Digital Logic Design

Lecture 13

Oscillators

Multivibrators

A **multivibrator** is a circuit that changes between the two digital levels on a continuous, free-running basis or on demand from some external trigger source.

The **bistable multivibrator** is triggered into one of the two digital states by an external source and stays in that state until it is triggered into the opposite state.

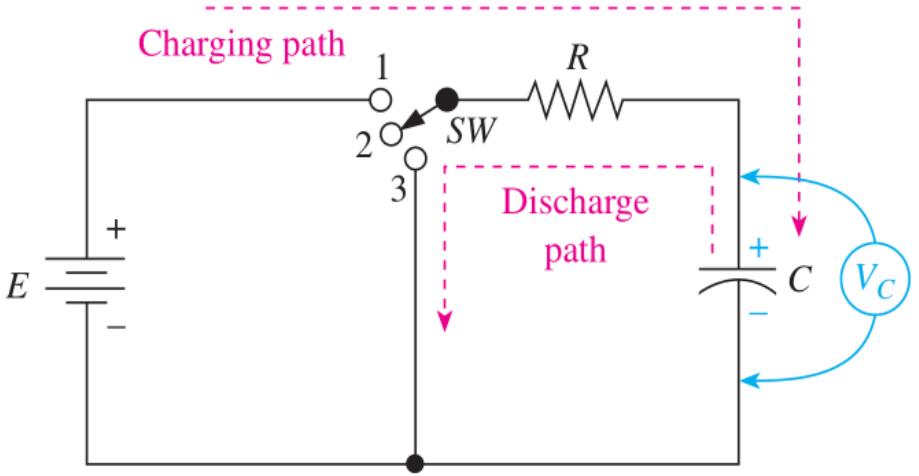
The S-R flip-flop is a bistable multivibrator.

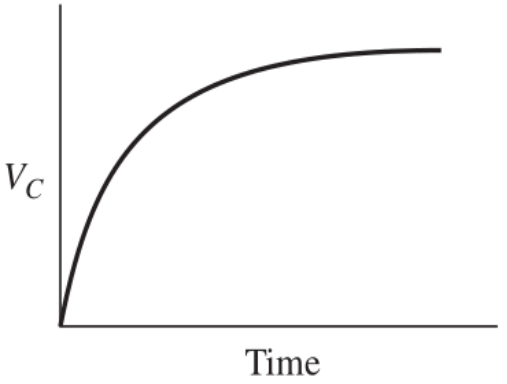
The **astable multivibrator** is a free-running oscillator that alternates between the two digital levels at a specific frequency and duty cycle.

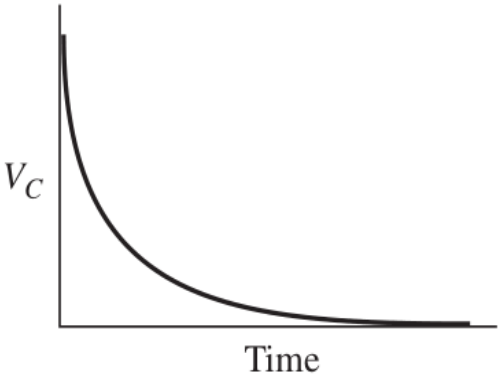
The **monostable multivibrator**, also known as a one shot, provides a single output pulse of a specific time length when it is triggered from an external source.

Capacitor charge and discharge rates

$$V_C = E\left(1 - e^{-t/RC}\right) \qquad V_C = E\,e^{-t/RC}$$



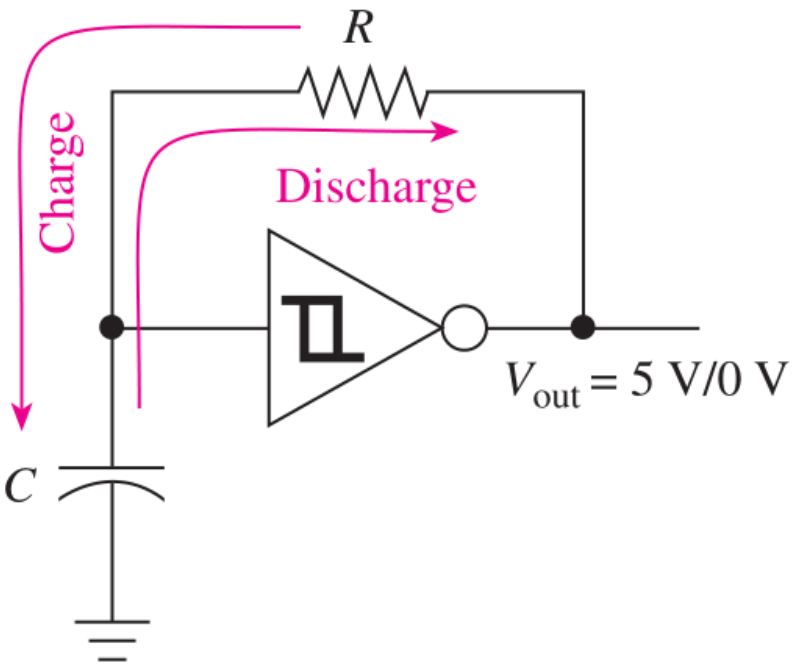




$$t = RC \ln\left(\frac{1}{1 - V_C/E}\right)$$

The predictable charging rate of capacitors can be used in the design of oscillator and timing circuits.

Schmitt trigger astable multivibrator



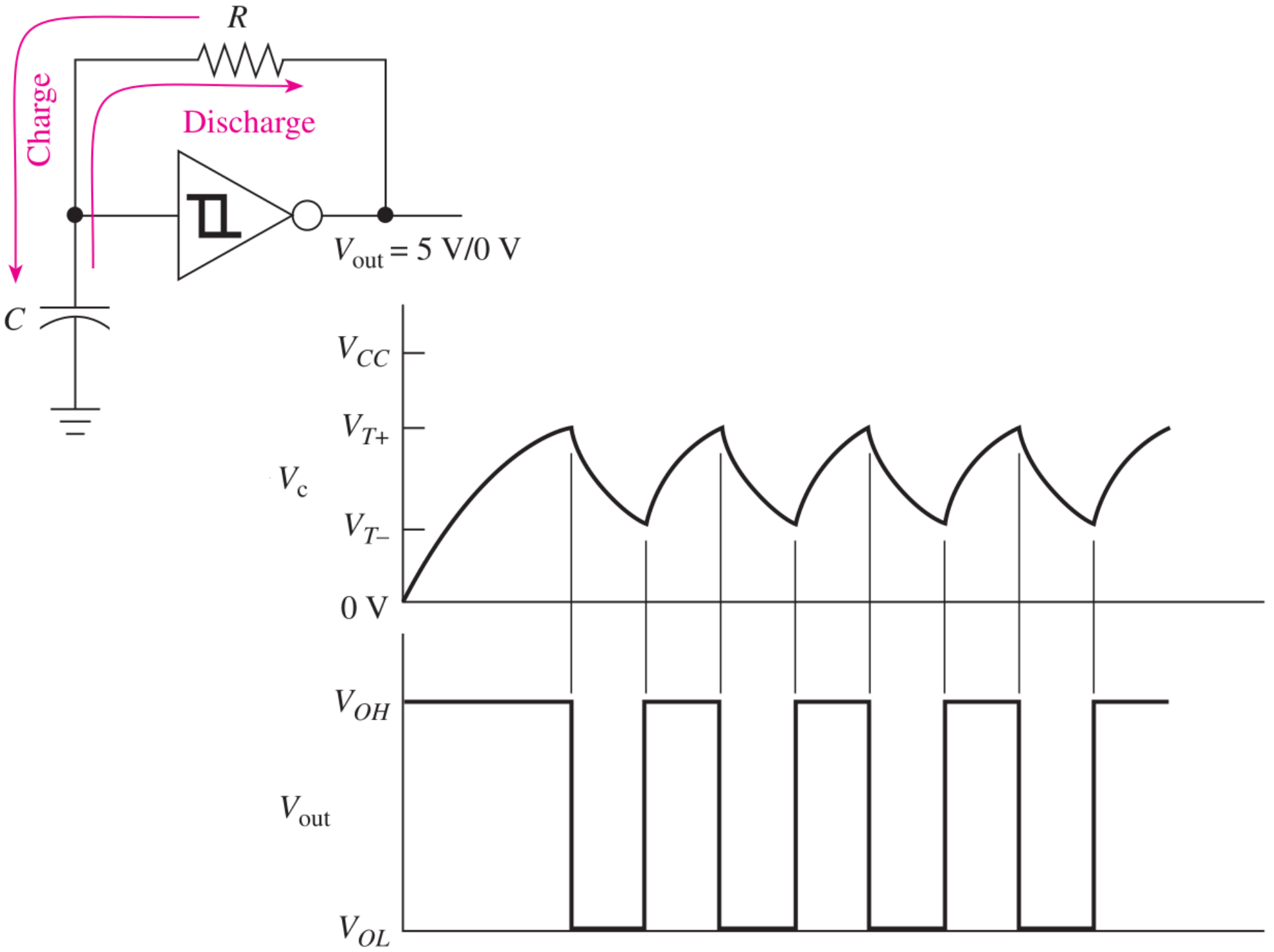
When the IC supply power is first turned on, V_C is 0 V, so V_{out} will be HIGH

When V_C reaches the positive-going threshold of the Schmitt trigger, the output of the Schmitt will change to a LOW

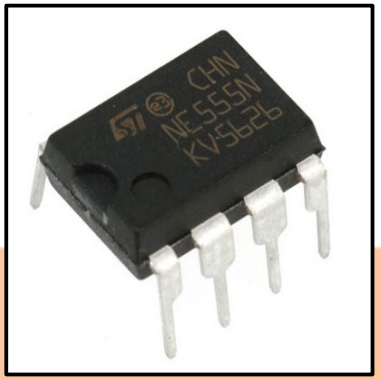
Now, with V_{out} = 0 the capacitor will start discharging.

When V_C drops below the negative-going threshold the output of the Schmitt will change back to a HIGH.

The cycle repeats indefinitely.


R
Charge
Discharge
V_{out} = 5 V/0 V
C
V_{CC}
V_{T+}
V_c
V_{T-}
0 V
V_{OH}
V_{out}
V_{OL}

555 timer IC



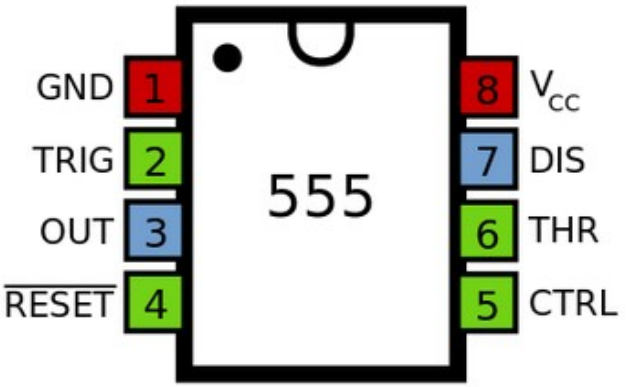
The threshold level is

$$\frac{2}{3}V_{CC}$$

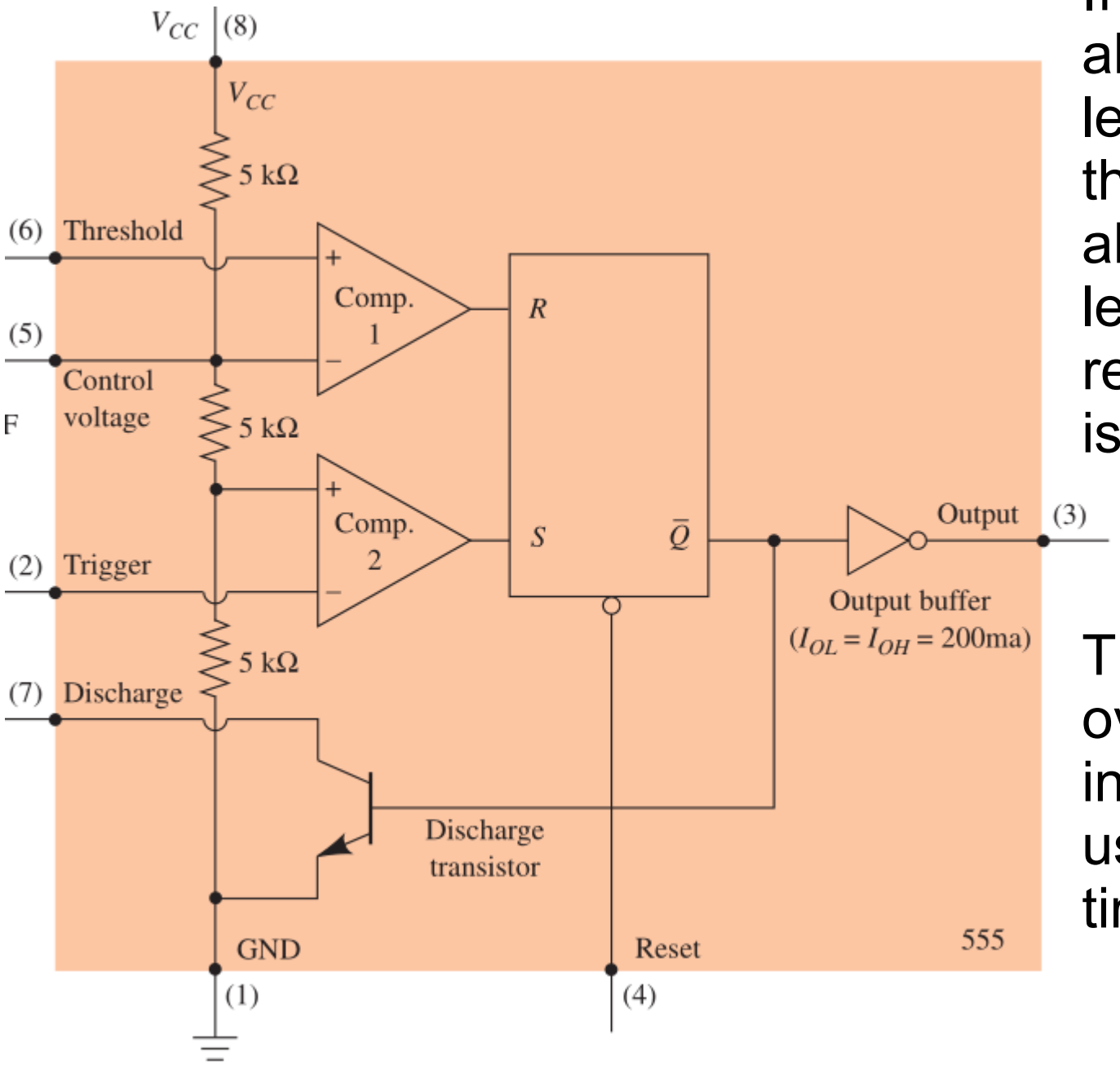
The trigger level is

$$\frac{1}{3}V_{CC}$$

These levels can be altered by use of the control-voltage terminal.



When the trigger input falls below the trigger level, the latch is set, and the output goes high.

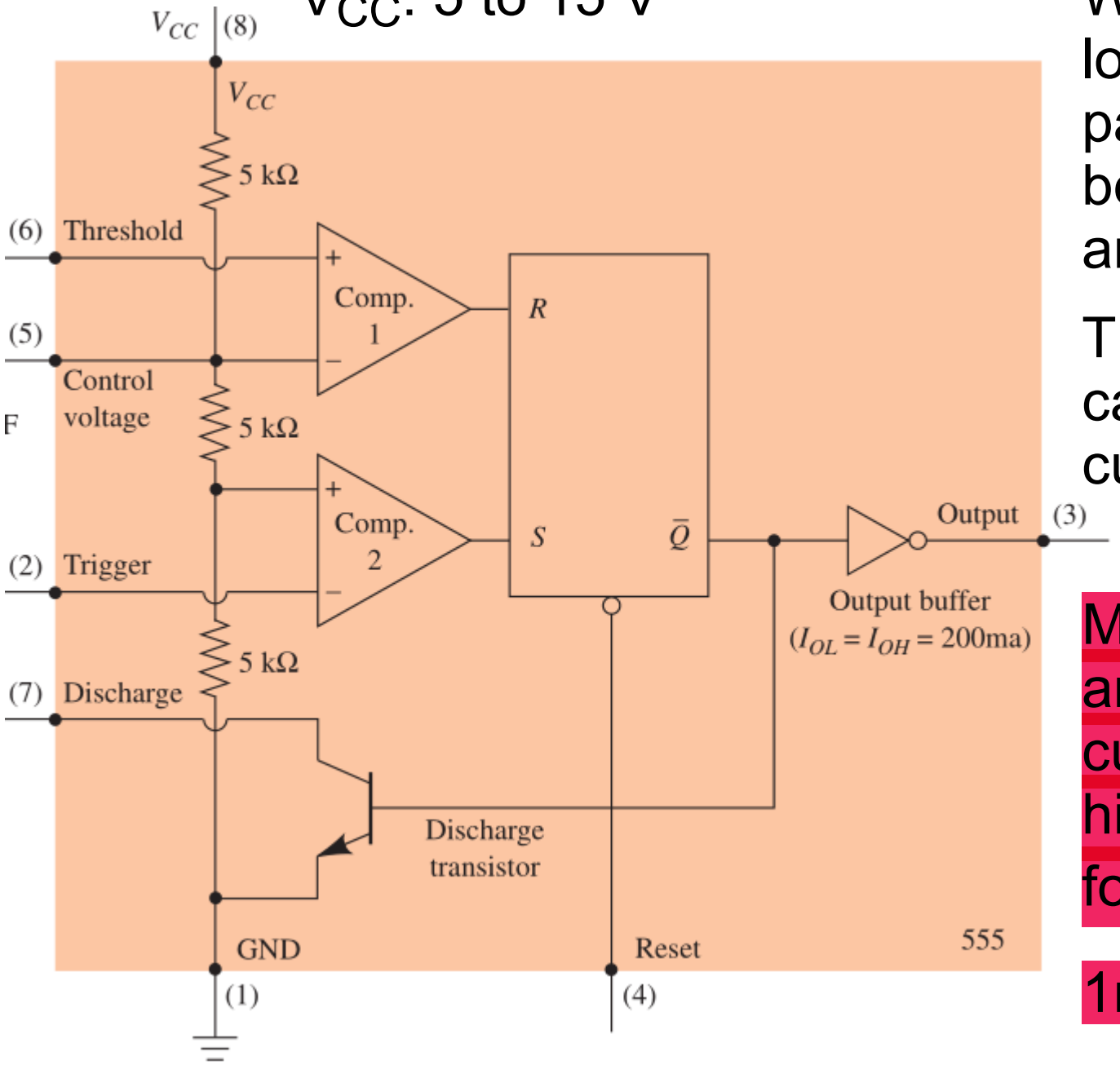


If the trigger input is above the trigger level and the threshold input is above the threshold level, the latch is reset and the output is low.

The reset input can override all other inputs and can be used to initiate a new timing cycle.

When reset goes low, the latch is reset, and the output is low.

V_{CC}: 5 to 15 V



When the output is low, a low-impedance path is provided between discharge and ground.

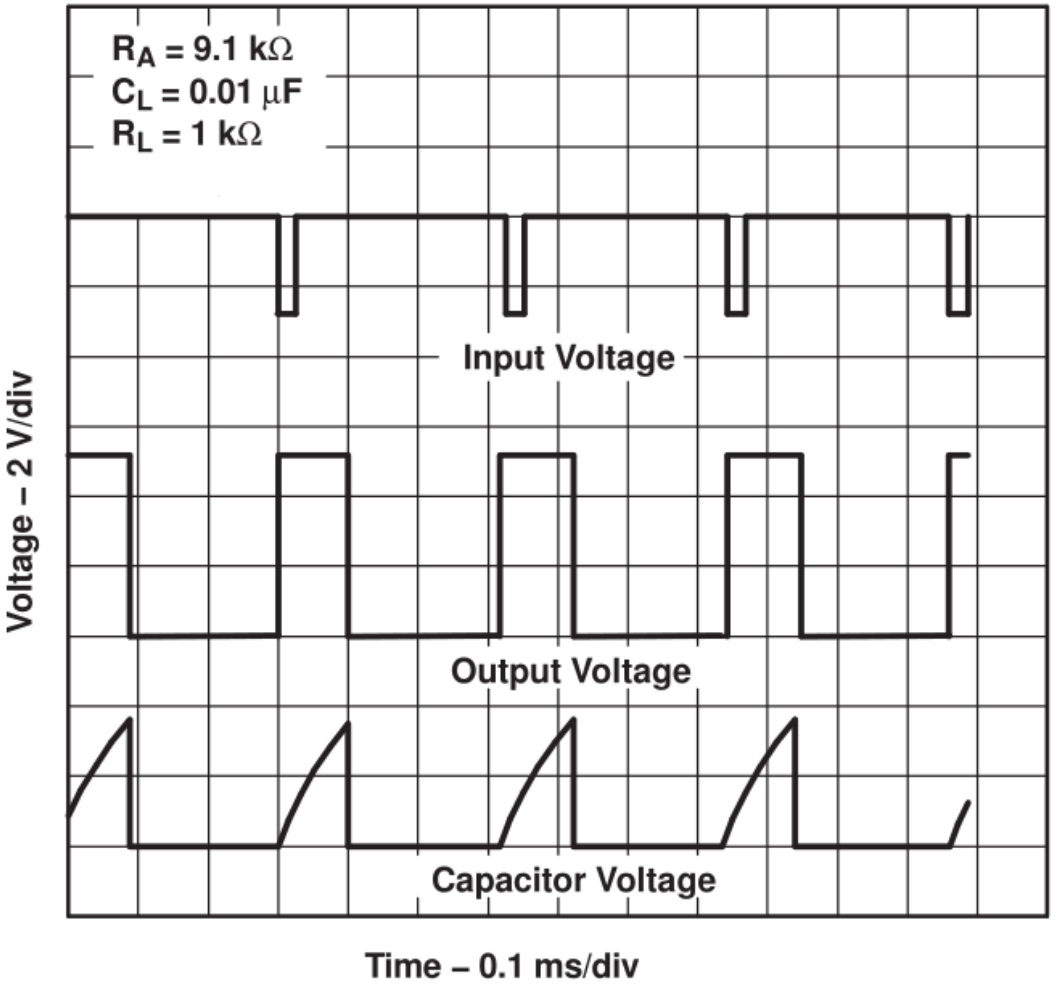
The output circuit can sink or source current up to 200mA

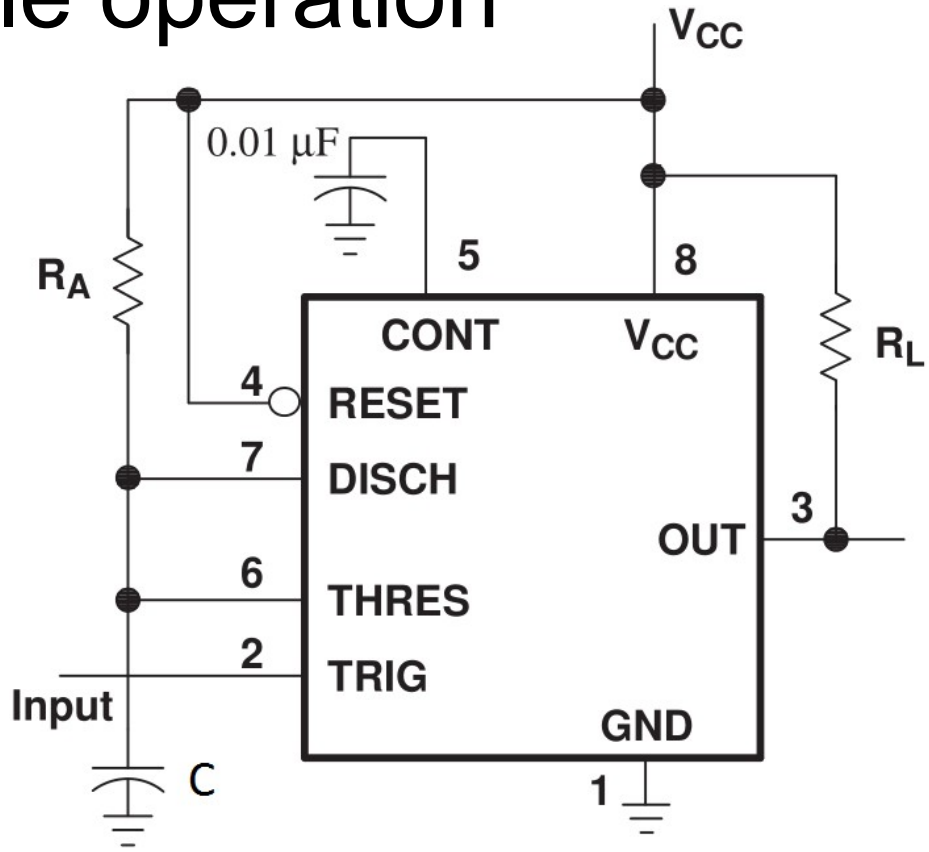
Maximum output sink and discharge sink current is greater for higher V_{CC} and less for lower V_{CC}.

1mHz to 100 kHz

555 circuit for monostable operation

Monostable operation is initiated when TRIG voltage falls below the trigger threshold





The output pulse duration:

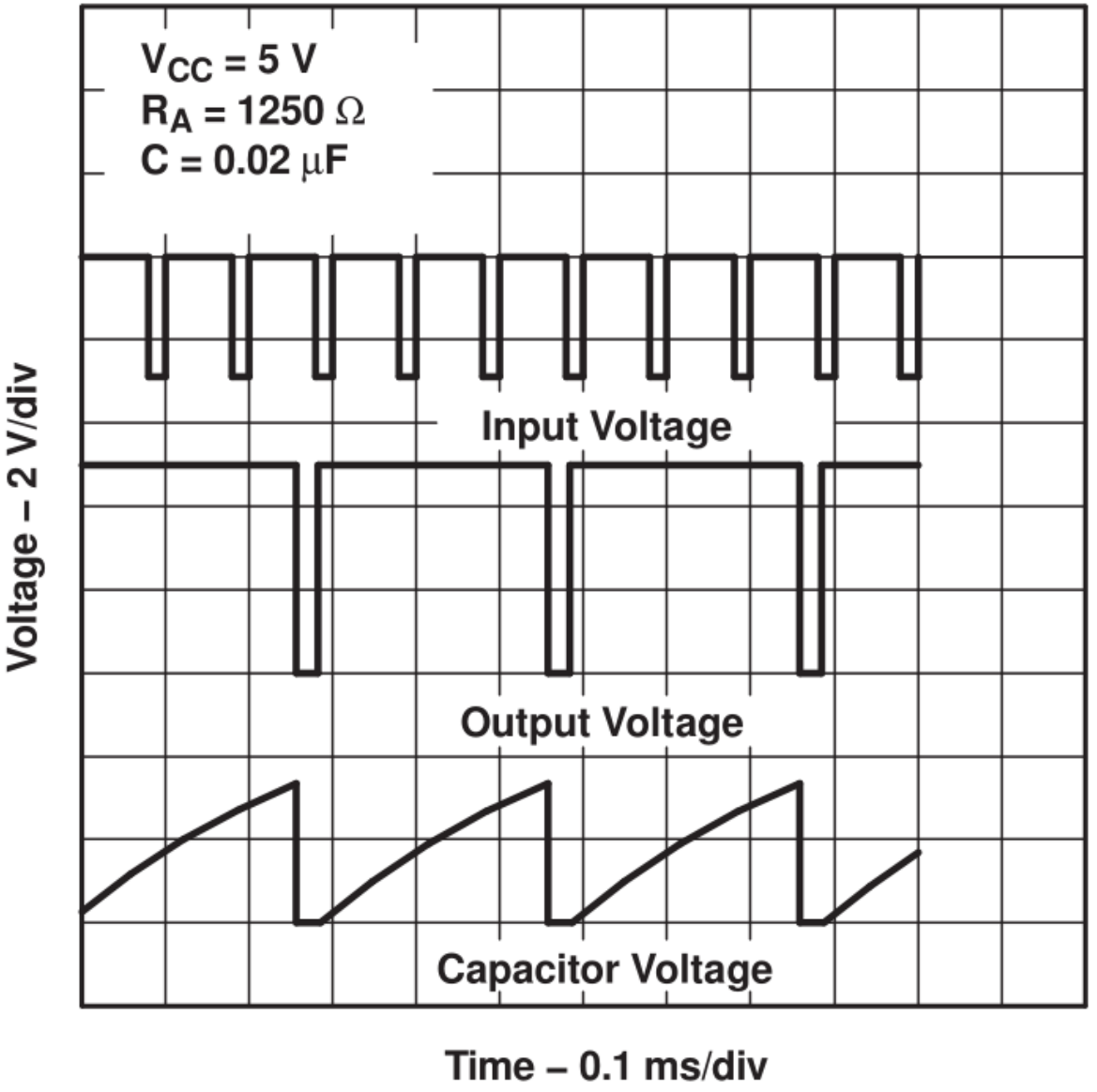
$$t_w = R_A C \ln\left(\frac{1}{1-\frac{2}{3}V_{CC}/V_{CC}}\right)$$

$$t_w \approx 1.1 R_A C$$

Frequency divider

The monostable circuit can be used as a frequency divider.

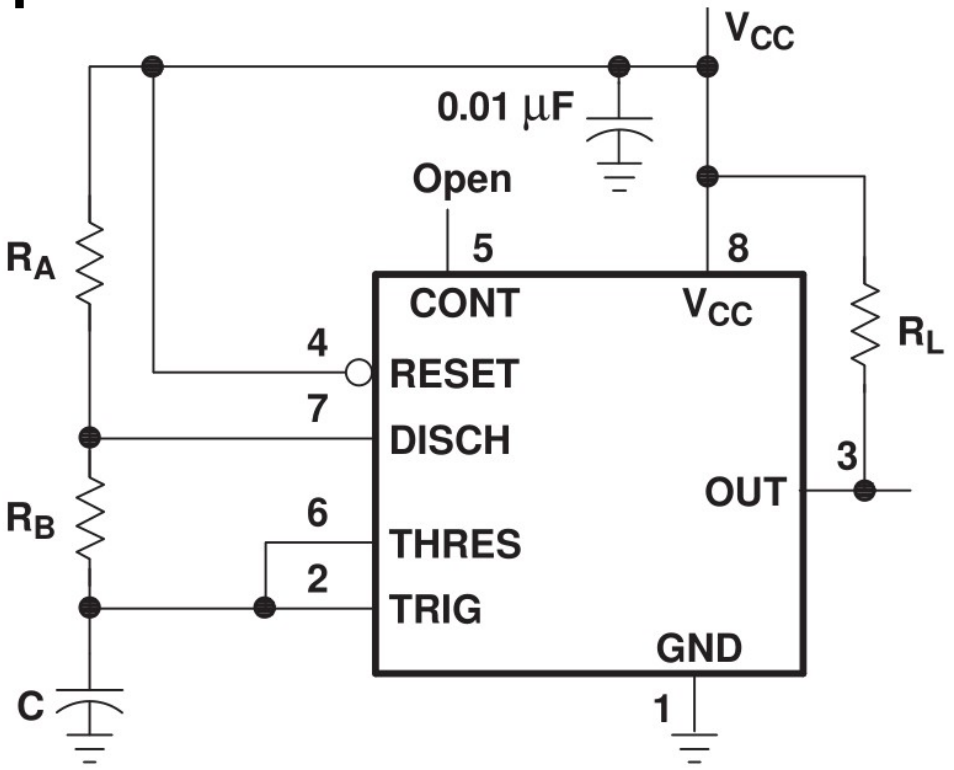
Example: a divide-by-three circuit that makes use of the fact that re-triggering cannot occur during the timing cycle.



555 circuit for astable operation

Adding a second resistor, R_B , and connecting the trigger input to the threshold input causes the timer to self-trigger.

The capacitor C charges through R_A and R_B and then discharges through R_B only.



The duty cycle is controlled by the values of R_A and R_B

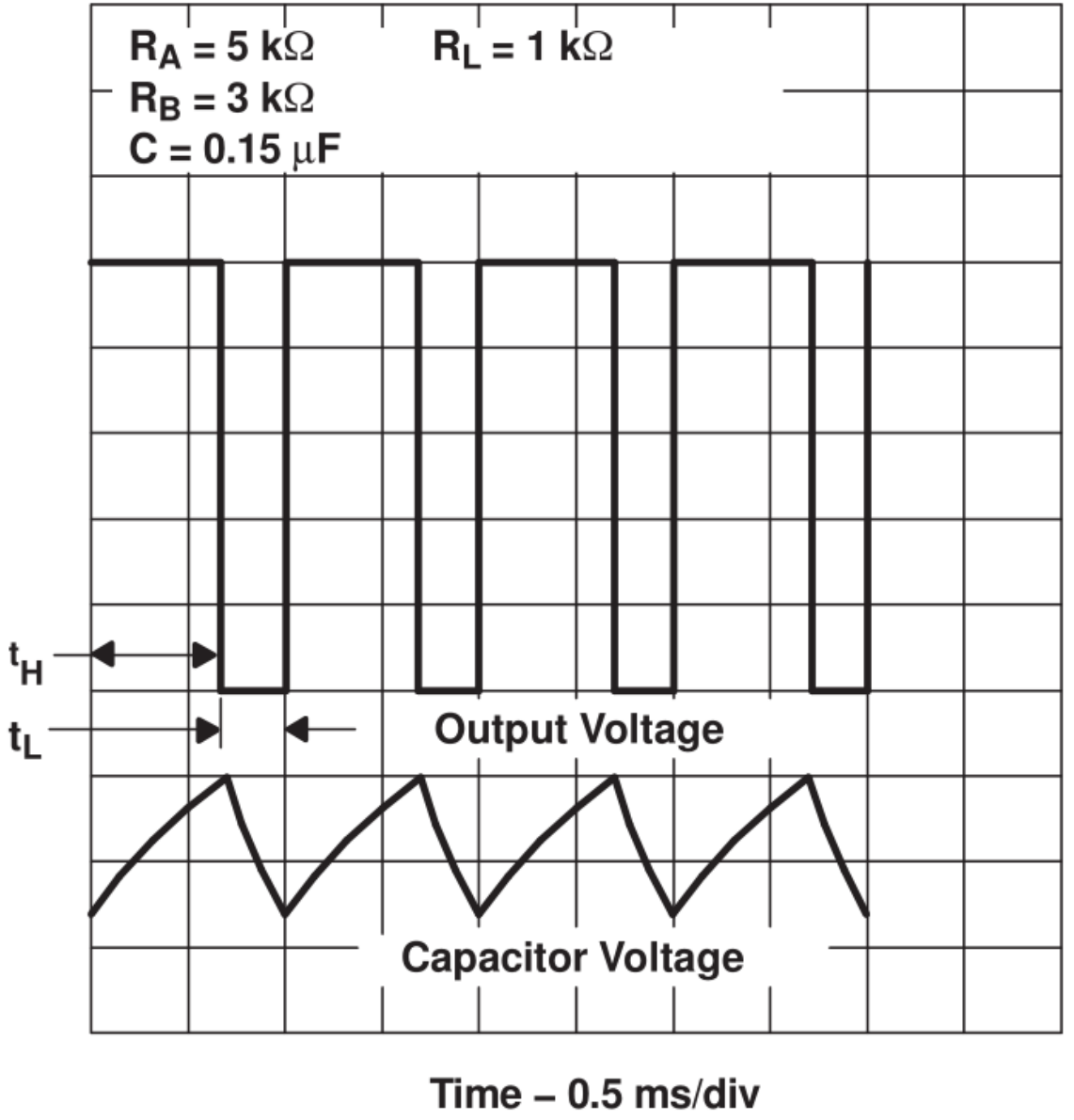
Decoupling CONT voltage to ground with a capacitor can improve operation. This should be evaluated for individual applications.

The astable connection results in capacitor C charging and discharging between the threshold-voltage level and the trigger-voltage level.

$$t_H \approx 0.693\left(R_A + R_B\right)C$$

$$t_L \approx 0.693\, R_B C$$

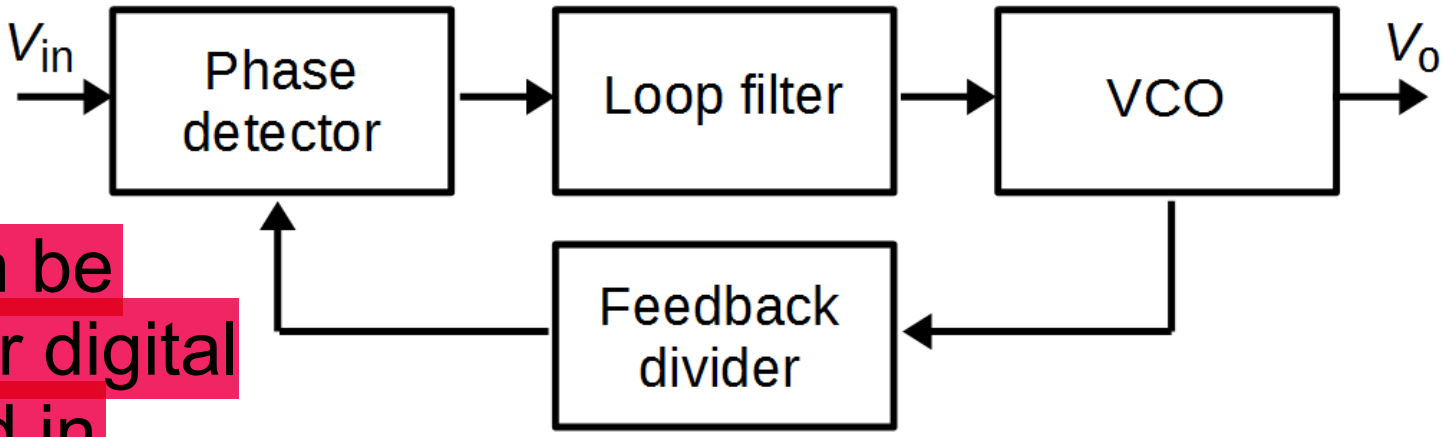
$$0.693 \approx \ln 2 = \ln\left(\frac{1}{1-\frac{\frac{1}{3}V_{CC}}{\frac{2}{3}V_{CC}}}\right)$$



A **phase-locked loop** (**PLL**) synchronizes the output phase and frequency of a controllable oscillator to match the output phase and frequency of a reference oscillator

The simplest PLL consists of four basic functional blocks:

1) Voltage-controlled oscillator (VCO)
2) Phase detector
3) Loop filter
4) Feedback divider



Each block can be either analog or digital or implemented in software.

To maintain synchronization, a phase comparison (by a phase detector) of the outputs of the reference and controllable oscillators generates an error signal that is processed by the loop filter to control the controllable oscillator for minimum phase error.

An increase in phase error produces a control signal that changes the controllable oscillator to decrease the phase error and vice versa.

Consequently, the loop tracks changes in the phase and frequency of the reference oscillator.

Keeping the input and output phase in lock step also implies keeping the input and output frequencies the same.

Consequently, in addition to synchronizing signals, a phase-locked loop can track an input frequency, or it can generate a frequency that is a multiple of the input frequency.

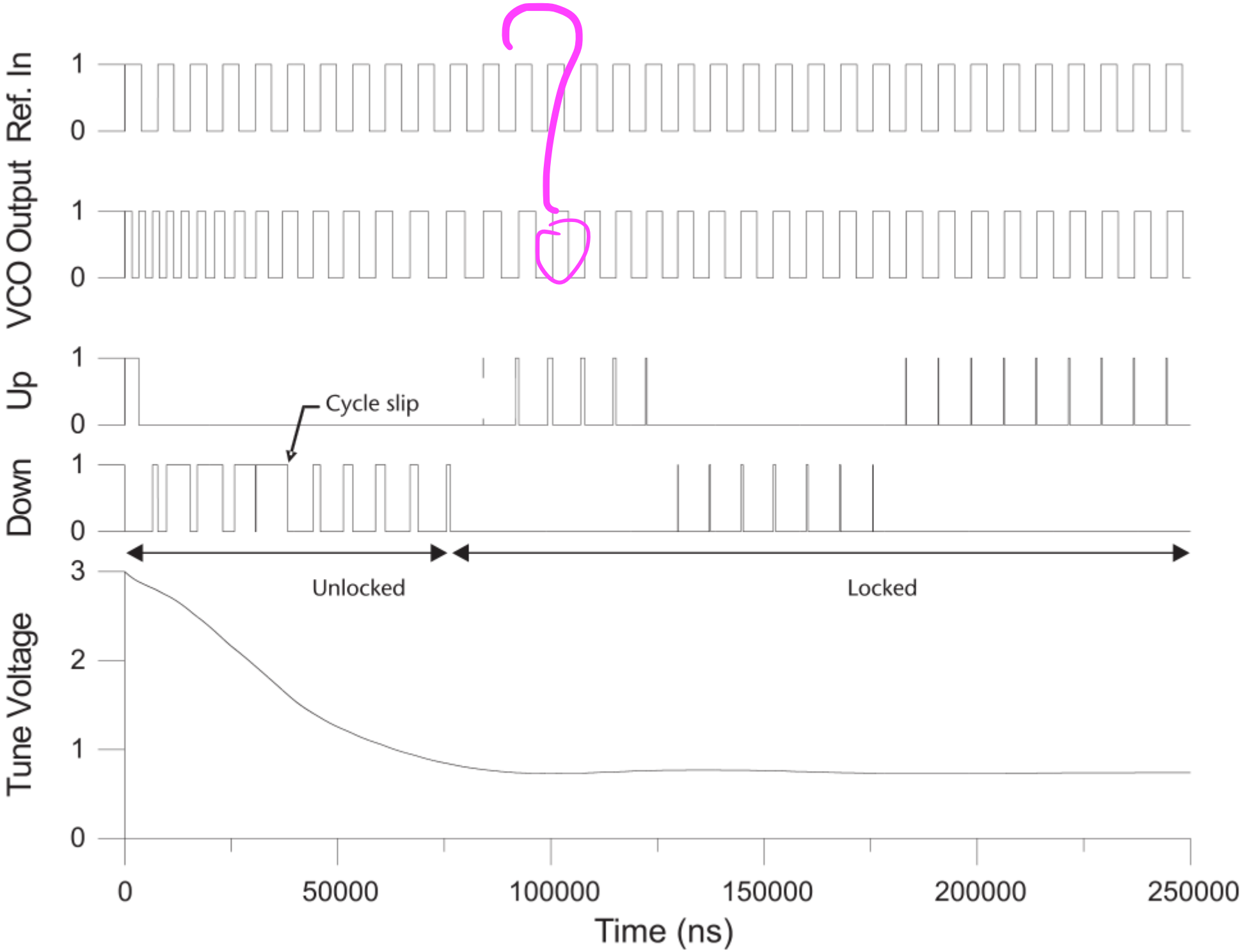
These properties are used for computer clock synchronization, demodulation, and frequency synthesis.

A PLL has several states of operation.

Initially, the loop begins in the **unlocked state**, which occurs at the moment when power is applied to the PLL.

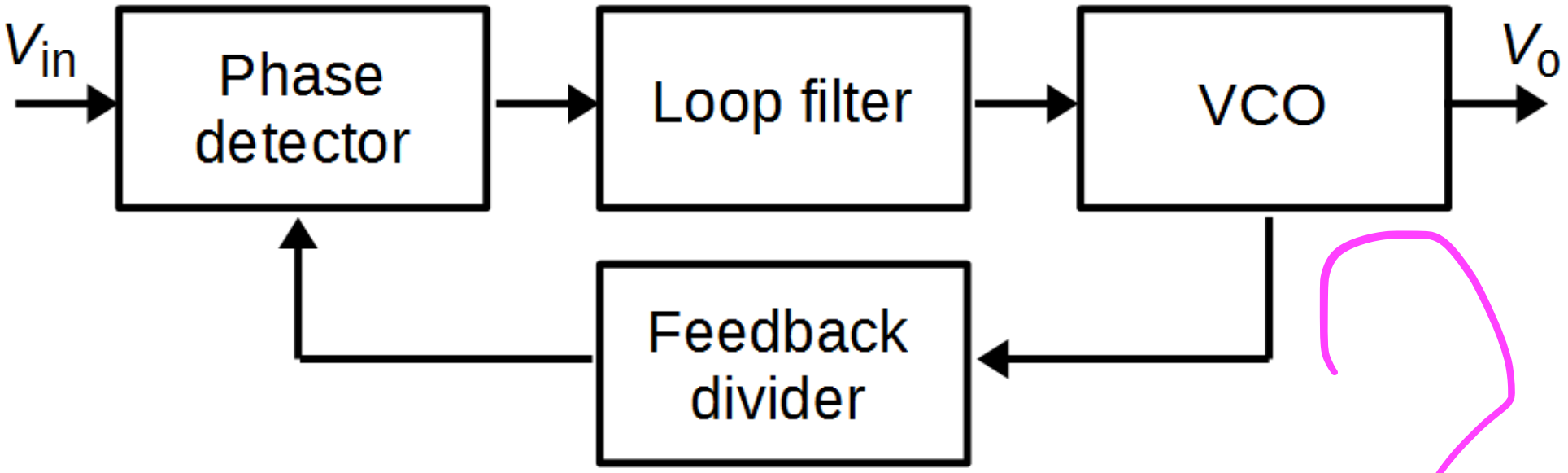
After a **transition period**, a **locked state** arrives, in which the frequency of the VCO equals the average frequency of the input signal, and each input cycle has only one cycle of VCO output.

The transition from the unlocked to the locked state defines the acquisition response for a PLL.


Ref. In
VCO Output
Up
Down
Tune Voltage
1
0
Cycle slip
Unlocked
Locked
3
2
0
50000
100000
150000
200000
250000
Time (ns)

One of the most common applications of a PLL is the multiplication of the reference frequency.

A **frequency divider** placed in the feedback path multiplies the reference frequency by the frequency-divide ratio.



Example: with a frequency divider that has a divide ratio of 2, the frequency of the signal at the output of the VCO will be twice that of the reference.

Applications of PLL

– for synchronization purposes

– to demodulate AM and FM signals

– to synthesize new frequencies which are a multiple of a reference frequency, with the same stability as the reference frequency

– recovery of small signals that otherwise would be lost in noise (lock-in amplifier to track the reference frequency)

– clock multipliers in microprocessors that allow internal processor elements to run faster than external connections, while maintaining precise timing relationships

– for remote control and telecommunications

– atomic force microscopy in tapping mode, to detect changes of the cantilever resonance frequency due to tip–surface interactions

Clock recovery

Some data streams, especially high-speed serial data streams (such as the raw stream of data from the magnetic head of a disk drive), are sent without an accompanying clock.

The receiver generates a clock from an approximate frequency reference, and then phase-aligns to the transitions in the data stream with a PLL.

This process is referred to as **clock recovery**.

Clock generation

Many electronic systems include processors of various sorts that operate at hundreds of megahertz.

Typically, the clocks supplied to these processors come from clock generator PLLs, which multiply a lower-frequency reference clock (usually 50 or 100 MHz) up to the operating frequency of the processor.

The multiplication factor can be quite large in cases where the operating frequency is multiple gigahertz and the reference crystal is just tens or hundreds of megahertz.

Deskewing

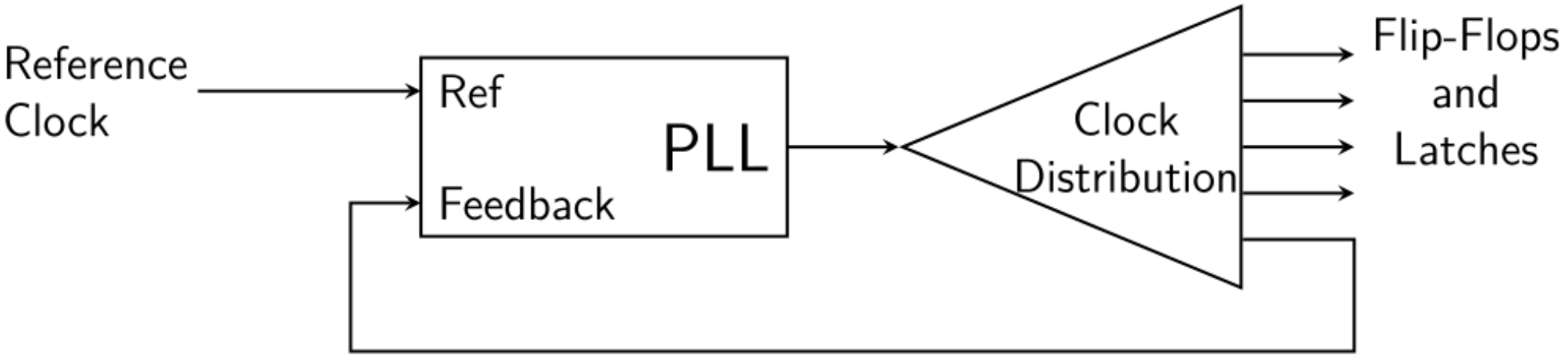
If a clock is sent in parallel with data, that clock can be used to sample the data.

Clock skew limits the frequency at which data can be sent.

One way of eliminating this clock skew is to include a deskew PLL on the receive side, so that the clock at each data flip-flop is phase-matched to the received clock.

In that type of application, a special form of a PLL called a delay-locked loop (DLL) is frequently used.

Typically, the reference clock enters the chip and drives a phase locked loop (PLL), which then drives the system's clock distribution.



The clock distribution is usually balanced so that the clock arrives at every endpoint simultaneously.

One of those endpoints is the PLL's feedback input.

The function of the PLL is to compare the distributed clock to the incoming reference clock, and vary the phase and frequency of its output until the reference and feedback clocks are phase and frequency matched.

Nyquist theorem

Also called **sampling theorem**

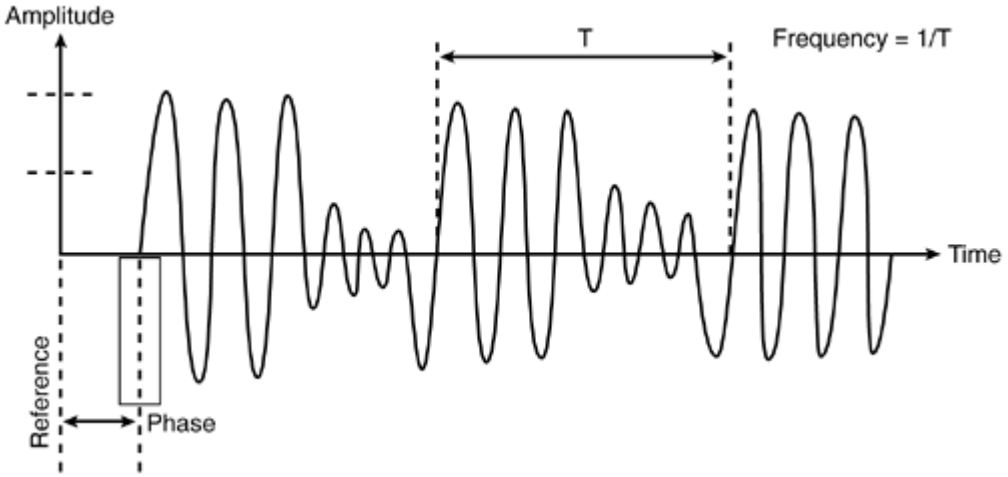
For correct reproduction of the sampled signal the sampling frequency has to be more than twice the highest signal frequency

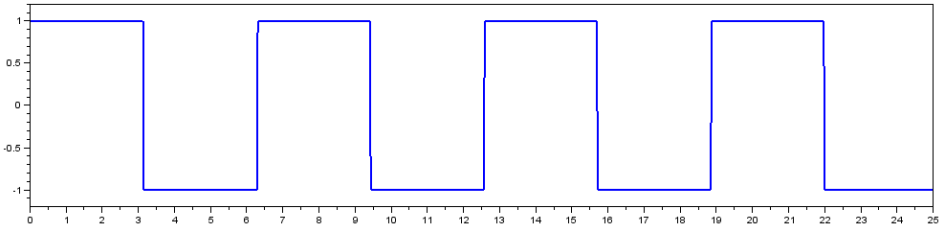
It is applicable only to uniform sampling.

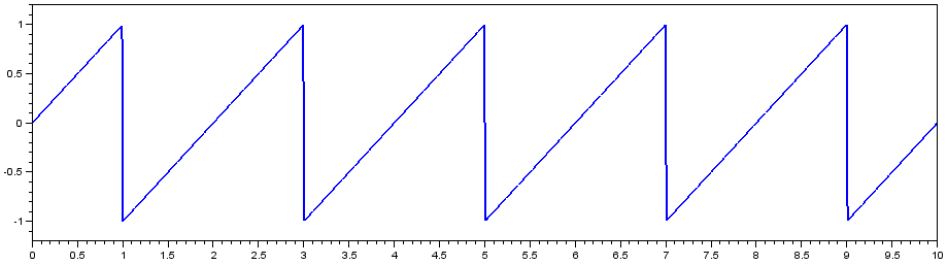
Direct digital synthesis (DDS)

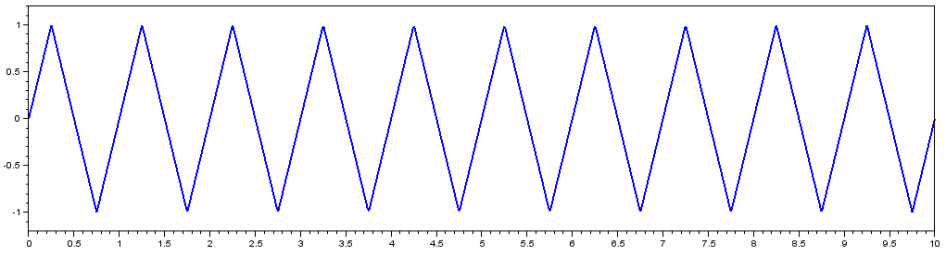
Direct digital synthesis (DDS) is a technique for using digital data processing blocks as a means to generate a frequency- and phase-tunable output signal referenced to a fixed-frequency precision clock source.

The waveform may be a sinewave, a saw-tooth wave, a triangle wave, a square wave, or any periodic waveform.









Advantages of DDS

– The tuning resolution can be made arbitrarily small

– The phase and the frequency of the waveform can be controlled in one sample period, making phase modulation feasible

– The DDS implementation relies upon integer arithmetic, allowing implementation on virtually any microcontroller or FPGA

– The DDS implementation is always stable, even with finite-length control words – no need for an automatic gain control

– The phase continuity is preserved whenever the frequency is changed – a valuable tool for tunable waveform generators

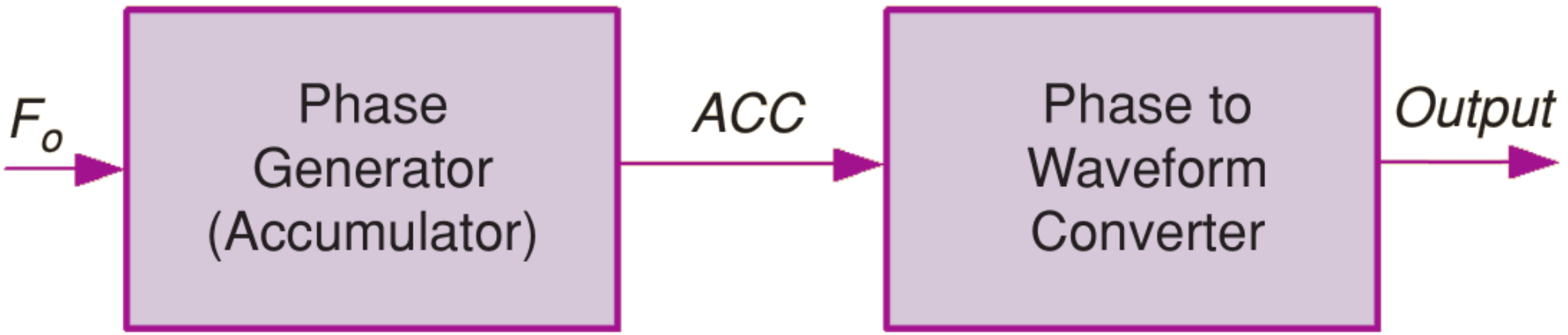
The implementation of DDS is divided into two parts:

1) a discrete-time phase generator (the accumulator)

– outputs a phase value ACC

2) a phase to waveform converter

– outputs the desired DDS signal



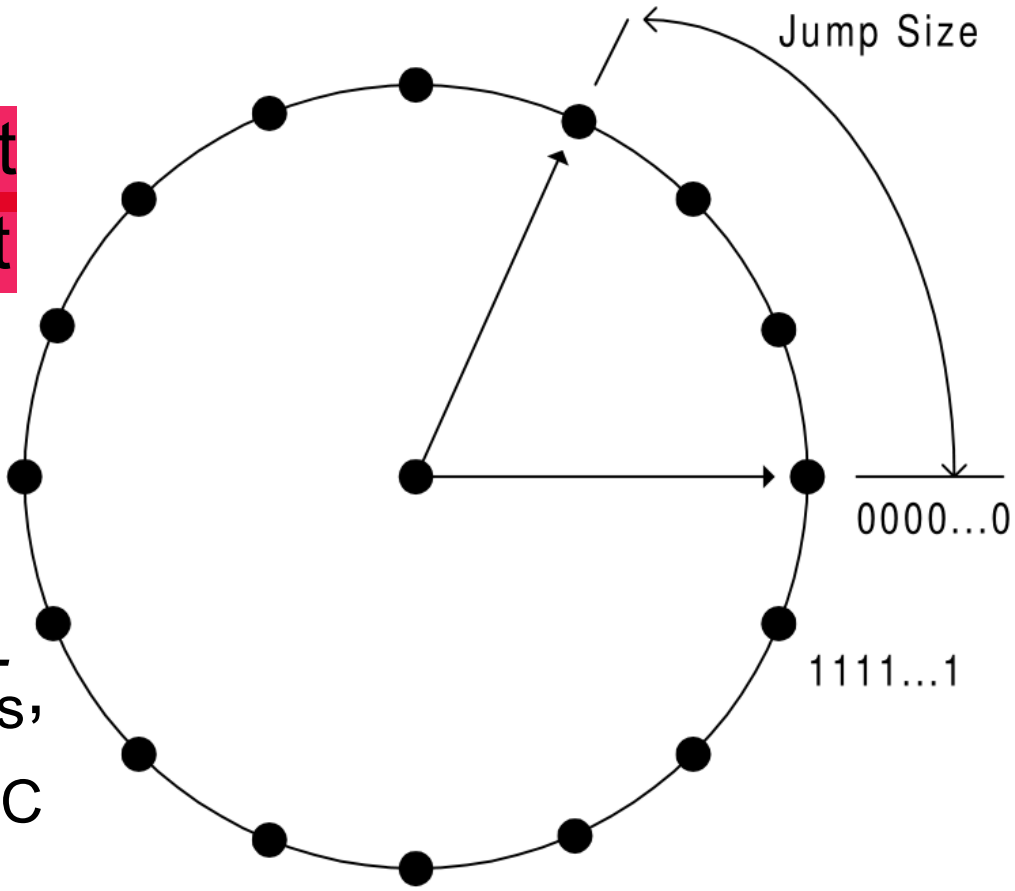
F_o – the frequency of the output waveform

The size of the accumulator (or word length) is N bits.

Assuming that the period of the output signal is 2π radians, the maximum phase is represented by the integer number 2^N

Δ_{ACC} – the phase increment related to the desired output F_o frequency.

Δ_{ACC} is an integer



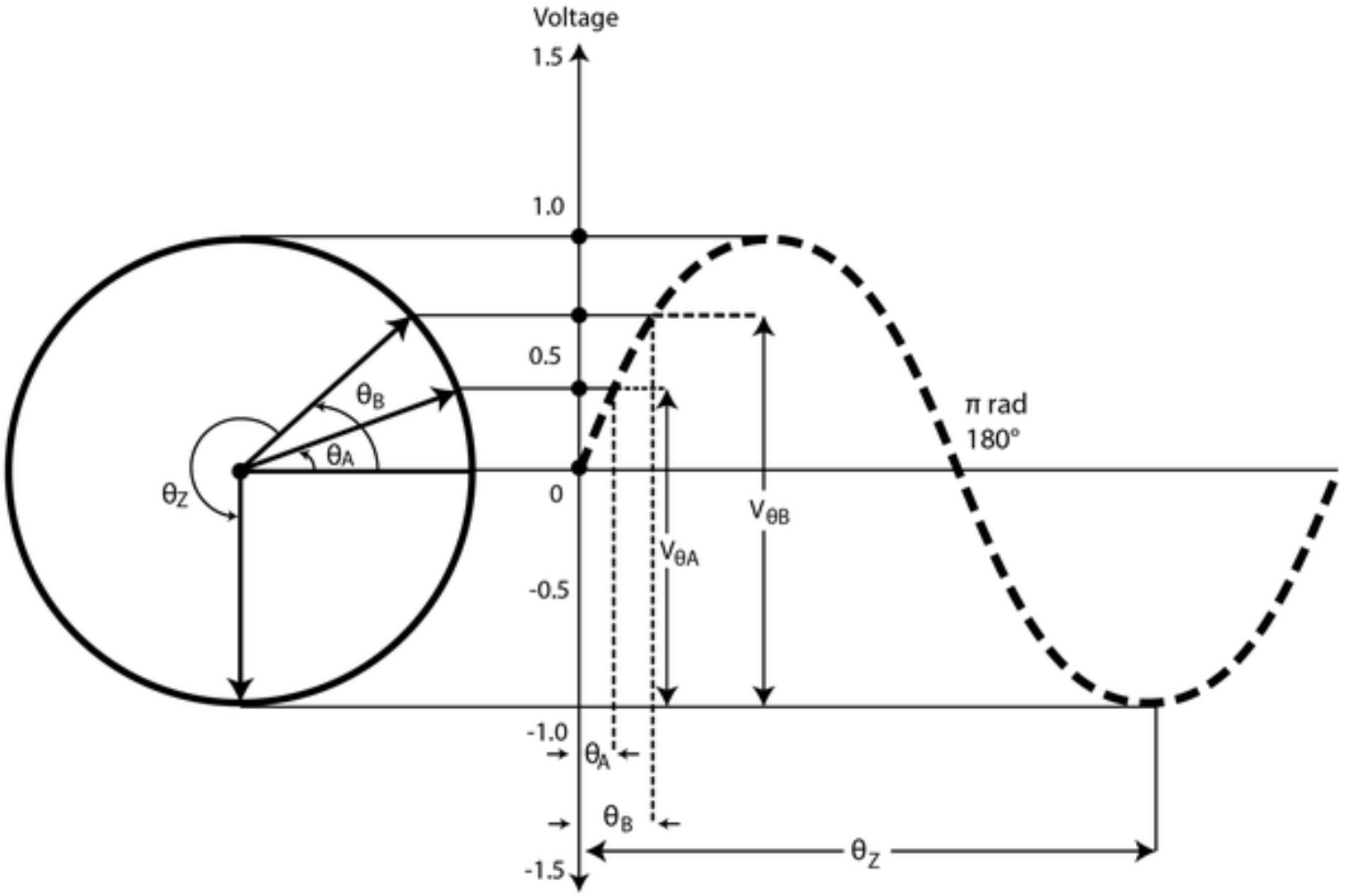
During one sample period T_s, the phase increases by Δ_{ACC}

It takes T_o to reach the maximum phase 2^N:

$$T_o = \frac{1}{F_o} = \frac{2^N T_s}{\Delta_{ACC}}$$

As the vector rotates around the wheel, visualize that a corresponding output waveform is being generated.

One revolution of the vector around the phase wheel, at a constant speed, results in one complete cycle of the output waveform.



$$T_o = \frac{1}{F_o} = \frac{2^N T_s}{\Delta_{ACC}}$$

Rewrite it in terms of frequency: $F_o = F_o(\Delta_{ACC}) = \frac{F_s}{2^N}\Delta_{ACC}$

The phase increment Δ_{ACC}, rounded to the nearest integer, is: $\Delta_{ACC} = \left\lfloor F_o \frac{2^N}{F_s} + 0.5 \right\rfloor$

The tuning step, $\Delta f_{o,min}$, is the smallest step in frequency, that the DDS can achieve

$$\Delta F_{o,min} = F_o(\Delta_{ACC} + 1) - F_o(\Delta_{ACC}) = \frac{F_s}{2^N}$$

This equation allows the designer to choose the number of bits, *N*, of the accumulator ACC.

This number *N* is often referred to as the frequency tuning word length

$$N = \left\lfloor \log_2\left(\frac{F_s}{\Delta F_{o,min}}\right) + 0.5 \right\rfloor$$

The minimum frequency $F_{o,min}$ the DDS can generate

$$F_{o,min} = F_o\left(\Delta_{ACC} = 1\right) = \frac{F_s}{2^N}$$

The maximum frequency $F_{o,max}$ the DDS can generate is given by the uniform sampling theorem (Nyquist):

$$F_{o,max} = \frac{F_s}{2}$$

Therefore,

$$\frac{F_s}{2^N} \le F_o \le \frac{F_s}{2}$$

From a phase to a waveform

The phase is coded with N bits in the accumulator.

Thus, the waveform can be defined with up to 2^N phase values.

In case 2^N is too large for a realistic implementation, the phase-to-amplitude converter uses fewer bits than N.

Let us note P as the number of bits used as the phase information (with $P \leq N$).

The output waveform values can be stored in a **lookup table** (**LUT**) with 2^P entries

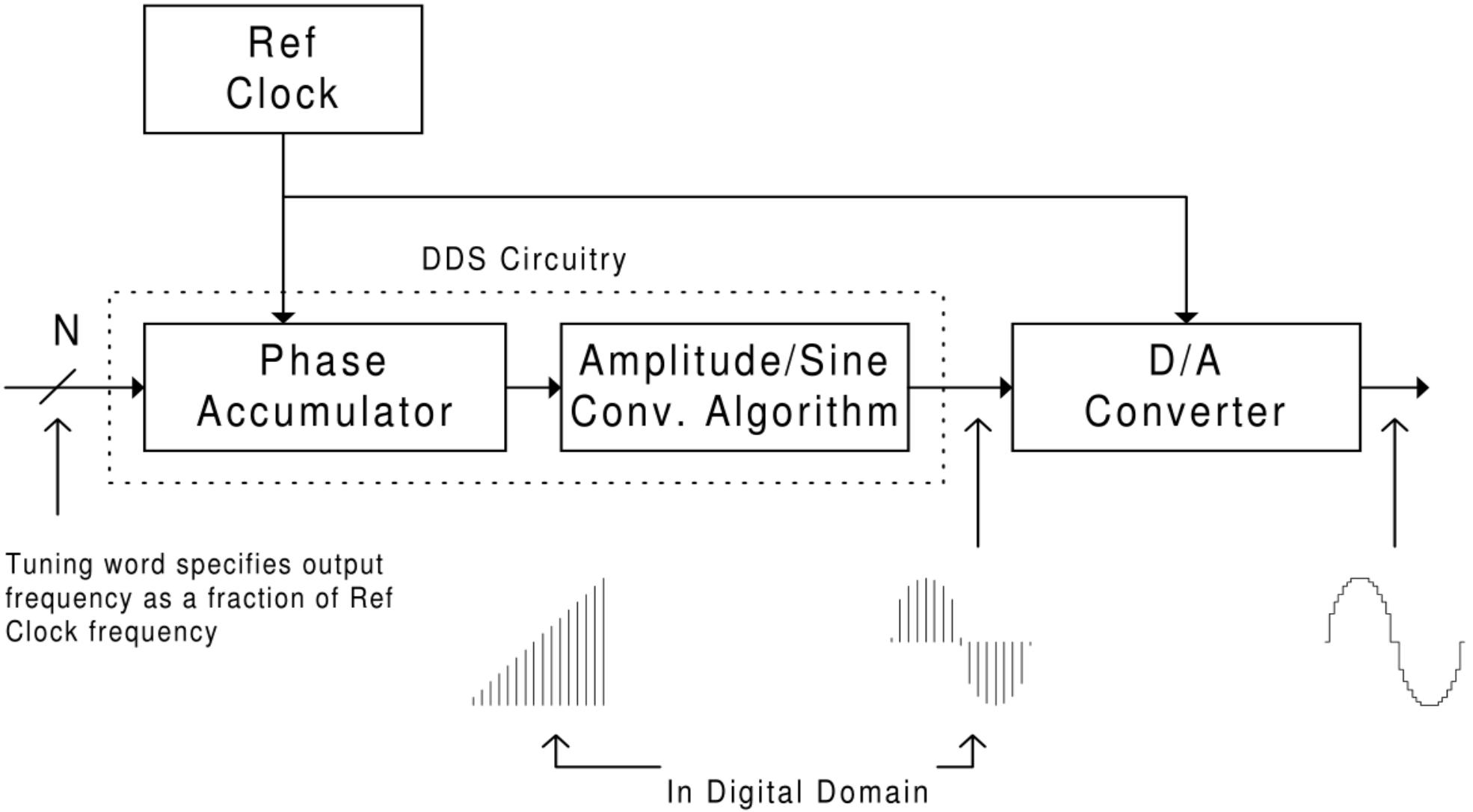
The output value is computed as

Output = LUT (ACC)

The lookup table can be stored in the PROM or a register file.

The address counter steps through and accesses each of the PROM's memory locations and the contents are presented to a D/A converter.

The D/A converter generates an analog waveform in response to the digital input words from the PROM.

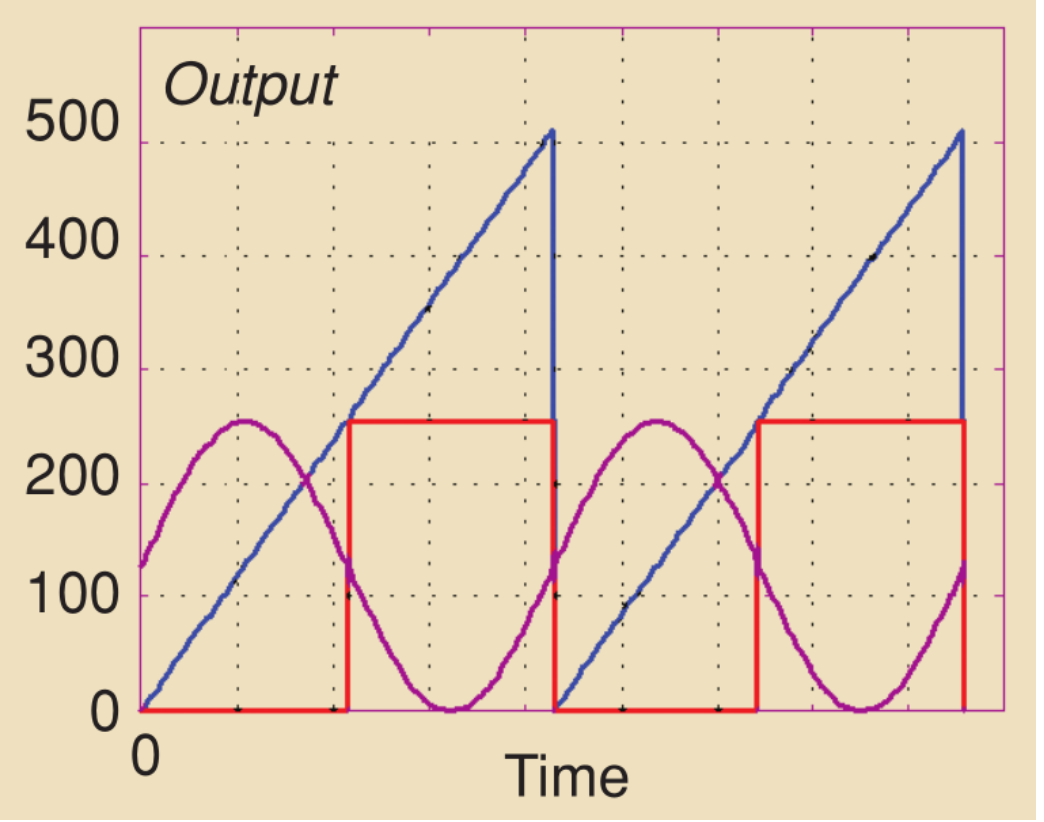

Ref
Clock
DDS Circuitry
N
Phase
Accumulator
Amplitude/Sine
Conv. Algorithm
D/A
Converter
Tuning word specifies output
frequency as a fraction of Ref
Clock frequency
In Digital Domain

DDS can generate a sinewave with an offset *B* and a peak amplitude *A*.

The content of the LUT, containing the DDS output values, is computed using

$$LUT(i) = \left\lfloor B + A \cdot \sin\left(\frac{2\pi i}{2^P}\right) + 0.5 \right\rfloor, \quad 0 \le i \le 2^P - 1$$

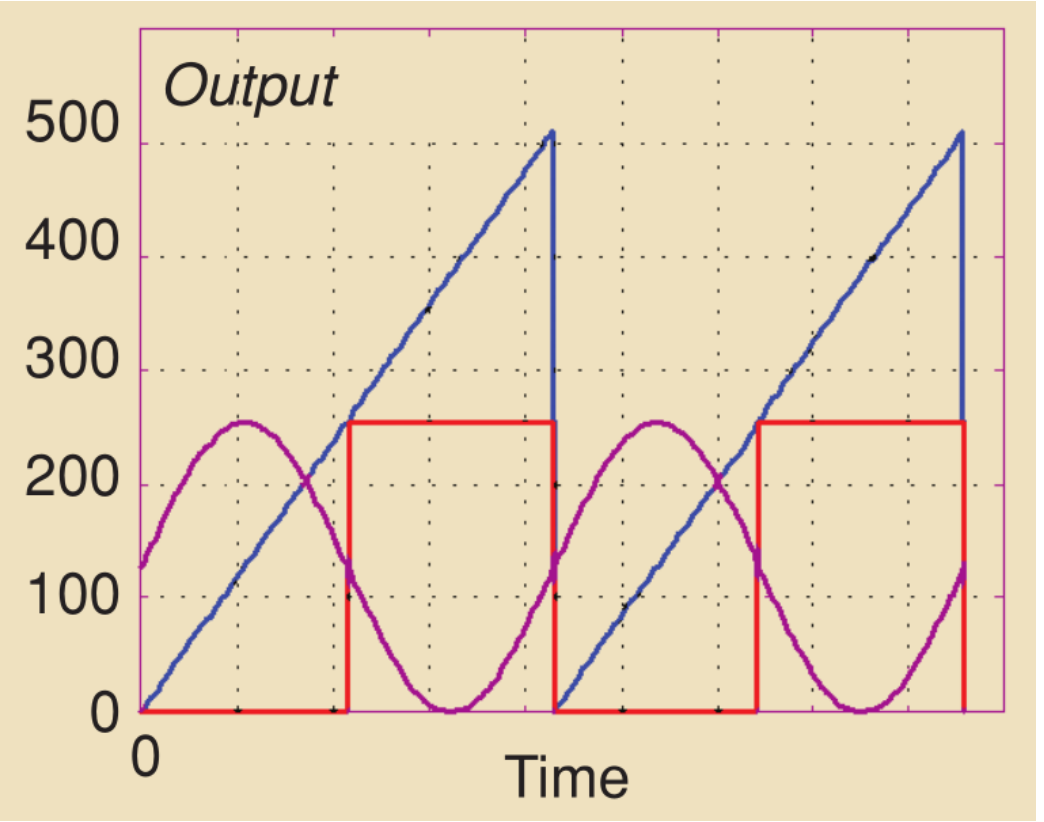
sine for $P = 9$, $A = 127.5$, $B = 127.5$,
the output waveform for
$F_s = 44{,}100$ Hz and
$F_o = 233$ Hz



Most DDS architectures exploit the symmetrical nature of a sinewave and utilize mapping logic to synthesize a complete sinewave cycle from ¼ cycle of data from the phase accumulator.

A square wave can be had with no computational overhead because that waveform is already available as the most significant bit of the phase accumulator ACC.

The most significant bit toggles every π radians, since the accumulator represents 2π radians.



This square wave is corrupted by **phase jitter** of one sampling period T_s.

This phase jitter is caused by the sampling scheme used to synthesize the waveform.

The output of the direct digital synthesizer can occur only at a clock edge.

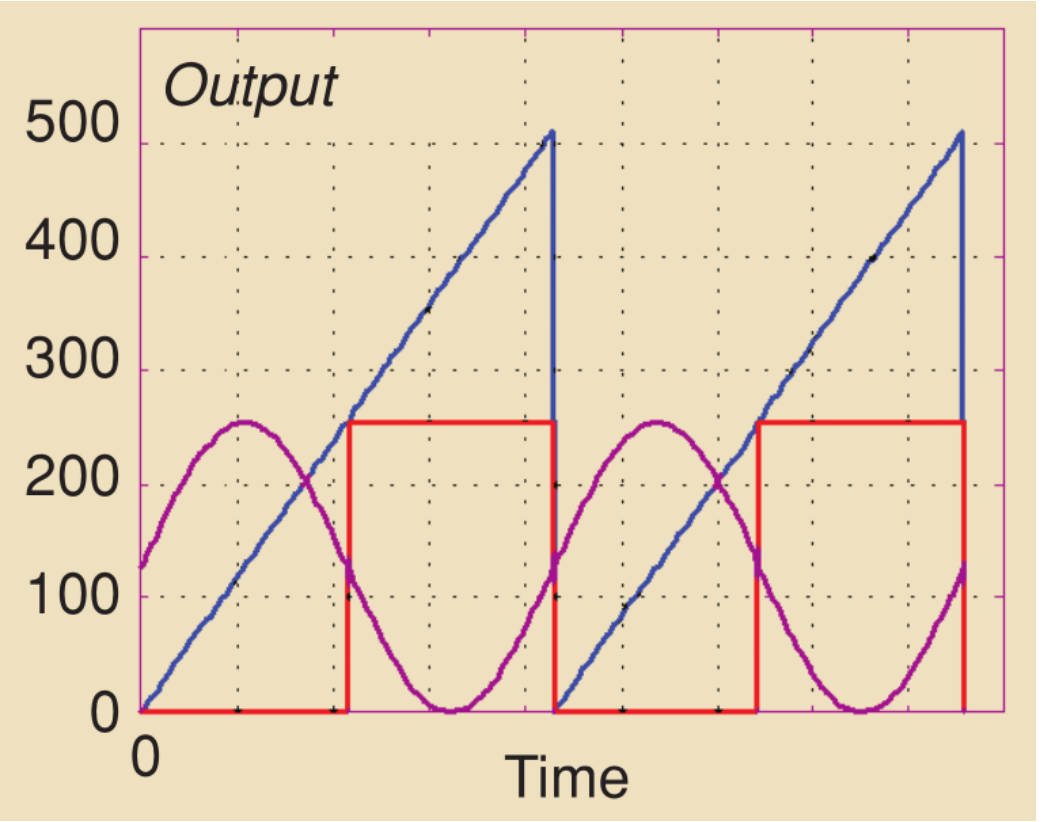
If the output frequency is not a direct submultiple of the clock, a phase error between the ideal output and the actual output slowly increases (or decreases) until it reaches one clock period, at which time the error returns to zero and starts to increase (or decrease) again.

A sawtooth signal is also available with no computational overhead.

The linearly increasing phase accumulator ACC value is stored modulo 2^N, thus leading to a sawtooth signal

The LUT is not used in this case, or it is the identity function: Output=ACC.

With the use of logic gates, a triangular output waveform can be generated from the sawtooth.



Phase quantization

Phase quantization occurs when the phase information ACC is truncated from N to P bits.

The reason behind this quantization is to keep the memory requirements of the phase to waveform converter low.

When implemented as a LUT, the size of the memory is $2^P \times M$ bits.

M is the word length of the Output amplitude word.

A realistic value for N is 32, but this would lead to a $2^{32} \times M$ memory that is not realistic.

Thus we quantize the phase information to P bits, as it decreases the number of entries of the LUT.

The phase quantization introduces noise on the phase signal.

It leads to **phase noise** and it produces unwanted spurious spectral components in the DDS output signals, called **spurs**.

The difference between the **carrier level** (which is the desired signal) and the maximum level of spurs is called **spurious free dynamic range** (**SFDR**).

Amplitude quantization

The output of the phase-to-waveform converter is quantized to *M* bits, with *M* being the word length of the Output amplitude word.

This adds spurious high-frequency components.