The Abdus Salam
International Centre for Theoretical Physics

United Nations
Educational, Scientific and
Cultural Organization

IAEA
International Atomic Energy Agency

2025-3

Satellite Navigation Science and Technology for Africa

23 March - 9 April, 2009

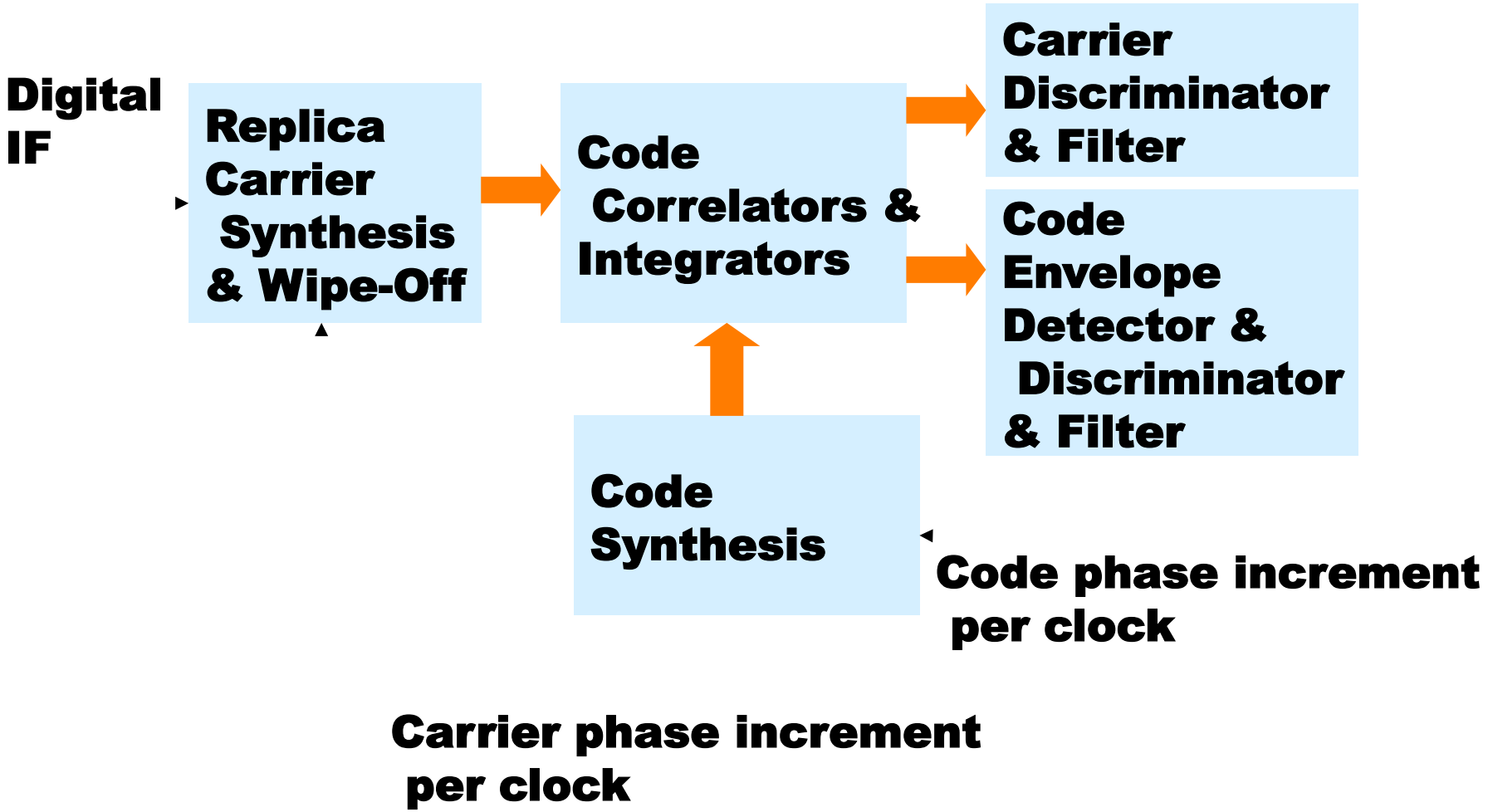
Introduction to GPS Receiver Design Principles (Part 2)

Ward Phillip W.
Navward
U.S.A.

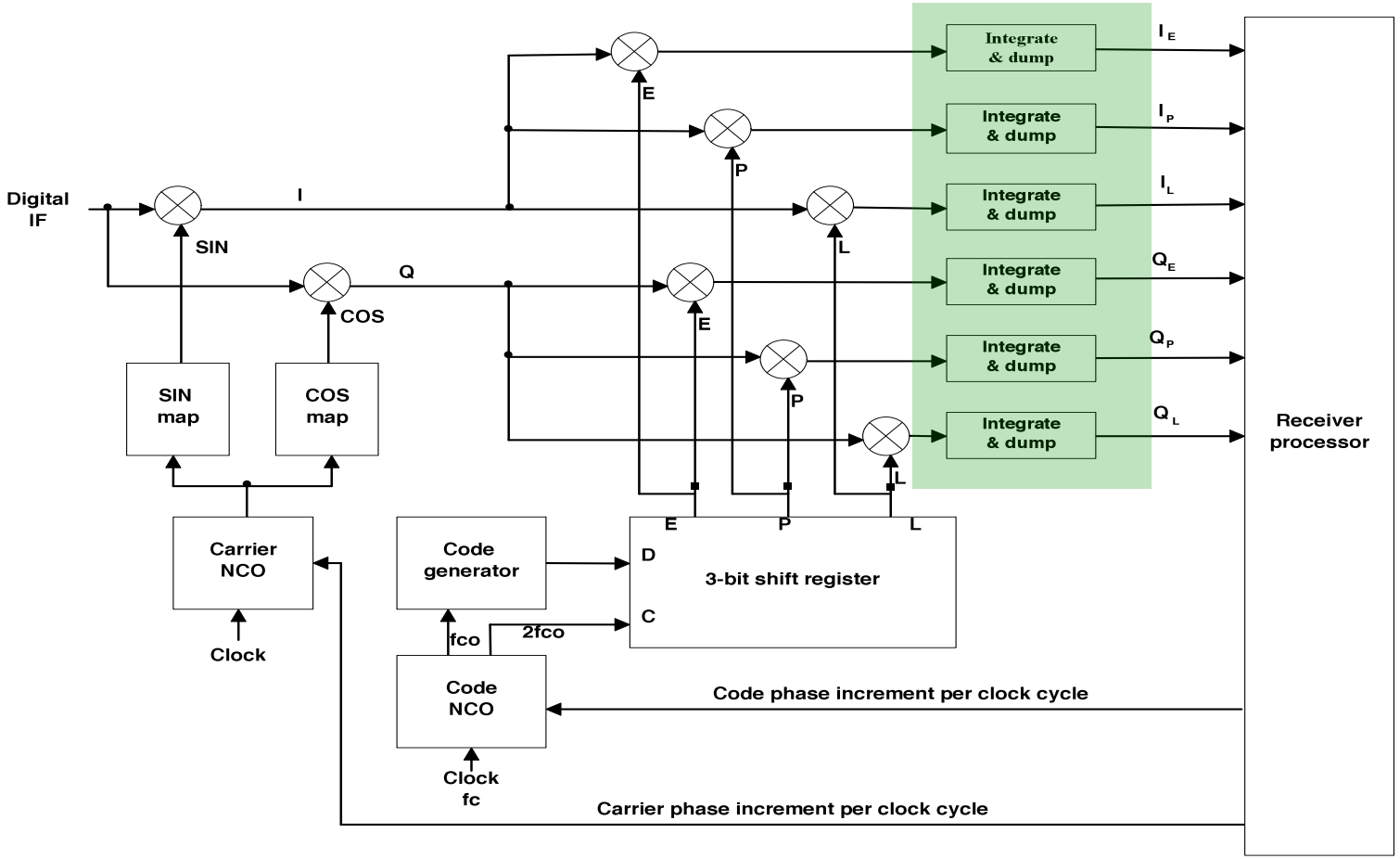
Session II
GPS receiver baseband processes

- Baseband signal processes block diagram
- Predetection integration: dealing with GPS data transition boundaries
- Carrier aiding of code loop: scale factors for carrier aided code
- External aiding: where and how it is injected in each tracking loop
- Digital frequency synthesizer block diagram and output waveforms
- Digital frequency synthesizer design

Baseband signal processes block diagram



Digital receiver channel block diagram – predetection integration



Predetection integration

Predetection is the signal processing after the IF signal has been converted to baseband by the carrier and code stripping processes, but prior to being passed through a signal discriminator; i.e., prior to the non-linear signal detection process. Extensive digital predetection integration and dump processes occur after the carrier and code stripping processes. This causes very large numbers to accumulate even though the IF A/D conversion process is typically with only one to three bits of quantization resolution.

The generic digital receiver channel block diagram shows three complex correlators required to produce three in-phase components which are integrated and dumped to produce I_E, I_P, I_L and three quadra-phase components integrated and dumped to produce Q_E, Q_P, Q_L. The carrier wipe-off and code wipe-off processes must be performed at the digital IF sample rate which is around 5 MHz for C/A-code and 50 MHz for P(Y)-code. The integrate and dump accumulators provide filtering and resampling at the processor baseband input rate which is around 200 Hz (which can be higher or lower depending on the desired predetection bandwidth). The 200 Hz rate is well within the interrupt servicing rate of modern high-speed microprocessors, but the 5 or 50 MHz rates would not be manageable. This further explains why the high speed, but simple processes are implemented in a custom digital ASIC while the low speed, but complex processes are implemented in a microprocessor.

The hardware integrate and dump process in combination with the baseband signal processing integrate and dump process (described below) defines the predetection integration time. The predetection integration time is a compromise design. It must be as long as possible to operate under weak or RF interference signal conditions and it must be as short as possible to operate under high dynamic stress signal conditions.

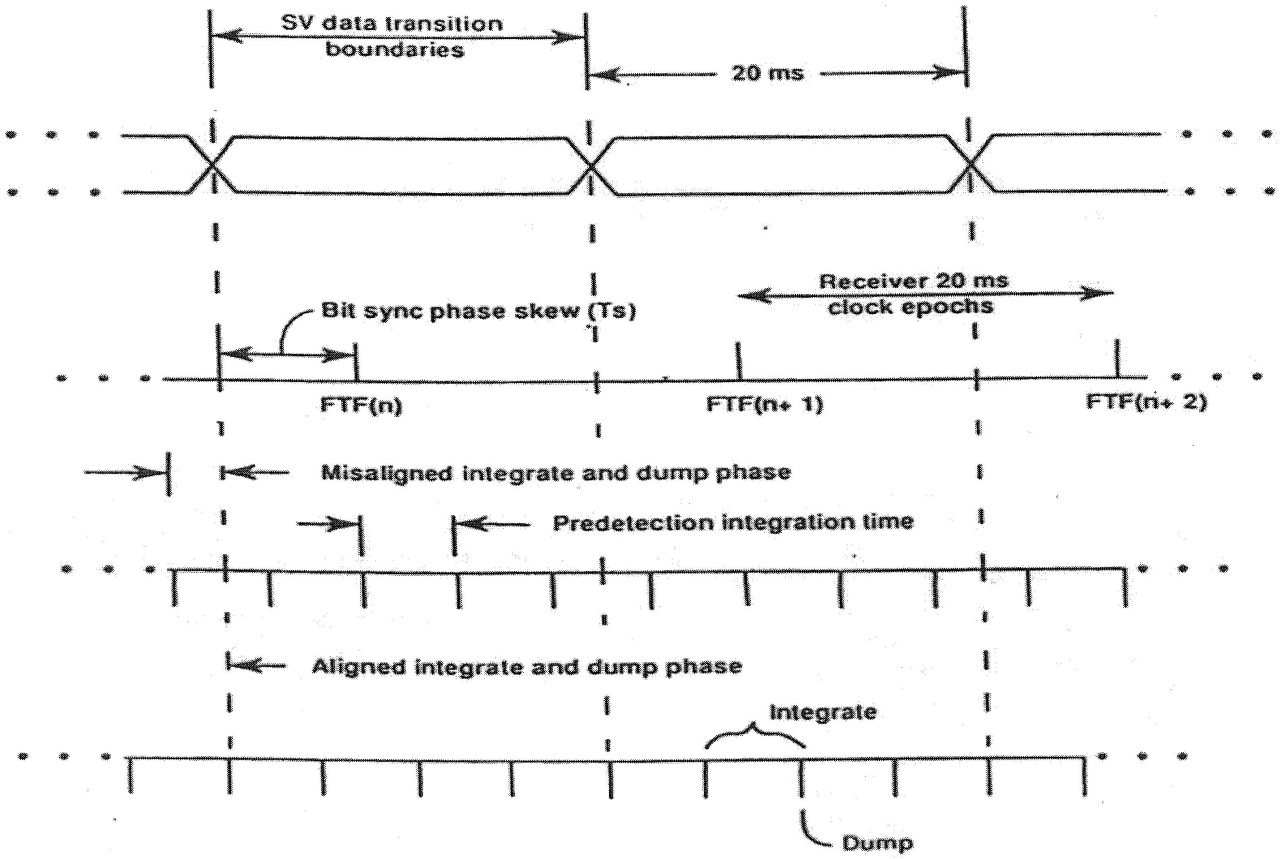
Predetection integration

- Signal integration and dump after carrier and code stripping but prior to signal discriminators
- Can accumulate large numbers with only one to three bits A/D quantization
- Code/carrier wipe-off processes at digital IF sample rates: - 5 MHz for C/A-code and - 50 MHz for P(Y)-code
- Integrate and dump output rates: 50 to 200 Hz
  - These rates well within the interrupt servicing rate of microprocessors

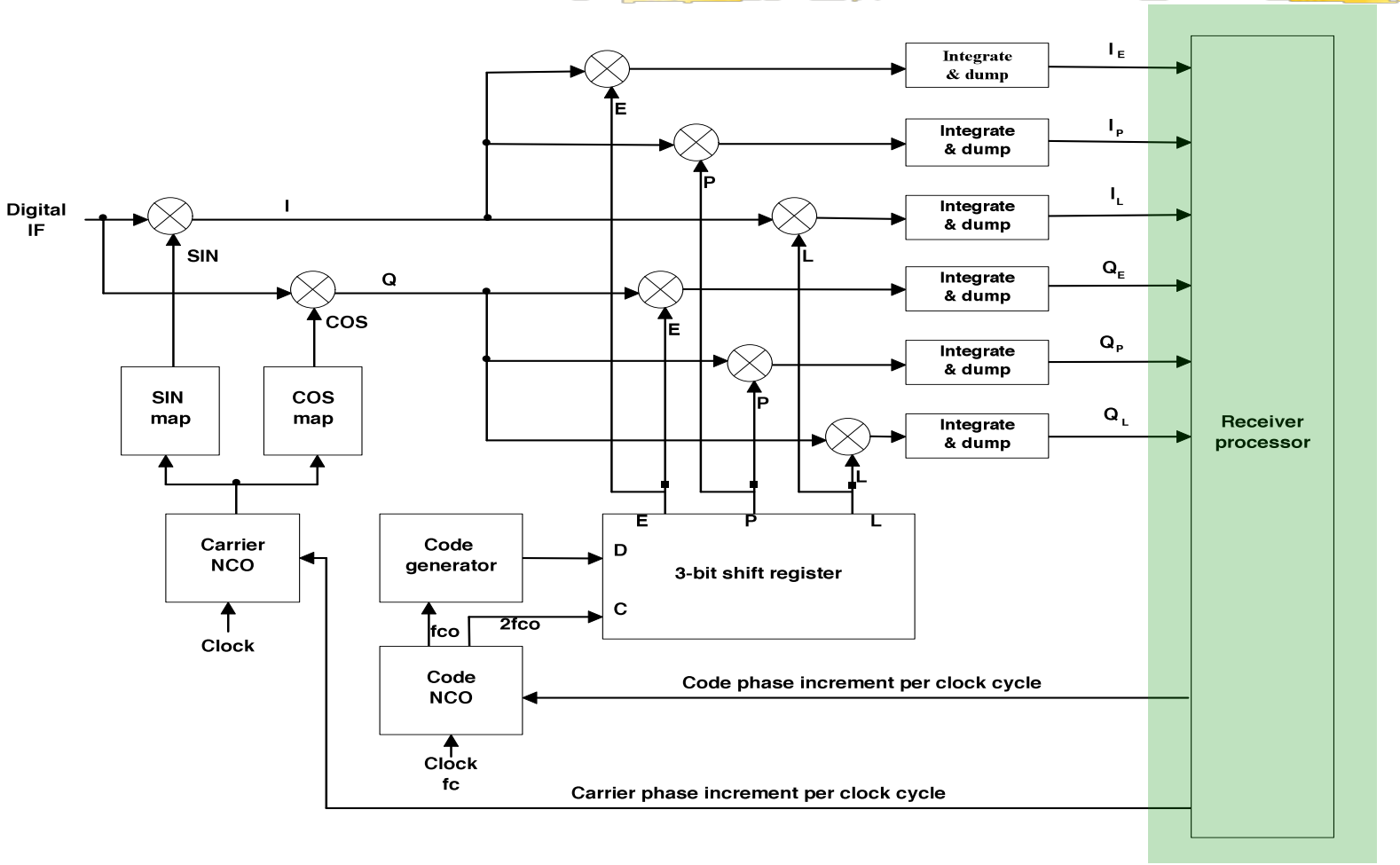
Phase alignment of predetection integrate and dump intervals with SV data transition boundaries

The figure below illustrates the phase alignment needed to prevent the predetection integrate and dump intervals from integrating across an SV data transition boundary. The start and stop boundaries for these integrate and dump functions should not straddle the data bit transition boundaries because each time the SV data bits change sign, the signs of the subsequent integrated I and Q data change. If the boundary is straddled, the result of the predetection integration for that interval will be degraded. Usually during initial searches the receiver does not know where the SV data bit transition boundaries are located. Then, the performance degradation has to be accepted until the bit synchronization process locates them. As shown in the figure below, the SV data transition boundary usually does not align with the receiver's 20 milliseconds clock boundary, which will hereafter be called the fundamental time frame (FTF). The phase offset is shown as "bit sync phase skew," because the SV bit synchronization process initially determines this phase offset. In general, the bit sync phase skew is different for every SV being tracked. The bit sync phase skew changes as the range to the SV changes. The receiver design should accommodate these data bit phase skews if an optimal predetection integration time of 20 ms is to be used during steady state tracking. However, some designers choose to keep the predetection integrate and dump time short (suboptimal at 1to 5 ms), accepting the increased squaring loss and the degradation due to occasional data transition straddle in exchange for the hardware and software design simplifications of ignoring these boundaries

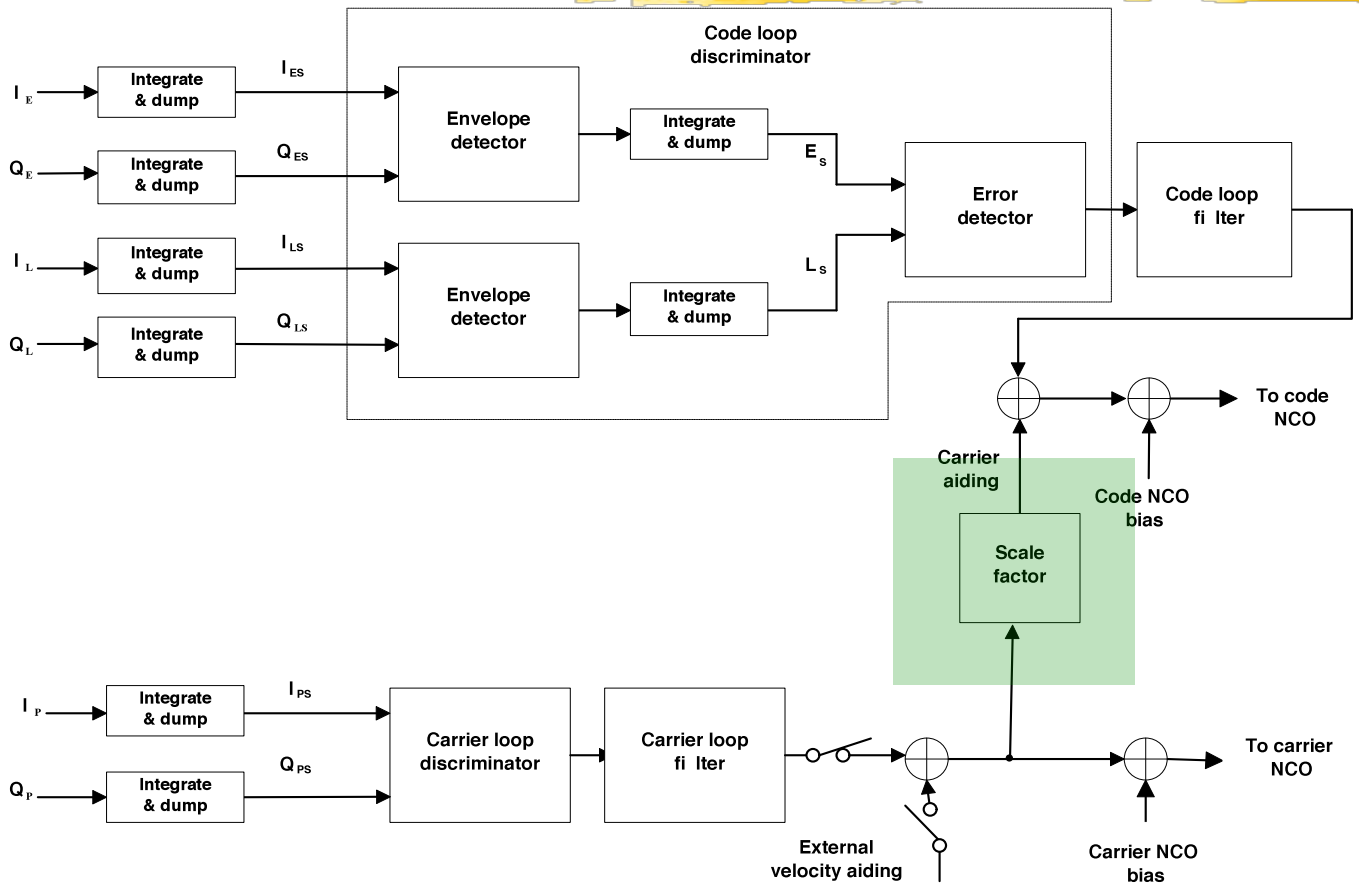
Phase alignment of predetection integrate ad dump intervals with SV data transition boundaries



Digital receiver channel block diagram – receiver processor code/carrier loops



Carrier aiding of code loop: scale factors for carrier aided loop



Carrier aiding of code loop

In the previous figure depicting the baseband processor code and carrier tracking loops, the carrier loop filter output is adjusted by a scale factor and added to the code loop filter output as aiding. This is called a carrier-aided code loop. The scale factor is required because the Doppler effect on the signal is inversely proportional to the wavelength of the signal. Therefore, for the same relative velocity between the SV and the GPS receiver, the Doppler on the code chipping rate is much smaller than the Doppler on the L-band carrier. The scale factor which compensates for this difference in frequency is given by:

$$\text{Scale Factor} = \frac{R_C}{f_L}$$

where: R_c = code chipping rate (chips/s)
= R_0 for P(Y)-code = 10.23×10^6 chips/s
= $R_0/10$ for C/A-code = 1.023×10^6 chips/s
f_L = L-band carrier (Hz)
= $154\ f_0$ for L1 = $154 \times 10.23 \times 10^6$ Hz = 1575.42 MHz
= $120\ f_0$ for L2 = $120 \times 10.23 \times 10^6$ Hz = 1227.60 MHz

Carrier aiding of code loop

- Note in previous block diagram that unbiased carrier Doppler is scaled and added to unbiased code Doppler
  - Scale adjusts Doppler effect on carrier frequency for code aiding as follows:

$$\text{Scale factor} = \frac{R_c}{f_L}$$

  - where: R_c = code chipping rate (chips/s)
  - f_L = L-band carrier (Hz)

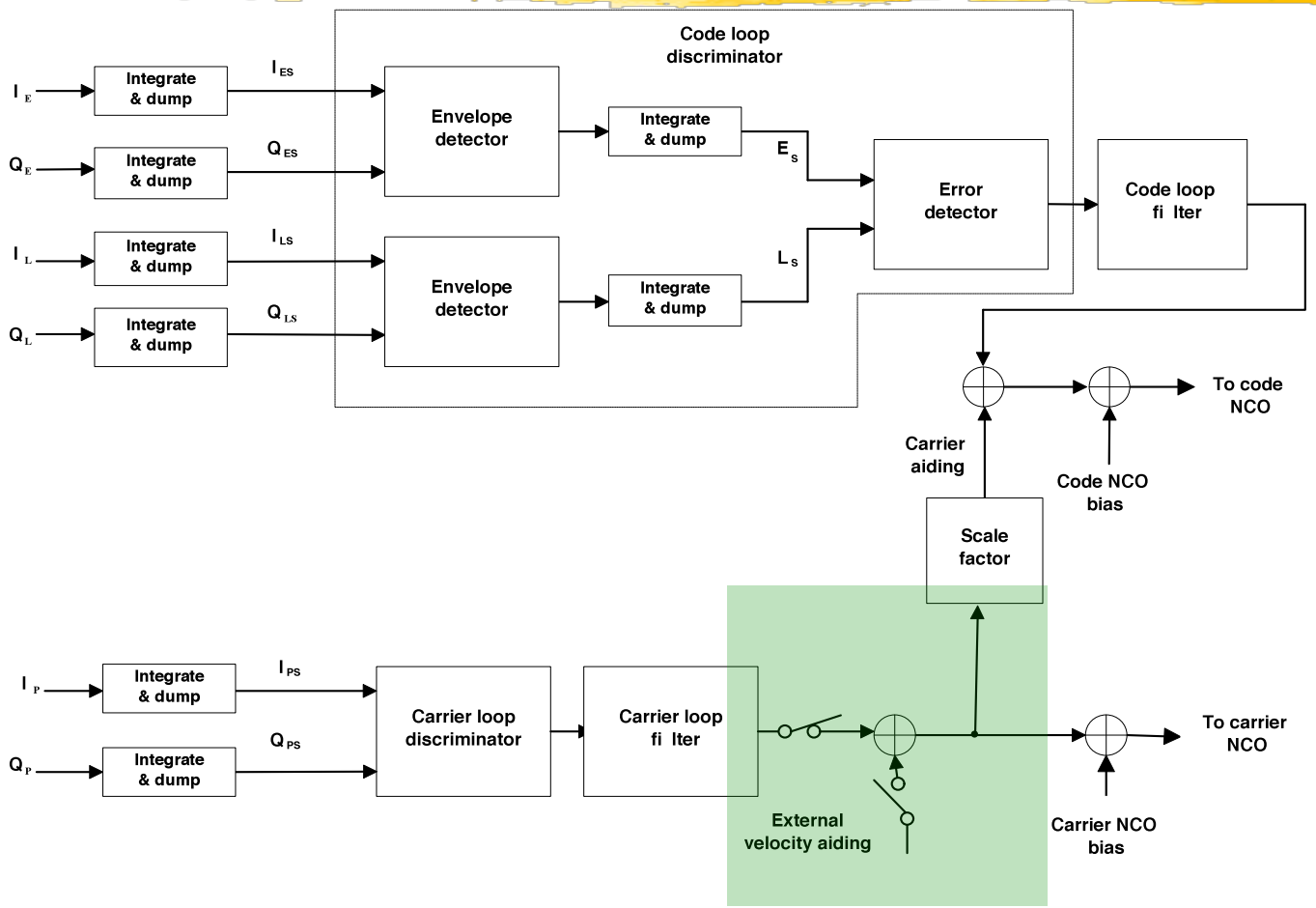
Scale factors for carrier aided code

The carrier loop output should always provide aiding to the code loop because the carrier loop jitter is about three orders of magnitude less noisy than the code loop for the same loop filter noise bandwidth and thus more accurate. The carrier loop aiding removes virtually all of the line-of-sight dynamics from the code loop, so the code loop filter order can be made smaller, the predetection integration time can be made longer and the code loop bandwidth can be made much narrower than for the unaided case, thereby increasing the code loop tracking threshold and reducing the noise in the code loop measurements. Since both the code and carrier loops must maintain track, there is nothing lost in tracking performance by using carrier aiding for an unaided GPS receiver even though the carrier loop is the weakest link.

Scale factors for carrier aided code

Carrier frequency f_L (Hz)	Code Rate R_c (chips/s)	Scale Factor
L1 = 154 f_0	C/A = R_0/10	1/1540 = 0.00064935
L1 = 154 f_0	P(Y) = R_0	1/154 = 0.00649350
L2 = 120 f_0	P(Y) = R_0	1/120 = 0.00833333

Carrier aiding of code loop: external aiding



External aiding

As shown in the generic baseband processor code and carrier tracking loops block diagram, external velocity aiding from an inertial measurement unit (IMU) can be provided to the receiver channel in closed carrier loop operation. The switch, shown in the unaided position, must be closed when external velocity aiding is applied. The external rate aiding must be converted into line-of-sight velocity aiding with respect to the GPS satellite. The lever arm effects on the aiding must be computed with respect to the GPS antenna phase center, which requires a knowledge of the vehicle attitude and the location of the antenna phase center with respect to the navigation center of the external source of velocity aiding. For closed carrier loop operation, the aiding must be very precise and have little or no latency or the tracking loop must be delay compensated for the latency. If open carrier loop aiding is implemented, less precise external velocity aiding is required, but there can be no delta range measurements available from the receiver so it is a short term, weak signal hold-on strategy. In this weak signal hold-on case, the output of the carrier loop filter must be set to zero and there is no need to process the prompt correlator signals in the carrier loop discriminator, but these signals are still used for C/N_0 computation, etc.

The benefits of external velocity aiding are that the tracking loop dynamics are removed by the external aiding to the extent that the aiding provides "true" line-of-sight velocity into the tracking loop. This permits the tracking loop filter noise bandwidth to be made more narrow, the predetection integration time can be made longer and, typically, the order of the loop filter to be lower, than would be the case for the unaided loop which would have to track through the maximum expected dynamic stress. Reducing the noise bandwidth and increasing the predetection integration time improves the tracking threshold which improves the weak signal hold-on characteristic for situations such as antenna gain roll-off or RF interference (jamming). Reducing the order of the carrier loop filter simplifies the computational burden. In the case of the carrier loop for a military GPS receiver, it would typically be third order if unaided and second order if aided. The benefit of the second order carrier loop is that it is unconditionally stable for all bandwidths. Whereas, the third order loop can become unstable under extremely low signal to noise ratio conditions.

The carrier loop output should always provide aiding to the code loop regardless of external aiding. This is because the carrier loop jitter is about three orders of magnitude less noisy than the code loop for the same loop filter noise bandwidth and thus more accurate. The carrier loop aiding removes virtually all of the line of sight dynamics from the code loop, so the code loop filter order can be made smaller, the predetection integration time can be made longer and the code loop bandwidth can be made much narrower than for the unaided case, thereby increasing the code loop tracking threshold and reducing the noise in the code loop measurements. Since both the code and carrier loops must maintain track, there is nothing lost in tracking performance by using carrier aiding for an unaided GPS receiver even though the carrier loop is the weakest link.

External aiding: where and how it is injected in each tracking loop

- Inertial measurement unit (IMU) - tightly coupled
  - Convert into line-of-sight velocity aiding with respect to the SV
  - Compute lever arm effects between IMU navigation center and receiver antenna phase center
  - Minimize latency or provide delay compensation in carrier tracking loop
- Open carrier loop aiding (code loop aiding)
  - Requires less precise IMU velocity aiding
  - No delta range measurements or SV data available
    - Use only as short term weak signal hold-on strategy

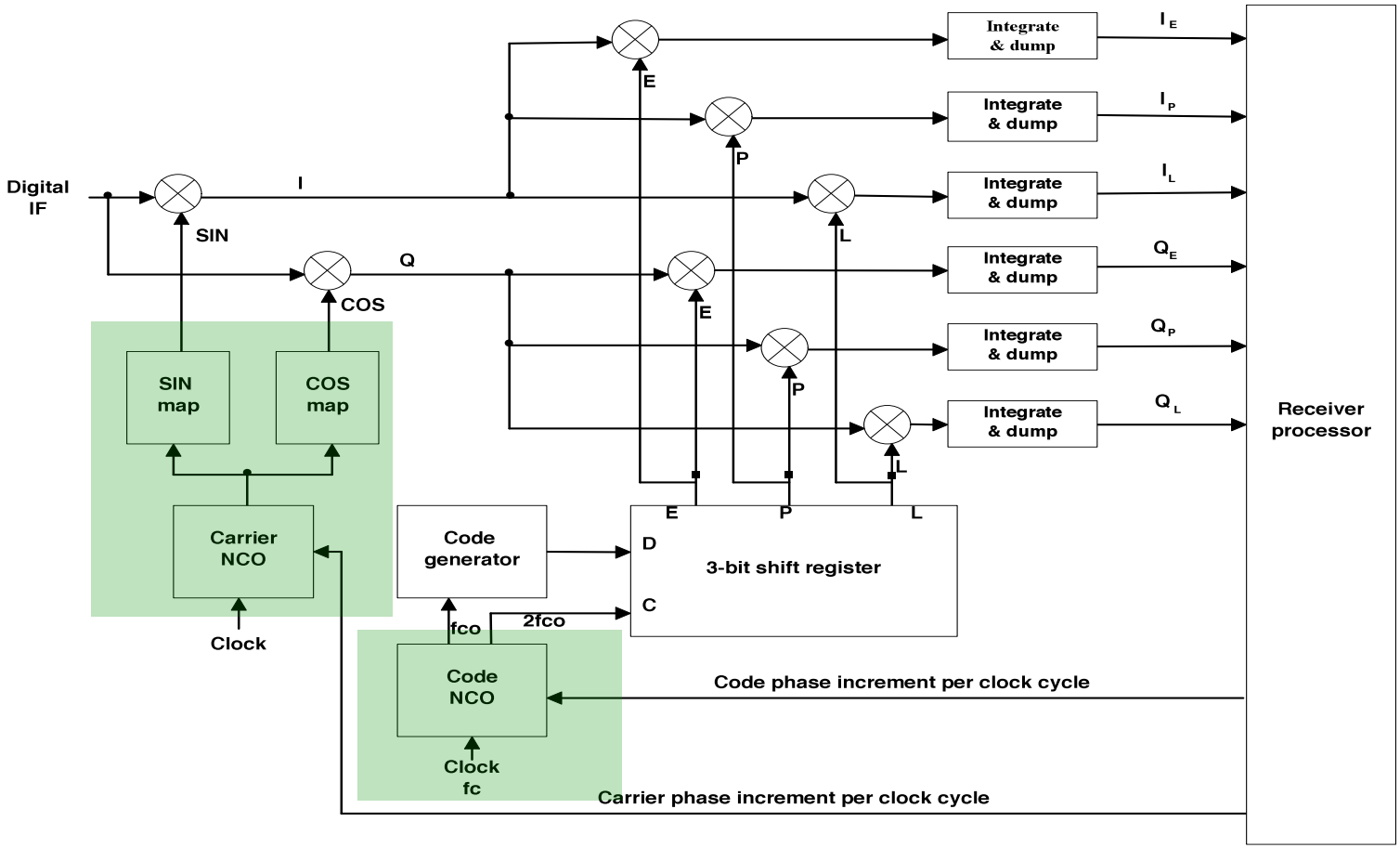
Digital frequency synthesizer block diagram and output waveforms

- Numerically controlled oscillator (NCO)
- Digital frequency synthesizer block diagram
- Digital frequency synthesizer output waveforms
- Digital frequency synthesizer phase diagram
- Mapping NCO output to COS and SIN outputs

Numerically controlled oscillator (NCO)

- NCO is simply an accumulator (increment counter)
- Receiver processor sends commands to NCO (usually at 50 - 100 Hz rate)
  - Based on microprocessor's estimate of phase/frequency tracking errors
  - Initialized with phase state
  - Controlled with phase-rate (phase increment per clock)
- NCO increments phase estimate by: phase + phase-rate × elapsed time
- For carrier wipe-off, NCO produces cosine and sine of phase estimate using lookup table

Digital receiver channel – carrier frequency synthesizer and code NCO

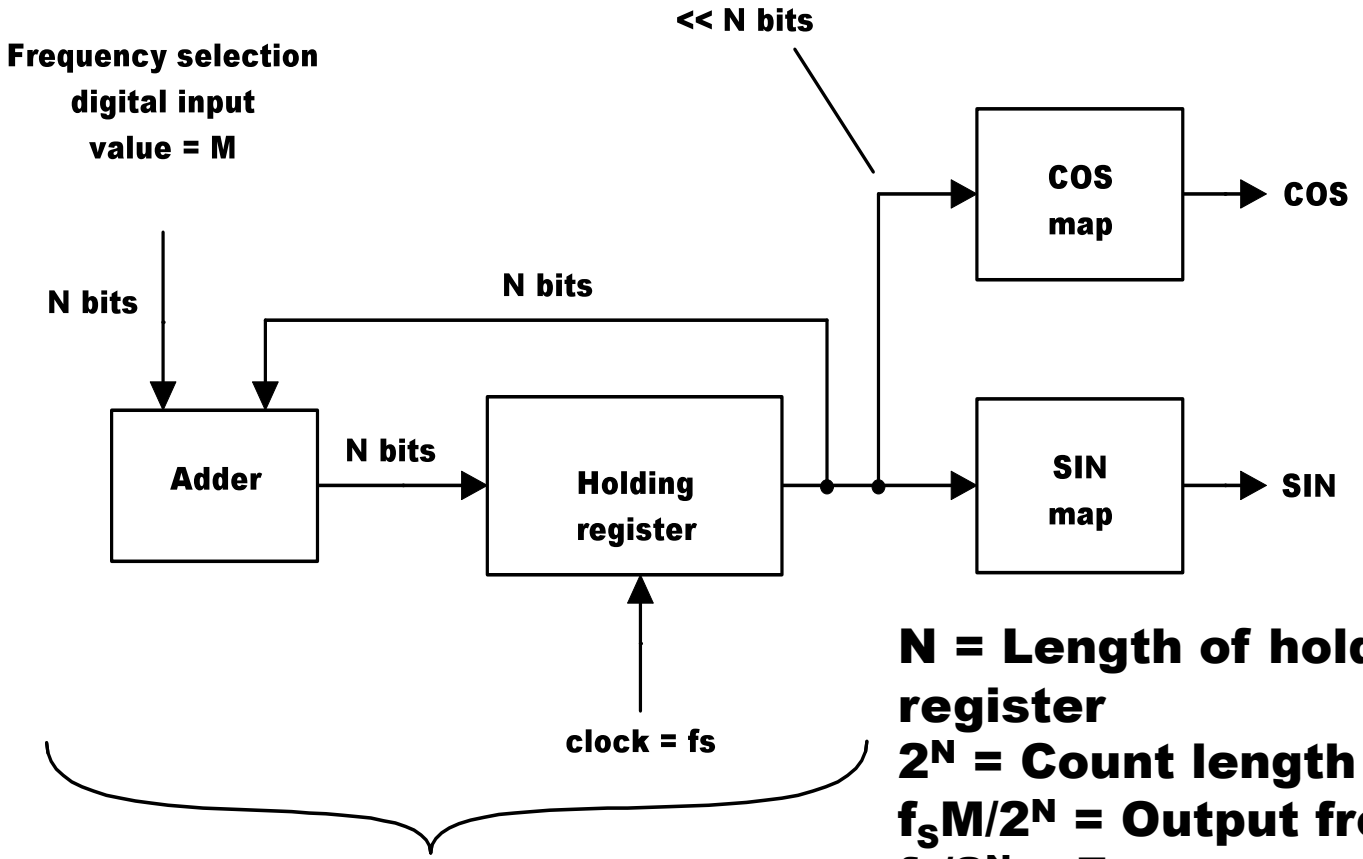


Digital frequency synthesizer block diagram

In this generic design example, both the carrier and code tracking loops use an NCO. One replica carrier cycle and one replica code cycle are completed each time the NCO overflows. A block diagram of the carrier loop NCO and its sine and cosine mapping functions are shown in the figure below

The figure depicts an expanded block diagram of the digital frequency synthesizer used in the carrier tracking loop of the generic digital receiver channel. It consists of a numerical controlled oscillator (NCO) and a cosine/sine mapping function. The typical number of bits for the carrier NCO is 32. This provides extremely high frequency resolution of $f_s/2^N$, where fs is the clock frequency and N is the number of bits. The NCO is used in both the code and carrier hardware synthesis functions. There are some commercial GPS receivers which do not implement a true code loop NCO. Instead, they simply operate the code generator at the nominal chipping rate from a simple divider. When the drift of the replica code is detected by the code tracking loop, then clock pulses are added or swallowed as required to re-center it. This results in very noisy pseudorange quantization. Instead of a fine adjustment of the code chipping rate of the NCO, the phase adjustment is in fractions of a chip as determined by the clock divider circuit. For example, if the clock divider circuit is 50, the adjustment is 1/50th of a chip, which is about 6 meters of noise for the C/A-code

Digital frequency synthesizer block diagram

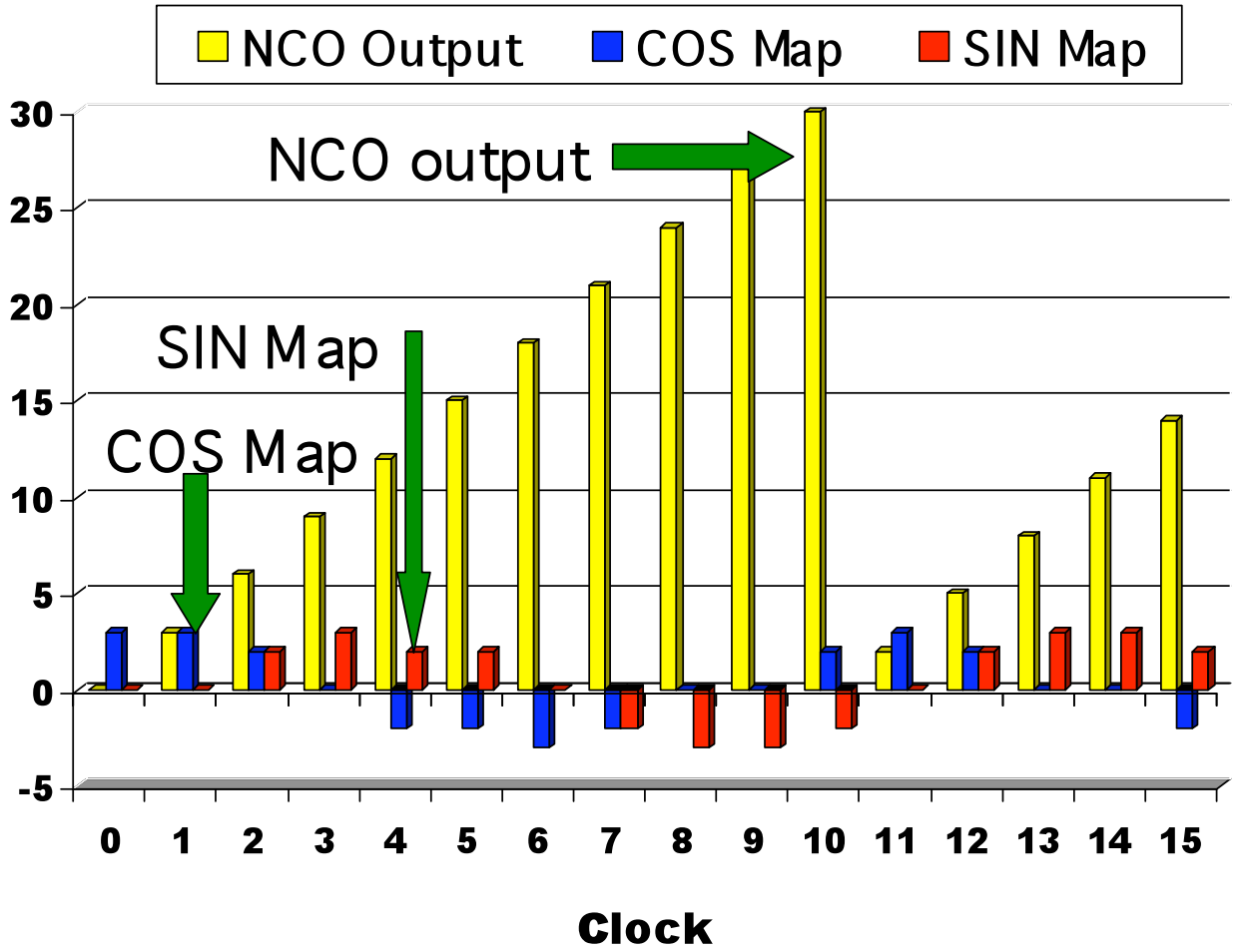


N = Length of holding register
2^N = Count length
$f_S M/2^N$ = Output frequency
$f_S/2^N$ = Frequency resolution

Digital frequency synthesizer output values

The figure below depicts the values of the staircase output of a simple 5-bit NCO with the cosine output and sine output of the carrier frequency synthesizer shown as a result of mapping the top. In the example the increment M is 3. The map function example shown corresponds to a 3-bit quantization map shown in a following figure. If the A/D quantization is only one bit, then it would be necessary only to map the sign bit of the holding register. Note that every overflow of the holding register corresponds to one carrier cycle. The frequency of this staircase output is increased by increasing the digital value of M, which is called the carrier phase increment per clock cycle in the generic digital receiver channel block diagram. The frequency is decreased by decreasing the digital value of M. The carrier NCO bias that is added in the generic baseband processor code and carrier tracking loops block diagram is the value which most closely sets the NCO to the zero Doppler carrier frequency (at the IF) of the SV being tracked. For the code NCO, only the staircase overflow output is needed. For the example shown in the generic digital receiver channel block diagram, two outputs are used. One, at twice the frequency of the code generator rate, provides ½-chip separation between the three code phases. This output is used by the 3-bit shift register. The output from the most significant bit of the NCO provides the chipping rate into the replica code generator. The code NCO bias shown in the generic baseband processor code and carrier tracking loops block diagram is the value which causes the NCO to run at the zero Doppler code chipping rate (10.23 Mchips/s for P(Y)-code and 1.023 Mchips/s for C/A-code). Note that the phase state of the NCO at any time represents the fractional part of the replica code phase. For a 32-bit holding register, this provides nanometers of code range resolution compared to the crude quantization capability of the fixed divider technique.

Digital frequency synthesizer output waveforms



Digital frequency synthesizer design

The figure and table below describe the techniques step-by-step for designing a digital frequency synthesizer. The example shown is for J = 3-bits, where J = the amplitude quantization used by the receiver A/D converter used at the IF of the receiver. $2^J = K$ values are computed. If more than one bit is used, a table look-up is usually provided for the sine and cosine maps. Note that only one quarter of a cycle need be stored in only one map if the correct logic is used to determine which quadrant the holding register output is in for the respective sine and cosine output functions.

Notes:

1. The number of bits, J, is determined for the sin and cos outputs. The phase plane of 360^o is subdivided into $2^J = K$ phase points.

2. K values are computed for each waveform, one value per phase point. Each value represents the amplitude of the waveform to be generated at that phase point. The upper J bits of the holding register are used to determine the address of the waveform amplitude.

3. Rate at which phase plane is traversed determines the frequency of the output waveform.

4. The upper bound of the amplitude error is:

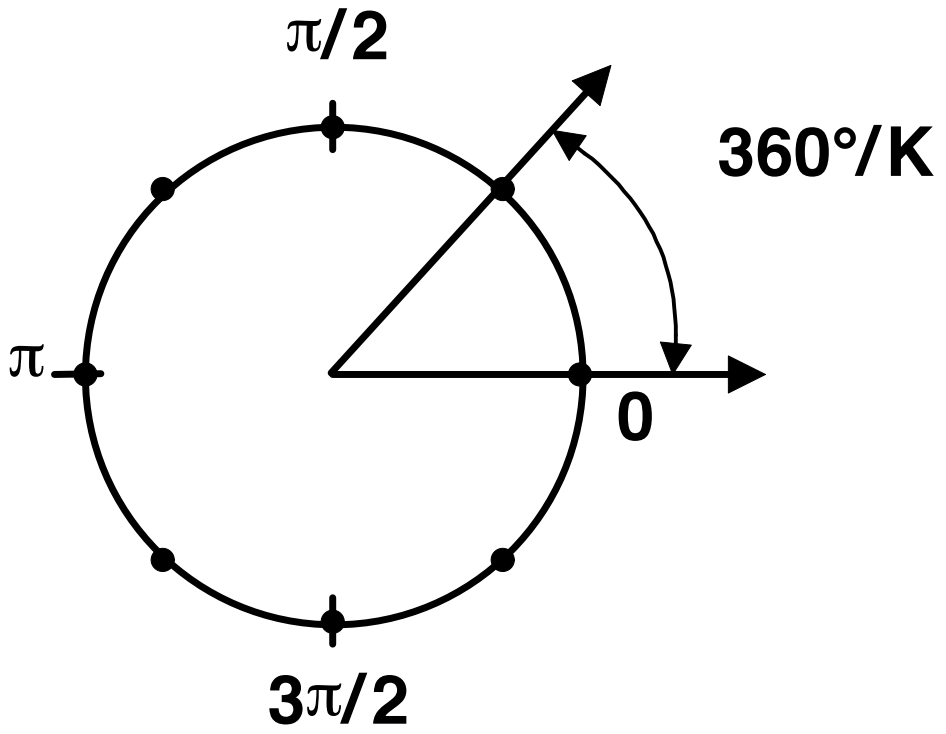
   $e_{MAX} = 2\pi/K$

5. The approximate amplitude error is: $e = e_{MAX}\cos\Phi(t)$
   where $\cos\Phi(t)$ is the phase angle.

Digital frequency synthesizer design

- Design example: J=3 bits
  - Where J = receiver A/D converter amplitude quantization (more bits if ADC is non-linear)
  - 2^J = K values (of sin and cos) are computed usually by table look-up (only ¼ cycle plus quadrant needed)
  - Rate at which phase plane is traversed determines frequency (see chart below)
  - Upper bound amplitude error: $e_{MAX} = 2\pi/K$
  - Approximate amplitude error: $e = e_{MAX} \cos \Phi(t)$
    - Where $\Phi(t)$ is the actual phase angle.

Digital frequency synthesizer phase diagram



Mapping NCO output to COS and SIN outputs

The table below illustrates the technique for mapping the NCO into the digital cosine and sine functions. The example shown is for J = 3-bits, where J = the amplitude quantization used by the receiver A/D converter at the IF of the receiver. $2^J = K = 8$ values are computed. Note that only one quarter of a cycle need be stored in only one map if the correct logic is used to determine which quadrant the holding register output is in for the respective cosine and sine output functions.

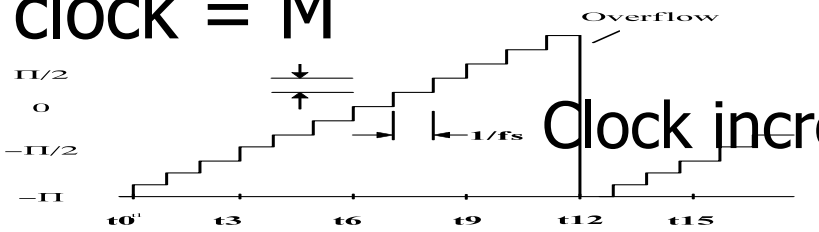
Mapping NCO output to COS and SIN outputs

Radians	Holding register	COS map	SIN map
0	000…	011…	000…
π/4	001…	010…	010…
π/2	010…	000…	011…
3π/4	011…	110…	010…
π	100…	111…	000…
5π/4	101…	110…	110…
3π/2	110…	000…	111…
7π/4	111…	010…	110…

Digital frequency synthesizer output waveforms for J=3 bits

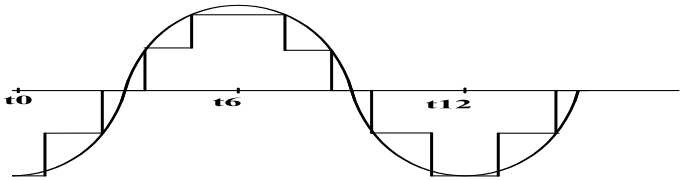
Phase increment per clock = M

NCO output

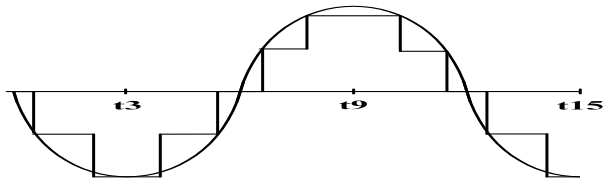


Clock increment

COS map output



SIN map output



Session III