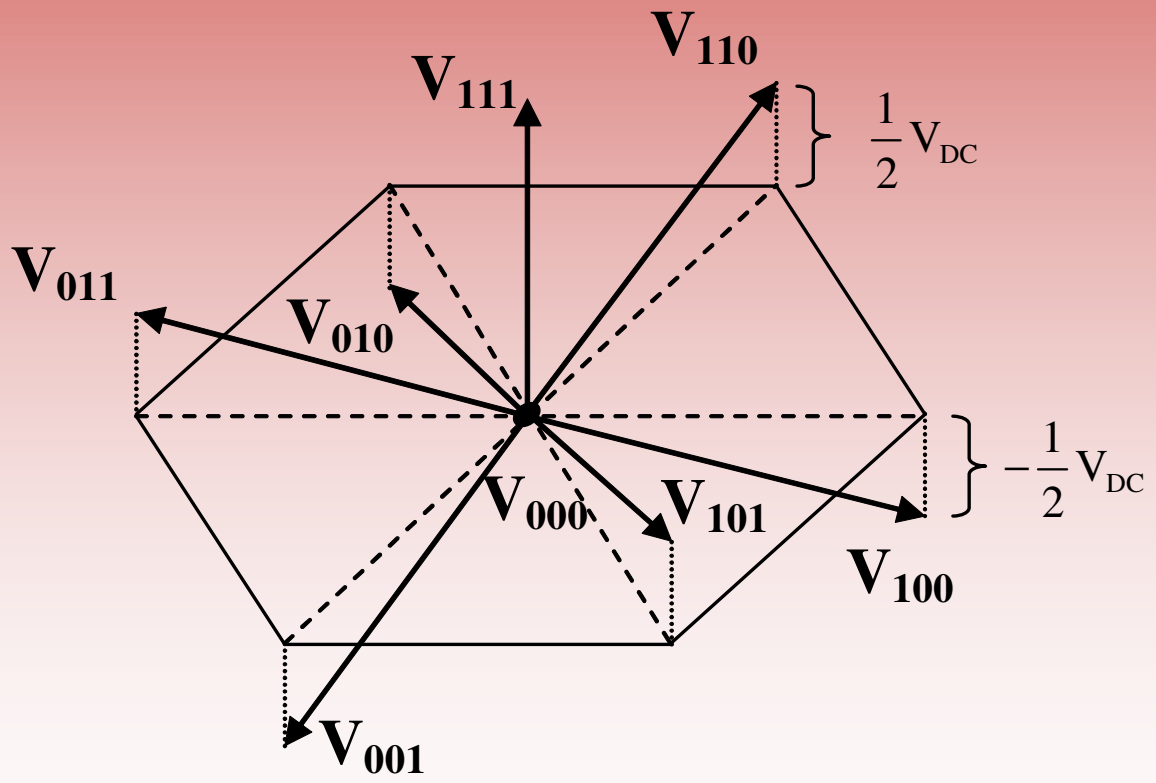
This situation is called inverter *saturation* and generally causes output voltage distortion.

It is easy to calculate the amplitude $U_{MMAX}$ of the voltage triplet that corresponds to a rotating vector having an amplitude equal to the radius of the inscribed circle. We find:

$$\sqrt{\frac{3}{2}}U_{MMAX} = \sqrt{\frac{2}{3}}V_{DC}\frac{\sqrt{3}}{2} \quad \Leftrightarrow \quad U_{MMAX} = \frac{2}{\sqrt{3}}\frac{V_{DC}}{2} \cong 1.15 \cdot \frac{V_{DC}}{2}$$

## Space Vector Modulation - SVM

Performing SVM, what is used to synthesize the desired output voltage vector is *not* the superposition of vectors laying on plane $\pi$. A more realistic representation of the inverter output vectors, that puts into evidence their $\gamma$ component, is shown here:



## Space Vector Modulation - SVM

The above observation means that SVM implies a particular modulation of the voltage between nodes N and G, $V_{NG}$. This is due to the common mode component of the inverter output voltage vectors. Indeed, it is easy to demonstrate that, in case of a symmetrical load structure, almost always encountered in practice, $V_{NG}$ is instantaneously and exactly equal to the $\gamma$ component of the inverter output voltage.

The most important implication of this fact is that the phase to neutral voltage of the load will always be insensitive to any common mode component of the inverter output voltage, i.e. one can freely add common mode components to the vector, without perturbing the load voltage.

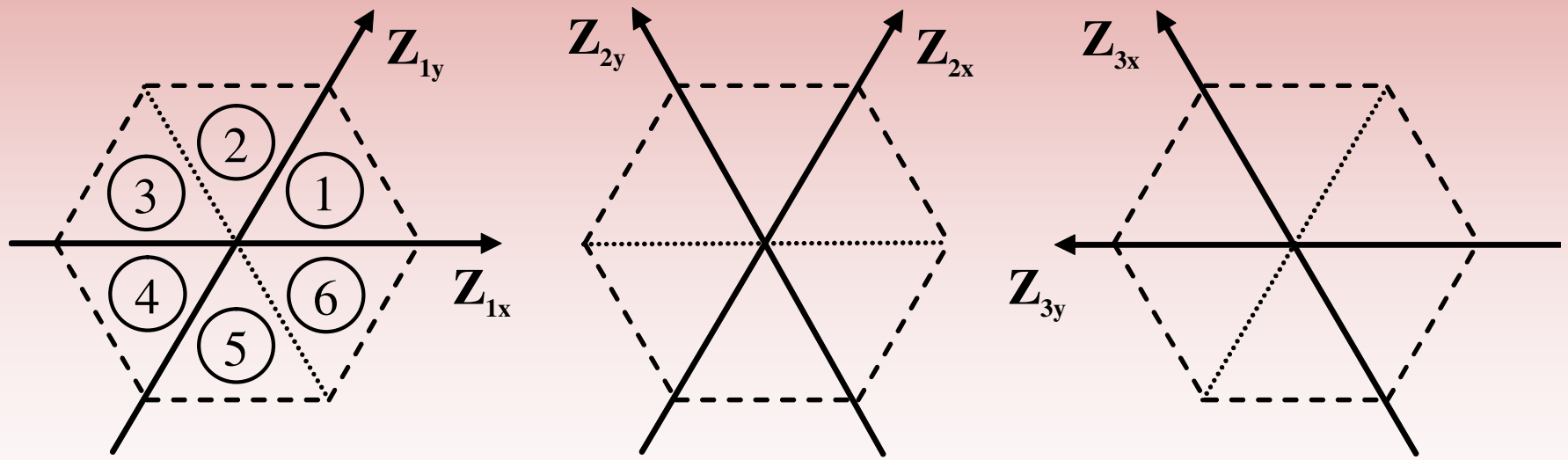
This is exactly what SVM implicitly does. Its effect, from the inverter's standpoint, can be proved to be very similar to that of third harmonic injection, sometimes employed in analog three phase PWM implementations.

An increase by 15% of the voltage amplitude range that corresponds to a linear converter operation, i.e. to the absence of any saturation phenomenon, is obtained, as clearly demonstrates.

## Space Vector Modulation - SVM

We consider now a possible implementation algorithm for space vector modulation, that can be directly programmed into a microcontroller or digital signal processor. The first issue in SVM implementation is the identification of the hexagon sector where the reference vector is laying.

This can be done by implementing once again a base change from the $\alpha\beta$ reference frame to a new set of three different reference frames.



## Space Vector Modulation - SVM

As can be seen, each frame refers to a particular couple of hexagon sectors. The method we propose simply requires the projection of the inverter output voltage reference vector onto each one of the three hexagon reference frames. This is easily implemented with the following set of reference base change matrixes:

$$M_1 = \begin{bmatrix} 1 & -\frac{1}{\sqrt{3}} \\ 0 & \frac{2}{\sqrt{3}} \end{bmatrix} \quad M_2 = \begin{bmatrix} 1 & \frac{1}{\sqrt{3}} \\ -1 & \frac{1}{\sqrt{3}} \end{bmatrix} \quad M_3 = \begin{bmatrix} 0 & \frac{2}{\sqrt{3}} \\ -1 & -\frac{1}{\sqrt{3}} \end{bmatrix}$$

that map the orthogonal set of axes $\alpha$ and $\beta$ onto the three, non-orthogonal sets *Z*. It is interesting to notice that the algorithm required to implement the three projections is quite simple.

## Space Vector Modulation - SVM

$$tmp = \frac{V_\beta^*}{\sqrt{3}}; \qquad \text{save to a temporary register}$$

$$Z_{1x} = V_\alpha^* - tmp; \qquad Z_{1x} \text{ found}$$

$$Z_{2y} = -Z_{1x}; \qquad Z_{2y} \text{ found}$$

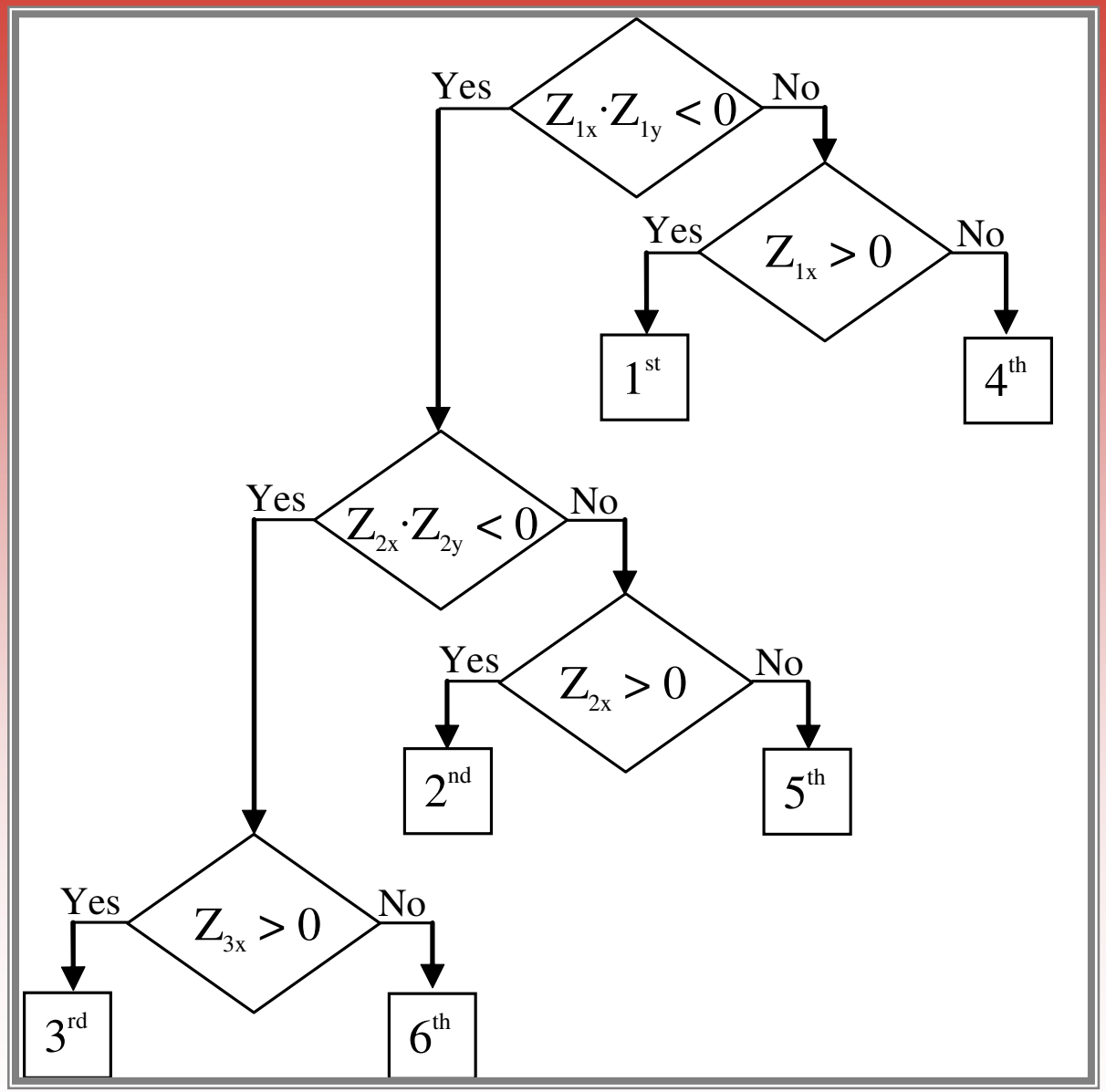
$$Z_{1y} = 2tmp; \qquad Z_{1y} \text{ found}$$

$$Z_{3x} = Z_{1y}; \qquad Z_{3x} \text{ found}$$

$$Z_{2x} = V_\alpha^* + tmp; \qquad Z_{2x} \text{ found}$$

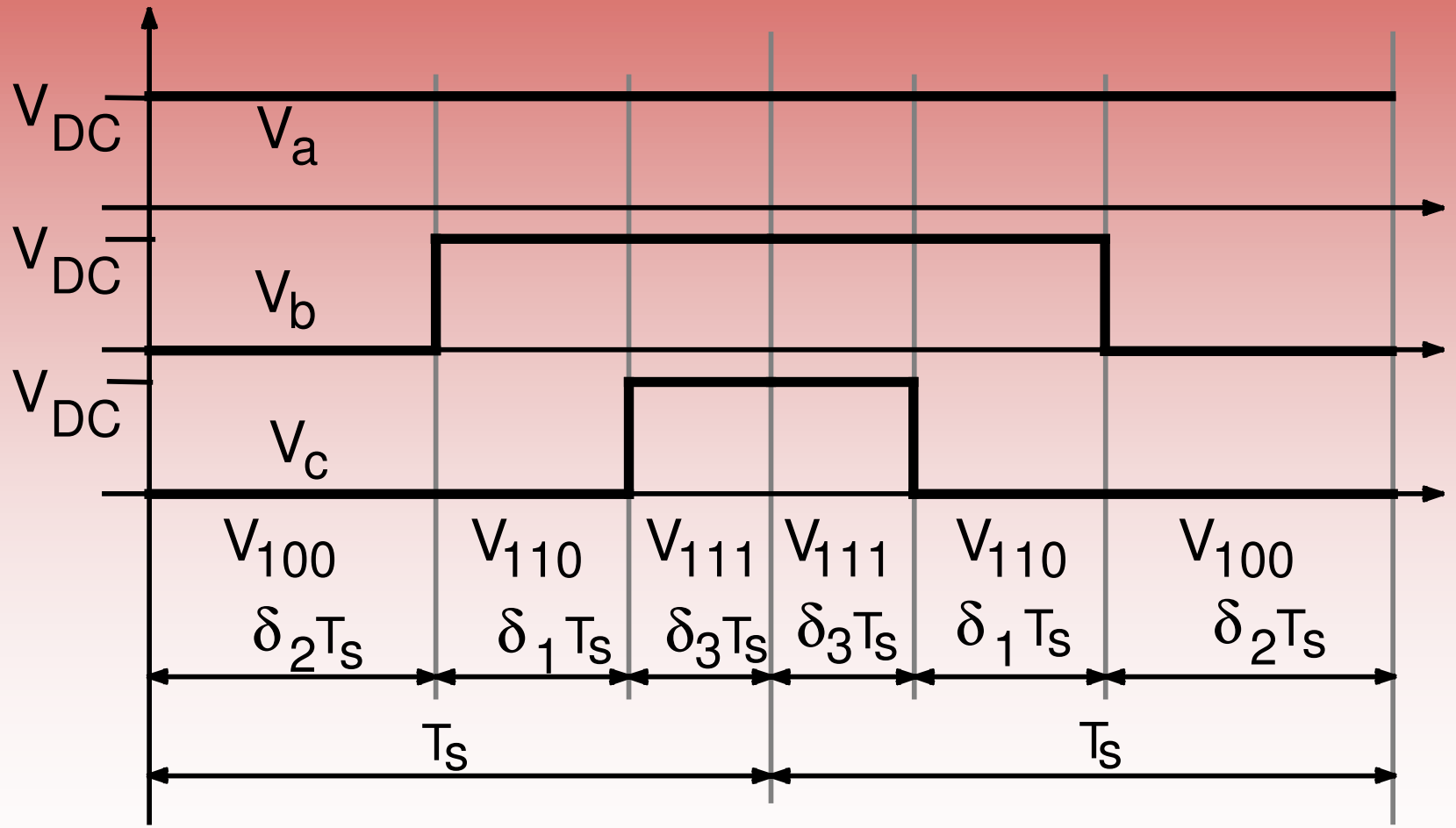
$$Z_{3y} = -Z_{2x}; \qquad Z_{3y} \text{ found}$$

# Space Vector Modulation - SVM



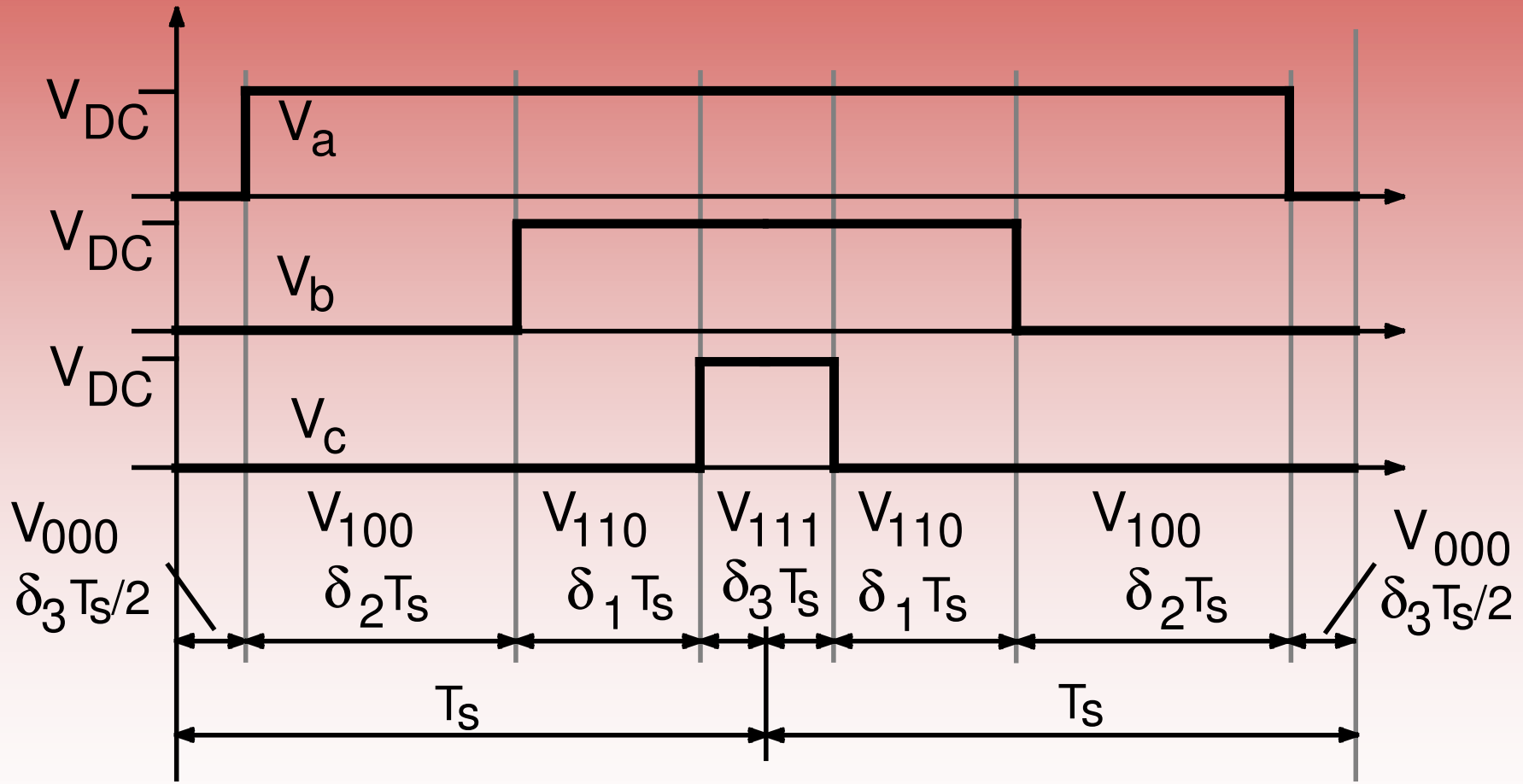
## Space Vector Modulation - SVM

There can be different possible generation sequences: depending on the controlled system characteristics, one can be more advantageous than the other. One is the following, that minimizes the commutations:



## Space Vector Modulation - SVM

While the following one minimizes the current ripple amplitude and, therefore, current distortion:



## Space Vector Modulation - SVM

The typical organization of a three-phase VSI controller based on SVM is shown here. As can be seen, the controller takes advantage of the application of $\alpha\beta$ transformations to operate on two sampled variables instead of three.