DIFFERENTIAL ENCODING REVEALED: AN EXPLANATION OF THE TIER-1 DIFFERENTIAL ENCODING IN IRIG 106

Michael Rice
Telemetry Laboratory, Brigham Young University, Provo, Utah

ABSTRACT

IRIG 106-04 specifies differential encoding for use with the interoperable Tier-1 modulations to deal with phase and delay-axis ambiguities associated with PLL-based carrier phase synchronization. The origins of the differential encoding have been shrouded in the mists of an unavailable technical report and a mysterious connection to previous published work in the open literature. This paper removes the mystery by showing that the differential encoding rule results from encoding bit-by-bit transitions in the phase trajectory of an offset QPSK modulated carrier.

INTRODUCTION

The IRIG 106-04 Standard [1] defines three interoperable "Tier-1" modulations known as FQPSK-B, FQPSK-JR, and SOQPSK-TG. The three are "interoperable" in the sense that they all have nominally the same bandwidth (as measured by both the –60 dBc and 99% power criteria) and achieve essentially the same bit error rate when demodulated with an integrate-and-dump offset QPSK demodulator. These waveforms, all variations of offset QPSK, were adopted in IRIG 106-04 because they occupy half the bandwidth of the legacy PCM/FM modulation while achieving the same power efficiency as PCM/FM (i.e., the signal-to-noise ratio required to achieve a given bit error rate is the same for PCM/FM with limiter-discriminator detection and the Tier-1 modulations).

The power efficiency of the Tier-1 demodulators is achieved by using *coherent detection*. In coherent detection, the demodulator estimates the unknown phase of the received carrier and uses this information to produce decisions. This is in contrast with limiter-discriminator detection of PCM/FM, which produces bit decisions without tracking the unknown carrier phase. Because no provision is made to transmit a sequence of known (and clear) bits for the purposes of estimating the carrier phase at the receiver, Tier-1 demodulators must estimate the unknown carrier phase using *decision-directed* techniques, usually implemented with a phase-lock loop (PLL).

One of the consequences of decision-directed estimation techniques is that the phase estimates possess a *phase ambiguity*. In the presence of phase ambiguity, the phase estimate may be offset from the true value by a phase shift that is determined from the rotational symmetry of the signal set. For example, in simple QPSK the PLL may lock in phase with the received carrier, or out of phase by −90°, 90°, or 180°. In offset QPSK, the situation is somewhat more complicated and is described later in this paper.

The two most common methods of dealing with the phase ambiguity are

1. Embed a known field of bits, such as a unique word or synchronization word, into the data stream. After PLL lock, a data processor searches for the known field and it variants (due to rotations by the phase possible phase ambiguities) to determine the phase ambiguity. Once identified, the phase ambiguity can be corrected using a number of straight-forward techniques.
2. Differentially encode the bits prior to modulation. With differential encoding, the information bits are used to select the phase *shift* rather than the absolute phase of the modulated carrier. After the PLL has achieved lock, the bit decisions are processed by a differential decoder to recover the original information bits. A properly designed differential encoding/decoding process works even in the presence of a phase ambiguity.

The most common method used in aeronautical telemetry is the second approach: differential encoding. Appendix M of the IRIG 106-04 standard defines differential encoding and decoding for use with the Tier-1 waveforms. The differential encoding is defined by Equation (4) below. The differential encoding and decoding rules perform the desired functions in the presence of the ambiguities associated with offset QPSK. The fact that this differential encoding rule (and its corresponding decoding rule) works is explained in the exceptionally well-written Section 4 of Appendix M of IRIG 106-04. MATLAB code is also supplied in Annex 1 of Appendix M to demonstrate this fact.

While it has been known for many years *how* the differential encoding works, it has been somewhat of a mystery *why* it works. The differential encoding rule defined in IRIG 106-04 was taken from the text by Feher [2] who, in turn cited a report by Clewer [3] that is no longer available. The issue of differential encoding for offset QPSK was addressed by Weber [4], but no clear connection between Weber's work and differential encoding defined in IRIG 106-04 was seen.

Until now.

This paper derives the differential encoding rule defined in IRIG 106-04 from the basic principles set forth by Weber. This derivation not only explains the origins of the rule, but

reveals *why* it works. This paper is organized as follows. The notation associated with offset QPSK is described in the next section. Based on this notation, the need for differential encoding is briefly reviewed and the differential encoding rule is derived. For readers not familiar with phase trajectory plots, the notion of a phase trajectory for QPSK and offset QPSK is described in the appendix.

OFFSET QPSK: NOTATION

An offset QPSK modulated carrier is a signal of the form

$$s(t) = I(t)\cos(2\pi f_c t) - Q(t)\sin(2\pi f_c t) \tag{1}$$

where f_c is the carrier frequency and the inphase and quadrature components are pulse trains of the form

$$\begin{aligned} I(t) &= \sum_n a_n\, p(t - nT_s) \\ Q(t) &= \sum_n b_n\, p\left(t - {}^{T_s}\!/_2 - nT_s\right) \end{aligned} \tag{2}$$

where T_s is the symbol time (the reciprocal of the symbol rate), $p(t)$ is the pulse shape, and the pair (a_n, b_n) represents the offset QPSK symbol. The offset QPSK symbol is usually defined using a constellation, such as the one illustrated in Figure 1. Each of the four symbols in the constellation of Figure 1 is labeled with a 2-bit pattern. This label specifies the bit-to-symbol mapping which defines how input bits are mapped to symbols. The mapping is illustrated in Figure 2 for the bit sequence $d_0 d_1 d_2 d_3 d_4 d_5 d_6 d_7 = 10011011$. These 8 bits produce the four symbols $(a_0, b_0) = (+1, -1)$, $(a_1, b_1) = (-1, +1)$, $(a_2, b_2) = (+1, -1)$, $(a_3, b_3) = (+1, +1)$.

Observe that one new symbol is produced for every two bits. If T_b is used to denote the bit interval (the reciprocal of the bit rate), then the relationship between the bit time and the symbol time is $T_s = 2T_b$. The temporal relationship is illustrated in Figure 3 for the bit sequence 10011011. Observe that $d_0 = 1$ defines $a_0 = +1$; $d_1 = 0$ defines $b_0 = -1$; $d_2 = 0$ defines $a_1 = -1$; $d_3 = 1$ defines $b_1 = +1$; and so on. The convention adopted in this paper is that the even-indexed bits determine the signs of the inphase waveforms while the odd-indexed bits determine the signs of the quadrature waveforms. In what follows the two bits d_{2k} and d_{2k+1} define the symbol (a_k, b_k). Note that with k an integer, $2k$ is always even and $2k+1$ is always odd.

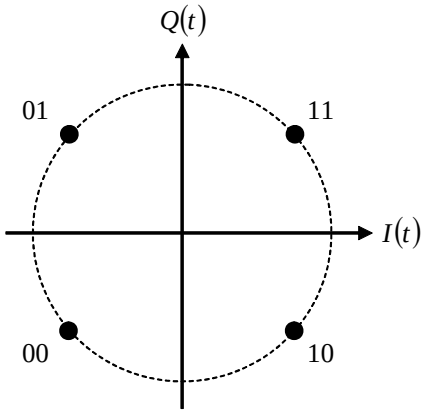


Figure 1: An example of a QPSK or offset QPSK constellation showing the bit-to-symbol assignments.

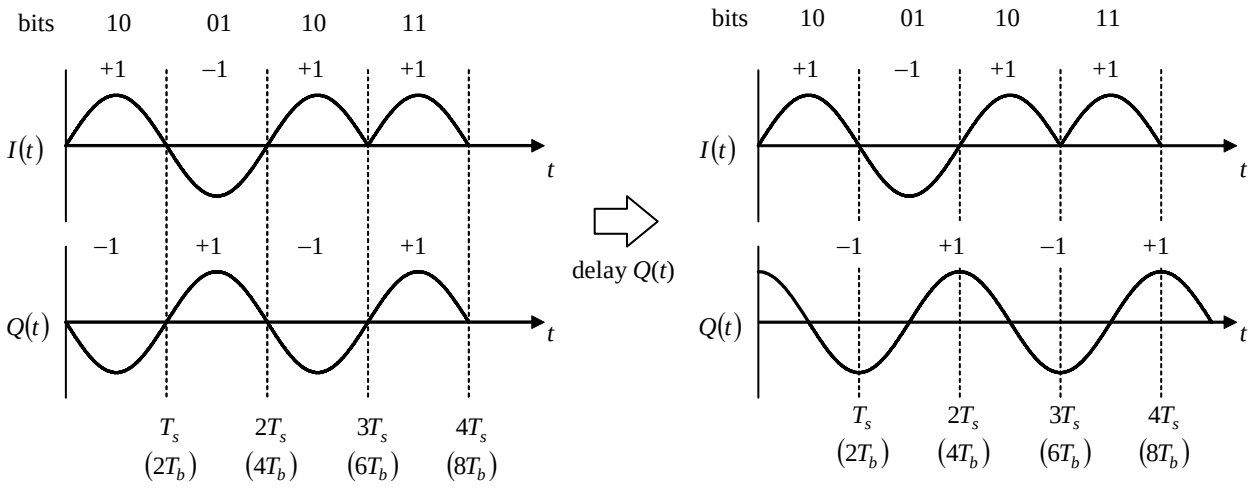


Figure 2: An illustration of the relationship between the input bits and the symbols for QPSK (left) and offset QPSK (right).

d_0	d_1	d_2	d_3	d_4	d_5	d_6	d_7	
1	0	0	1	1	0	1	1	
$a_0 = +1$		$a_1 = -1$		$a_2 = +1$		$a_3 = +1$		$\cdots$
$\cdots$	$b_0 = -1$		$b_1 = +1$		$b_2 = -1$		$b_3 = +1$	

Figure 3 An illustration of the temporal relationship between the input bits d_{2k} and d_{2k+1} and the offset QPSK symbol components a_k and b_k. Compare with right-hand plot of Figure 2.

THE NEED FOR DIFFERENTIAL ENCODING

A block diagram of an offset QPSK demodulator, when an emphasis on the phase-lock-loop (PLL) used for carrier phase synchronization is illustrated in Figure 4. In most implementations, the IF signal is mixed to I/Q baseband, filtered by low-pass detection filters, and sampled two times per symbol (or once per bit) in synchronism with the symbol transitions. The samples are derotated by the residual carrier phase offset that is tracked by the carrier phase PLL as shown.[1] The even-indexed samples of the derotated inphase component x_{2k} and the odd-indexed samples of the derotated quadrature component y_{2k+1} are used to make the bit decisions. The other two derotated detection filter outputs, x_{2k+1} and y_{2k} are required by the phase error detector to compute the phase error signal. The maximum likelihood phase error signal is

$$e_{2k} = \text{sgn}\{x_{2k}\}y_{2k} - \text{sgn}\{y_{2k+1}\}x_{2k+1}. \tag{3}$$

Observe that the error signal is produced only once per symbol (or every two bit times). This is captured by using 2*k* (which is always even) for the index. The relationship between the residual carrier phase offset θ_e and the phase error detector output e_{2k} is given by the "S-curve." The S-curve is obtained by expressing the phase error detector output as a function of θ_e and averaging over the data symbol values. The S-curve for the maximum likelihood phase error detector (3) is plotted in Figure 5. The loop dynamics are such that the stable lock points correspond to phase errors where the S-curve crosses 0 with a positive slope. Careful examination the S-curve of Figure 5 shows that there are two stable lock points: $\theta_e = 0$ and $\theta_e = 180^\circ$. This means the PLL can lock in phase with the phase of the IF carrier (desired) or 180° out of phase with the IF carrier (undesired). This characteristic is called a *phase ambiguity* and is a consequence of the rotational symmetry of the offset QPSK signal.

There is another ambiguity associated with decision-directed carrier phase recovery for offset QPSK. The samples used for data detection have a delay of one bit time: the quadrature component is sampled one-bit time after the inphase component. The rotation caused by an unknown carrier phase offset produces an ambiguity as to which component should be delayed. This ambiguity is called *delay-axis ambiguity*. The carrier phase PLL can lock when the inphase and quadrature components are swapped and the delay relationship between their samples is incorrect.

[1] The block diagram of Figure 4 shows the phase-lock loop closing at the rotation block. The discrete-time VCO produces the phase necessary to perform the desired rotation. Another very common implementation is to close the loop at the quadrature mixers. In this implementation, the VCO output is a pair of sinusoids whose phase varies around the IF frequency f_0. Closing the loop here does not change the S-curve or the loop dynamics (as long as the loop bandwidth is not too large.) As formulated above, a D/A converter would be required somewhere in the feedback path since the phase error computation is in discrete time and the quadrature mixers operate in continuous time. In reality, most modern demodulators sample the IF signal and perform the signal processing outlined in Figure 4 in discrete time.

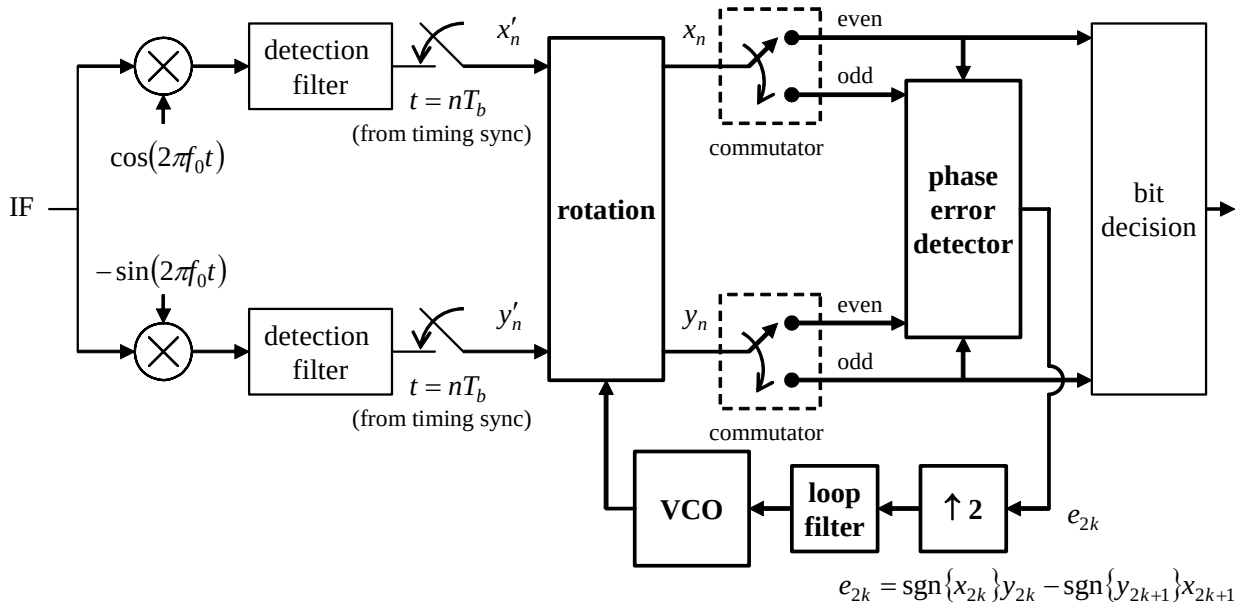


Figure 4: A block diagram of an offset QPSK demodulator emphasizing the carrier phase PLL; (b) the S-curve for the phase error detector used by the PLL.

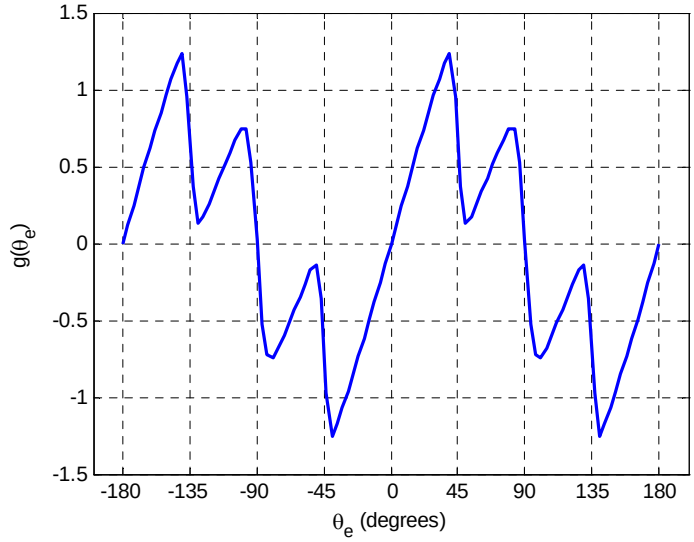


Figure 5: The S-curve corresponding to the maximum likelihood phase error signal (3) used by the PLL in Figure 4.

DIFFERENTIAL ENCODING: ORIGINS REVEALED

In differential encoding, the input bits $\ldots, d_{2k}, d_{2k+1}, \ldots$ are encoded to produce the differentially encoded bit sequence $\ldots, \delta_{2k}, \delta_{2k+1}, \ldots$ using the following encoding rule [1]

$$\begin{aligned} \delta_{2k} &= d_{2k} \oplus \overline{\delta}_{2k-1} \\ \delta_{2k+1} &= d_{2k+1} \oplus \delta_{2k} \end{aligned} \tag{4}$$

where $\oplus$ is the Boolean "exclusive OR" operation and $\overline{A}$ is the logical complement of the Boolean variable *A*. The differentially encoded bits are used in place of the data bits to select the *a*'s and *b*'s defined by the bit-to-symbol mappings in the constellation. The differential encoding is such that when the carrier phase PLL locks with phase ambiguity or delay-axis ambiguity in place, the true data bits $\ldots, d_{2k}, d_{2k+1}, \ldots$ are recoverable from the differentially encoded bits $\ldots, \delta_{2k}, \delta_{2k+1}, \ldots$ The differential decoding rule is

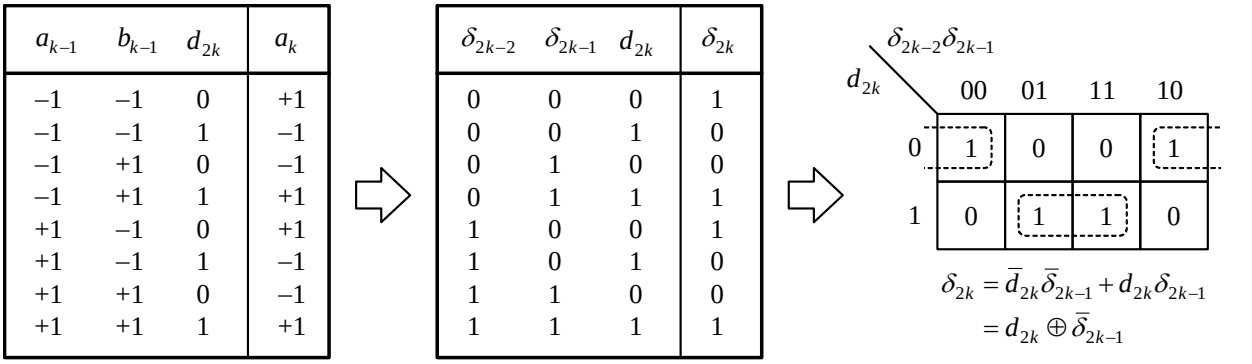
$$\begin{aligned} d_{2k} &= \delta_{2k-1} \oplus \overline{\delta}_{2k} \\ d_{2k+1} &= \delta_{2k} \oplus \delta_{2k+1} \end{aligned} \tag{5}$$

Weber [4] explained that transitions in the phase trajectory can be used to define a differential encoding rule that deals with both the 180° phase ambiguity and the delay-axis ambiguity. The delay-axis ambiguity requires the transitions in the phase trajectory be defined on a bit-by-bit basis. As explained in the Appendix, the possible phase shifts in the phase trajectory for offset QPSK are –90°, +90°, or 0 during each bit interval.

The differential encoding rule (4) results when an input bit value of 1 shifts the carrier by –90° while an input bit value of 0 shifts the carrier by +90°. When the input bit is d_{2k}, a_{k-1} and b_{k-1} are in place and a_k is to be chosen to produce the desired phase shift. Listing all possible combinations of a_{k-1}, b_{k-1}, and d_{2k} produces the left-most truth table shown in Figure 6. The first two rows of this truth table are determined with the aid of Figure 8.[2] The first two rows correspond to $a_{k-1} = -1$ and $b_{k-1} = -1$ as illustrated in Figure 8 (a). Note the phase of the modulated carrier at $t = 2kT_b$ is –90°. When the input bit $d_{2k} = 0$, a +90° phase shift is required. This is accomplished by setting $a_k = +1$ as shown in Figure 8 (b). When the input bit $d_{2k} = 1$, a –90° phase shift is required. This is accomplished by setting $a_k = -1$ as shown in Figure 8 (c). The remaining entries in the truth table for a_k are computed in a similar manner. Using the temporal relationships between a_{k-1}, b_{k-1}, a_k and δ_{2k-2}, δ_{2k-1}, δ_{2k} together with the bit-to-symbol mapping of Figure 1, the right-most truth table in Figure 6 is produced. From this truth

[2] The half-sine pulse shape is used to illustrate the principle. The idea applies with any pulse shape, even those used by the Tier-1 waveforms FQPSK-B, FQPSK-JR, and SOQPSK-TG.

table, the Karnaugh map shown in Figure 6 is constructed. From the Karnaugh map, the first relationship of (4) is obtained.

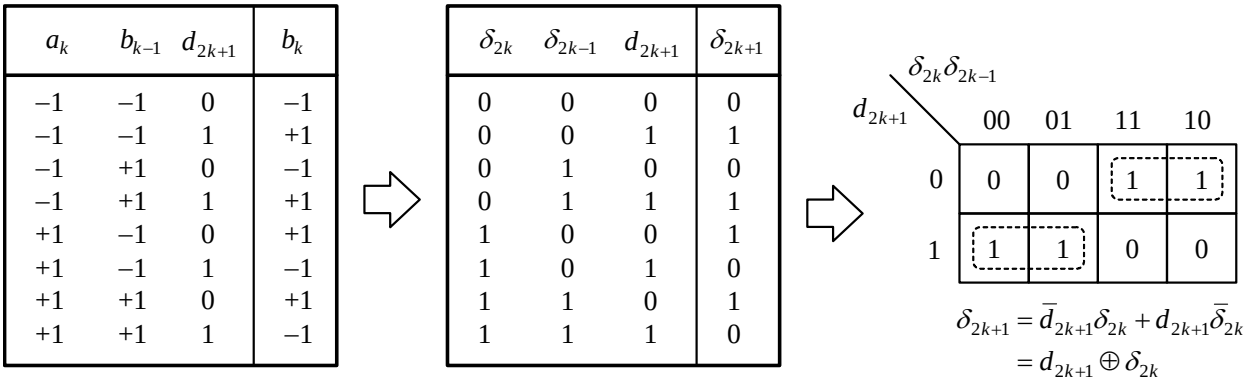


a_{k-1}	b_{k-1}	d_{2k}	a_k
−1	−1	0	+1
−1	−1	1	−1
−1	+1	0	−1
−1	+1	1	+1
+1	−1	0	+1
+1	−1	1	−1
+1	+1	0	−1
+1	+1	1	+1

δ_{2k-2}	δ_{2k-1}	d_{2k}	δ_{2k}
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

Figure 6: Truth tables and corresponding Karnaugh map defining the relationship between the differentially encoded bit δ_{2k} and the input bit d_{2k}.

When the input bit is d_{2k+1}, a_k and b_{k-1} are in place and b_k is to be chosen to produce the desired phase shift. Listing all possible combinations of a_k, b_{k-1}, and d_{2k+1} produces the left-most truth table shown in Figure 7. The first two rows of this truth table are determined with the aid of Figure 9. The first two rows correspond to $a_k = -1$ and $b_{k-1} = -1$ as illustrated in Figure 9 (a). Note the phase of the modulated carrier at $t = (2k+1)T_b$ is 180°. When the input bit $d_{2k+1} = 0$, a +90° phase shift is required. This is accomplished by setting $b_k = +1$ as shown in Figure 9 (b). When the input bit $d_{2k+1} = 1$, a −90° phase shift is required. This is accomplished by setting $b_k = -1$ as shown in Figure 9 (c). The remaining entries in the truth table for b_k are computed in a similar manner. Using the temporal relationships between a_k, b_{k-1}, b_k and δ_{2k}, δ_{2k-1}, δ_{2k+1} together with the bit-to-symbol mapping of Figure 1, the right-most truth table in Figure 7 follows. From the this truth table, the Karnaugh map shown in Figure 7 is produced. From the Karnaugh map, the second relationship of (4) is obtained.



a_k	b_{k-1}	d_{2k+1}	b_k
−1	−1	0	−1
−1	−1	1	+1
−1	+1	0	−1
−1	+1	1	+1
+1	−1	0	+1
+1	−1	1	−1
+1	+1	0	+1
+1	+1	1	−1

δ_{2k}	δ_{2k-1}	d_{2k+1}	δ_{2k+1}
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

Figure 7: Truth tables and corresponding Karnaugh map defining the relationship between the differentially encoded bit δ_{2k+1} and the input bit d_{2k+1}.

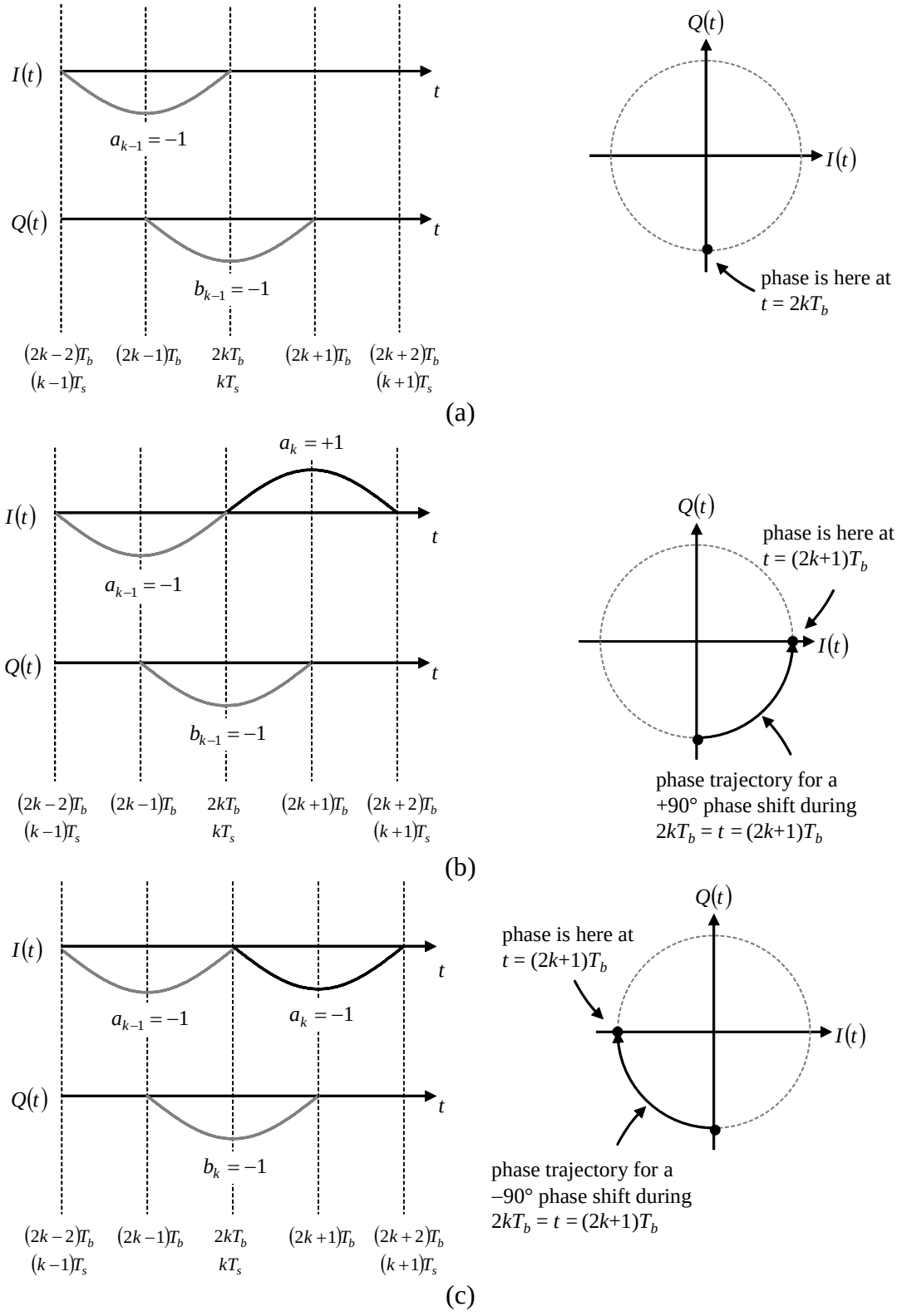


Figure 8: An illustration of the first two rows of the truth table of Figure 6.

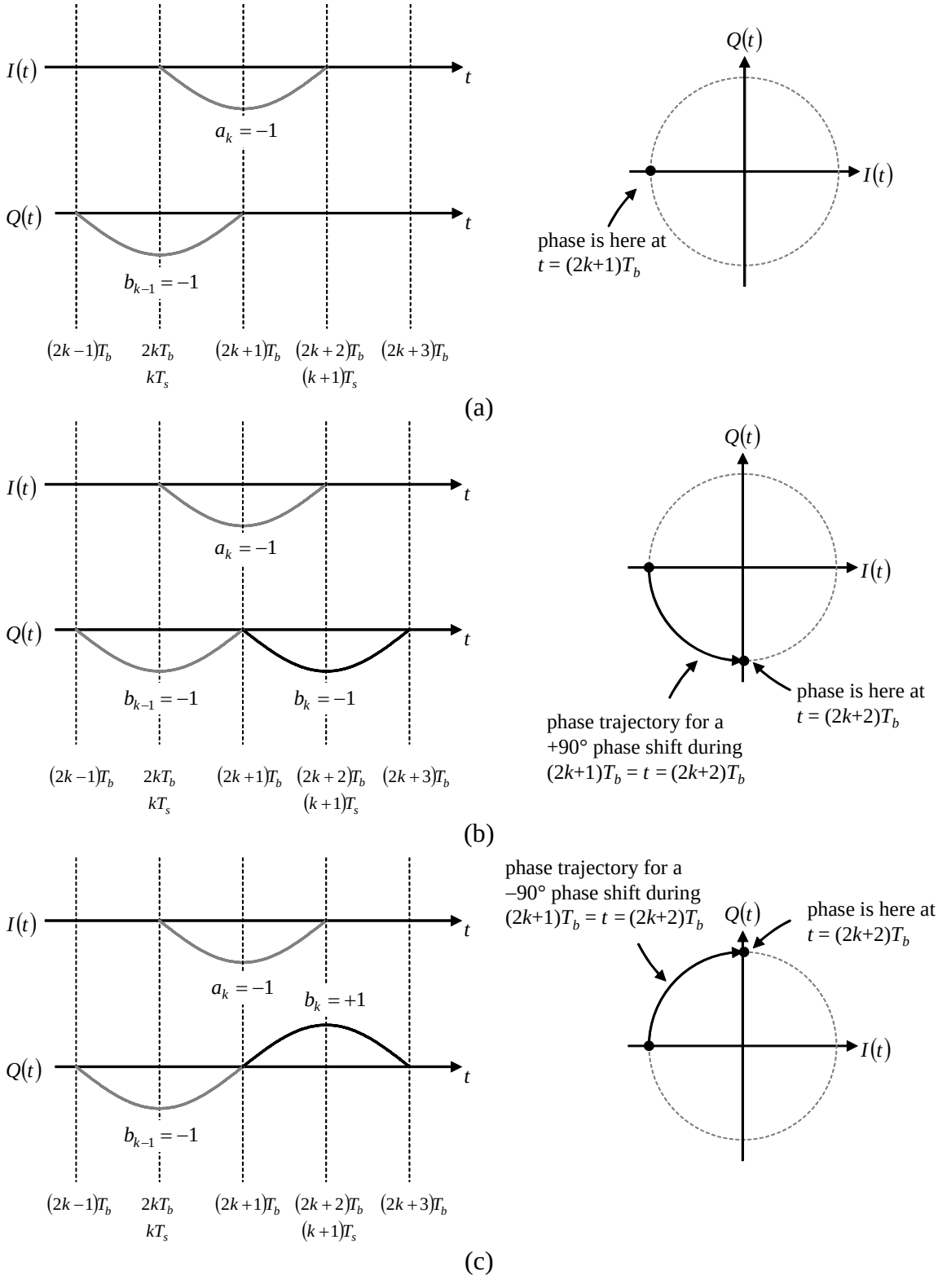


Figure 9: An illustration of the first two rows of the truth table of Figure 7.

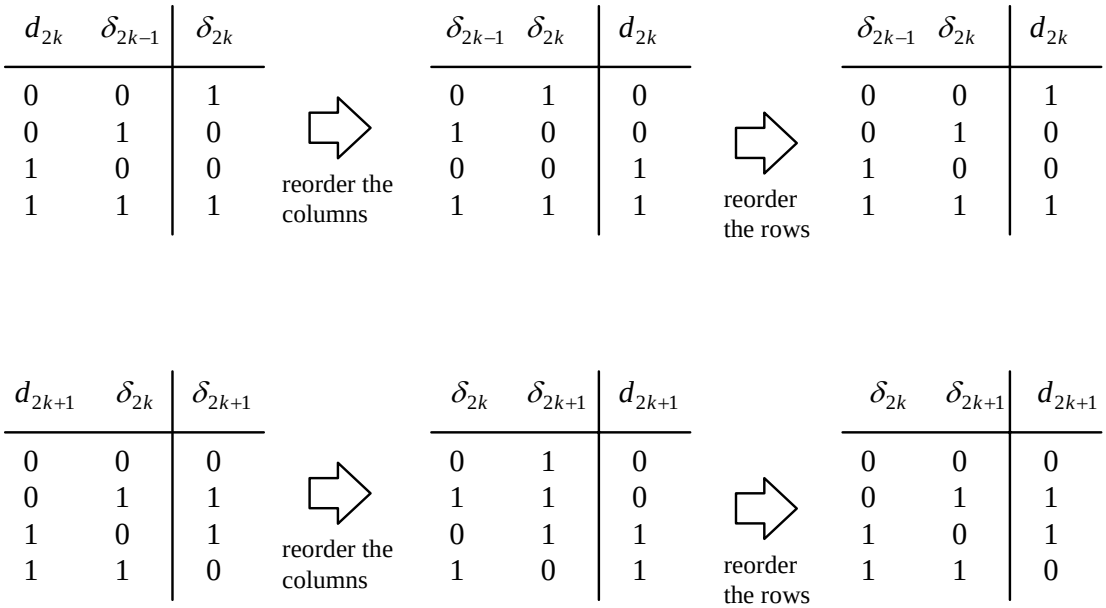


Figure 10: Development of the differential decoding rules for d_{2k} (top) and d_{2k+1} (bottom).

The differential decoding rule (5) is obtained from the differential encoding rule (4) by "solving" the two Boolean equations that define the encoding rule. This process is illustrated in Figure 10. The first step is to construct the truth table corresponding to the first Boolean equation in (4). This truth table is the left-most truth table in the upper portion of Figure 10. The convention is that the Boolean variables in the columns to the left of the vertical line are the input variables and the variables to the right of the vertical line are the output variables. Moving the d_{2k} column to the last column and moving the vertical line to the right one column produces the center truth table in the upper portion of Figure 10. Now the Boolean variable d_{2k} is an output and the variables δ_{2k-1} and δ_{2k} are input variables. Rearranging the rows of this table produces the right-most truth table in the upper portion of Figure 10. This truth table is in the standard form and defines the relationship between δ_{2k-1} and δ_{2k} (the inputs) and d_{2k} (the output). This relationship is the first Boolean expression in the differential decoding rule (5). The second Boolean expression in the differential encoding rule (5) is obtained in precisely the same way as illustrated in the lower portion of Figure 10.

The astute reader will note that the there are *two* possible differential encoding rules that could be defined in this manner. The IRIG 106-04 differential encoding rule is based on $0 \rightarrow +90°$ phase shift and $1 \rightarrow -90°$ phase shift. The alternative is to base a differential encoding rule on $0 \rightarrow -90°$ phase shift and $1 \rightarrow +90°$ phase shift. This leads to the encoding rule

$$\begin{aligned} \delta_{2k} &= d_{2k} \oplus \delta_{2k-1} \\ \delta_{2k+1} &= d_{2k+1} \oplus \overline{\delta}_{2k} \end{aligned} \tag{6}$$

Note the similarity between this encoding rule and the encoding rule (4): the logical complement is applied to δ_{2k} in the second relationship in (6) whereas the logical complement is applied to δ_{2k-1} in the first relationship in (4). While this differential encoding rule is based on the same principles as the differential encoding rule (4), it is not interoperable with that rule. The encoder and decoder must be based on the same rule to work correctly. The differential encoding rule (6) is not inferior in any way.

CONCLUSIONS

The differential encoding rule, required for phase and delay-axis ambiguity resolution was derived from basic principles. The rule is based on the use of input bits to define phase transitions in the phase trajectory. This point of view also revealed a different (but non-compatible) differential encoding that could have been used.

ACKNOWLEDGEMENTS

I am indebted to Mr. Robert Jefferis, Tybrin Corp., AFFTC Edwards AFB, for asking me the questions that led to this development. It was his desire to better understand differential decoding that prompted the investigation that culminated in this paper.

REFERENCES

[1] IRIG Standard 106-04: Telemetry Standards, 2004. Telemetry Group, Range Commanders Council, White Sands Missile Rage, NM.

[2] Feher, K., Digital Communications: Satellite/Earth Station Engineering, Prentice-Hall, Upper Saddle Rive, NJ, 1983.

[3] Clewer, R., "Report on the Status of Development of the High Speed Digital Satellite Modem," RM-009-79-24, Spar Aerospace, Ltd., St. Anne de Bellevue, P.Q., Canada, November 1979.

[4] Weber, W., "Differential Encoding for Multiple Amplitude and Phase Shift Keying Systems, *IEEE Transactions on Communications*, vol. 26, March 1978, pp. 385 – 391.

[5] Rappaport, T., Wireless Communications: Principles and Practice, Prentice Hall, Upper Saddle River, NJ, 2002.

[6] Svensson, A. and Sundberg, C.-E., "Optimum MSK-Type Receivers for CPM on Gaussian and Rayleigh Fading Channels," IEE Proceedings, August 1984, p. 480 – 490.

[7] Svensson, A., and Sundberg, C.-E., "Serial MSK-Type Detection of Partial Response Continuous Phase Modulation," IEEE Transactions on Communications, vol. 33, January 1985, pp. 44 – 52.

[8] Galko, P. and Pasupathy, S., "Optimization of Linear Receivers for Data Communication Signals," IEEE Transactions on Information Theory, vol. 34, January 1988, pp. 79 – 92.

[9] Galko, P. and Pasupathy, S., "Linear Receivers for Correlatively Coded MSK," IEEE Transactions on Communications, vol. 33, April 1985, pp. 328 – 347.

[10] Rhodes, R., Wilson, S., and Svensson, A., "MSK-Type Reception of Continuous Phase Modulation: Cochannel and Adjacent Channel Interference," IEEE Transactions on Communications, vol. 35, February 1987, pp. 185 – 193.

[11] Laurent, P., "Exact and Approximate Construction of Digital Phase Modulations by Superposition of Amplitude Modulated Pulses (AMP)," IEEE Transactions on Communications, vol. 34, February 1986, pp. 150 – 160.

[12] Kaleh, G., "Simple Coherent Receivers for Partial Response Continuous Phase Modulation," IEEE Journal on Selected Areas in Communications, vol. 7, December 1989, pp. 1427 – 1436.

[13] Mengali, U. and Morelli, M., "Decomposition of M-ary CPM Signals into PAM Waveforms," IEEE Transactions on Information Theory, vol. 41, September 1995, pp. 1265 – 1275.

[14] Colavolpe, G. and Raheli, R., "Reduced-Complexity Detection and Phase Synchronization of CPM Signals," IEEE Transactions on Communications, vol. 45, September 1997, pp. 1070 – 1079.

[15] Perrins, E. and Rice, M., "PAM Decomposition of M-ary Multi-h CPM," IEEE Transactions on Communications, vol. 53, vol. 12, December 2005, pp. 2065 – 2075.

[16] Perrins, E. and Rice, M., "Reduced-Complexity Detectors for Multi-h CPM in Aeronautical Telemetry," IEEE Transactions on Aerospace and Electronic Systems, vol. 43, January 2007, pp. 286 – 300.

[17] Hill, T., "An Enhanced Constant Envelope, Interoperable Shaped Offset QPSK (SOQPSK) Waveform for Improved Spectral Efficiency," in Proceedings of the International Telemetering Conference, San Diego, CA, October 2000, pp. 137 – 136.

[18] Gao, W. and Feher, K., "FQPSK: A Bandwidth and RF Power Efficient Technology for Telemetry Applications," in Proceedings of the International Telemetering Conference, Las Vegas, NV, October 1997, pp. 480 – 488.

[19] Nelson, T., Perrins, E., and Rice, M., "Common Detectors for Tier 1 Modulations," in Proceedings of the International Telemetering Conference, Las Vegas, NV, October 2005, pp. 518-527.

APPENDIX: OFFSET QPSK AND PHASE TRAJECTORY PLOTS

The notion of a phase trajectory plot is fundamental to understanding the nature of the differential encoding used for offset QPSK. Any modulated carrier of the form may be written in the form

$$s(t) = A(t)\cos(2\pi f_c t + \phi(t)) \tag{7}$$

where $A(t)$ is the amplitude and $\phi(t)$ is the phase. Using the trigonometric identity

$$\cos(A+B) = \cos(A)\cos(B) - \sin(A)\sin(B), \tag{8}$$

the modulated carrier may be expressed as

$$s(t) = \underbrace{A(t)\cos(\phi(t))}_{I(t)}\cos(2\pi f_c t) - \underbrace{A(t)\sin(\phi(t))}_{Q(t)}\sin(2\pi f_c t) \tag{9}$$

where $I(t)$ is the *inphase* component of $s(t)$ and $Q(t)$ is the *quadrature* component of $s(t)$. A plot of $Q(t)$ versus $I(t)$ is called a phase trajectory plot. The phase trajectory plot displays the instantaneous amplitude and phase of the modulated carrier. The instantaneous amplitude is given by

$$A(t) = \sqrt{I^2(t) + Q^2(t)} \tag{10}$$

which can be interpreted as the distance from the origin to the point in two-space $(I(t), Q(t))$. The instantaneous phase is

$$\phi(t) = \tan^{-1}\left\{\frac{Q(t)}{I(t)}\right\} \tag{11}$$

which can be interpreted as the angle formed by line defined by the origin and the point $(I(t), Q(t))$ and the line formed by the $I(t)$-axis.

As an example consider the QPSK constellation shown in Figure 1. For *unshaped QPSK*, the amplitudes for the inphase and quadrature components are constants whose sign is determined by the input bits through the bit-to-symbol mapping, such as the mapping defined by the labels on the constellation points in Figure 1. The input bits arrive with period T_b and are grouped into nonoverlapping blocks of 2 bits to form a QPSK *symbol*. A new QPSK symbol is formed every $T_s = 2T_b$ seconds. An example of the inphase and quadrature components for unshaped QPSK for the bit pattern 10011011 is illustrated in Figure 11 (a). The corresponding phase trajectory is also shown on the right. The initial state of the phase trajectory is the point on the in the fourth

quadrant. The arrows are notional as the phase changes state *instantaneously*. (This is why unshaped QPSK has such a large bandwidth.)

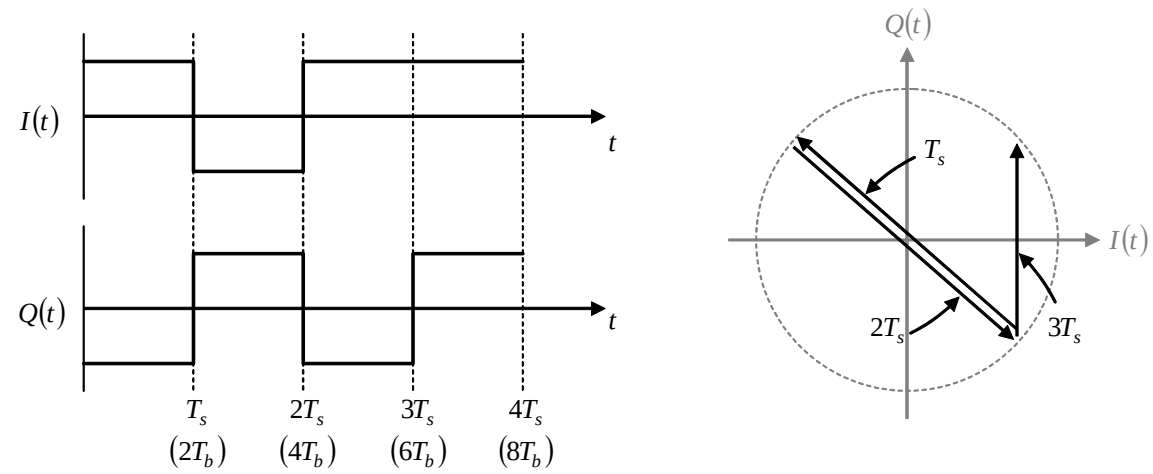
When a non-linear power amplifier is used, the phase trajectory transitions through the origin fully exercise the nonlinear characteristics of the power amplifier and increase the bandwidth of the amplified signal. Phase trajectory transitions through the origin occur when sign changes for the inphase and quadrature components coincide. This can be avoided by delaying the quadrature component by half of a symbol time (or $T_s / 2 = T_b$ seconds). This variant of QPSK is known as *staggered* or *offset* QPSK. Offset QPSK can be defined using the same constellation that was used for QPSK, such as the one shown in Figure 1. The inphase and quadrature components for unshaped offset QPSK are shown in Figure 11 (b) for the same 10011011 input bit sequence. Again, the arrows are notional because the transitions are instantaneous.

Two observations are important:

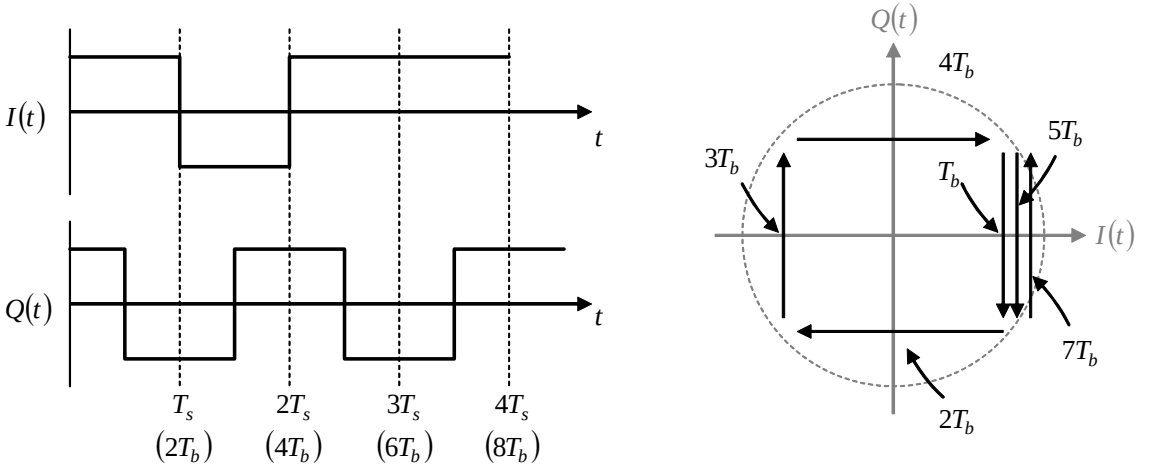
1. The phase trajectory does not pass through the origin. This is because the inphase and quadrature component cannot change sign simultaneously.
2. Transitions occur every bit time and correspond to a phase shift of 0°, +90°, or –90°.

When shaping applied to the inphase and quadrature components, the bandwidth is reduced. An example of a simple pulse shape is the half-sine pulse shape. Revisiting the example of Figure 11 (b) and substituting the half-sine pulse shape on the inphase and quadrature components produces the plots illustrated in Figure 12. Note that the phase trajectory with the half-sine pulse shape is restricted to the a circle in the I/Q plane. This property is called the *constant envelope* property and is the very desirable property when a non-linear power amplifier is used.

The ARTM Tier-1 waveforms – FQPSK-B, FQPSK-JR, and SOQPSK-TG – use more sophisticated pulse shaping. An eye diagram is useful for displaying the properties of the inphase and quadrature components, especially when there are a variety of possible data-dependent phase transitions. Eye diagrams for unshaped offset QPSK, offset QPSK with the half-sine pulse shape, SOQPSK-TG, and FQPSK-JR are illustrated in Figure 13. Observe that in all cases, the eye diagrams for the inphase and quadrature components show eye openings that are staggered or *offset* from each other by one-half the symbol time. The phase trajectory plots corresponding to the FQPSK-JR and SOQPSK-TG cases have the same two properties mentioned above.



(a)



(b)

Figure 11: Examples of a phase trajectory plot corresponding to the bit pattern 10011011 using the bit-to-symbol mapping of Figure 1: (a) the phase trajectory for QPSK; (b) phase trajectory for offset QPSK.

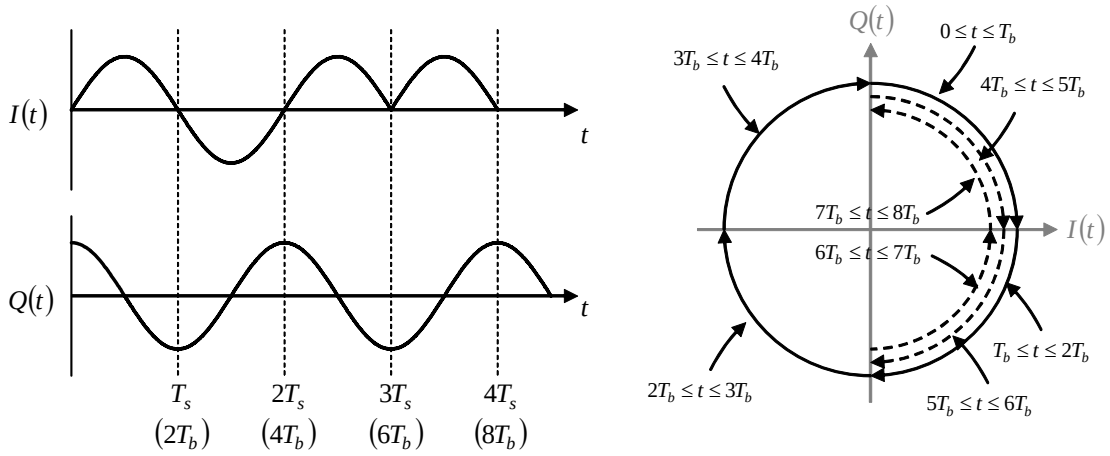
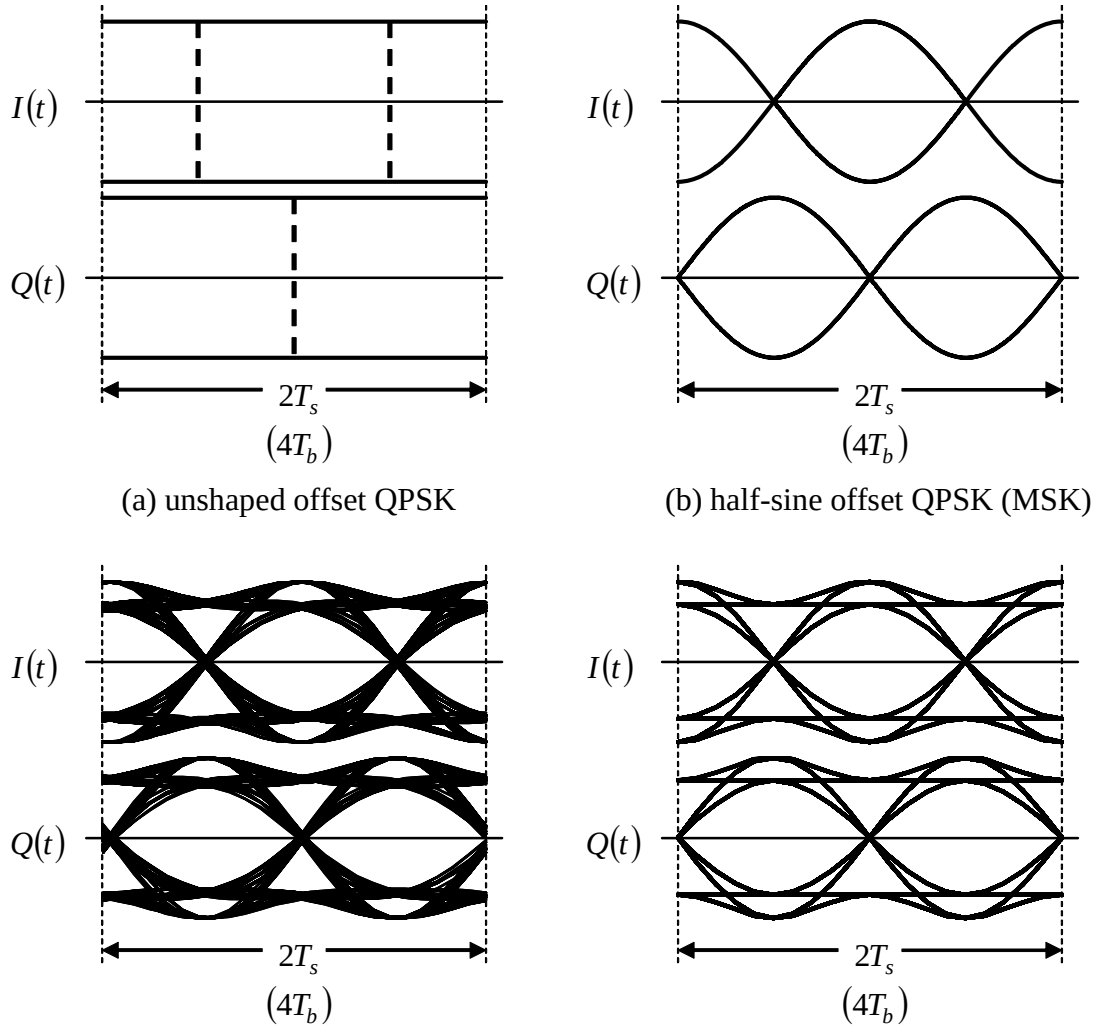


Figure 12: Example of the inphase and quadrature components for offset QPSK using the half-sine pulse shape corresponding to the bit sequence 10011011 using the bit-to-symbol mapping of Figure 1. The phase trajectory plot is shown on the right. The dashed lines are for display purposes only. All transitions occur along the circle. Compare with Figure 11 (b).

It should be pointed out that the phase trajectories for offset QPSK with pulse shapes that achieve the constant envelope property are identical to the phase trajectories for continuous phase modulations (CPMs) with modulation index $h = ½$. Examples of CPMs with $h = ½$ include minimum shift keying (MSK) – the special case of continuous phase FSK for $h = ½$ – and Gaussian minimum shift keying (GMSK) – the modulation used in GSM [5]. The connection between CPM with $h = ½$ and offset QPSK has been the basis for simplified demodulator design [6] – [10]. The relationship between binary CPM and linear modulation was generalized by Laurent [11] and exploited by Kaleh [12] to device reduced complexity demodulators. The extension of Laurent's relationship to non-binary CPM was developed by Mengali and Morelli [13] and formed the basis for simplified demodulator design by Colavolpe and Raheli [14]. Perrins and Rice [15] extended Mengali's relationship to multi-*h* CPM (the ARTM Tier-2 modulation is a special case) and applied this relationship to derive reduced-complexity demodulators for the Tier-2 modulation [16].

SOQPSK was defined by Hill as a constrained ternary CPM with $h = ½$. [17]. As a consequence, it inherits the close connection to offset QPSK (hence, the name). FQPSK, as defined by Gao and Feher [18], consists of a set of inphase and quadrature waveforms. During each bit interval, the member of the waveform set to be transmitted is selected by the input data. The constraints placed on allowed combinations and sequences of inphase and quadrature waveforms produces a form of offset QPSK with data-dependent pulse shapes. A CPM representation of FQPSK-JR was developed by Nelson, Perrins, and Rice [19] and was the basis for a common detector for FQPSK-JR and SOQPSK-TG. In [4], Weber used the MSK – offset QPSK connection to describe the phase shifts (in the phase trajectory) as positive or negative *frequency* shifts. In this way, differential encoding could be cast in terms of differential frequency shifts rather than the phase shifts used in the body of this paper.



(a) unshaped offset QPSK (b) half-sine offset QPSK (MSK)

(c) SOQPSK-TG (d) FQPSK-JR

Figure 13: Eye diagrams for unshaped offset QPSK (a), half-sine shaped offset QPSK (b), SOQPSK-TG (c), and FQPSK-JR (d).