

Experiment 18 - Quadrature Phase Shift Keying

Preliminary discussion

As its name implies, *quadrature phase shift keying* (QPSK) is a variation of binary phase shift keying (BPSK). Recall that BPSK is basically a DSBSC modulation scheme with digital information for the message. Importantly though, the digital information is sent one bit at a time. QPSK is a DSBSC modulation scheme also but it sends two bits of digital information at a time (without the use of another carrier frequency).

As QPSK sends two bits of data at a time, it's tempting to think that QPSK is twice as fast as BPSK but this is not so. Converting the digital data from a series of individual bits to a series of bit-pairs necessarily halves the data's bit-rate. This cancels the speed advantage of sending two bits at a time.

So why bother with QPSK? Well, halving the data bit rate does have one significant advantage. The amount of the radio-frequency spectrum required to transmit QPSK reliably is half that required for BPSK signals. This in turn makes room for more users on the channel.

Figure 1 below shows the block diagram of the mathematical implementation of QPSK.

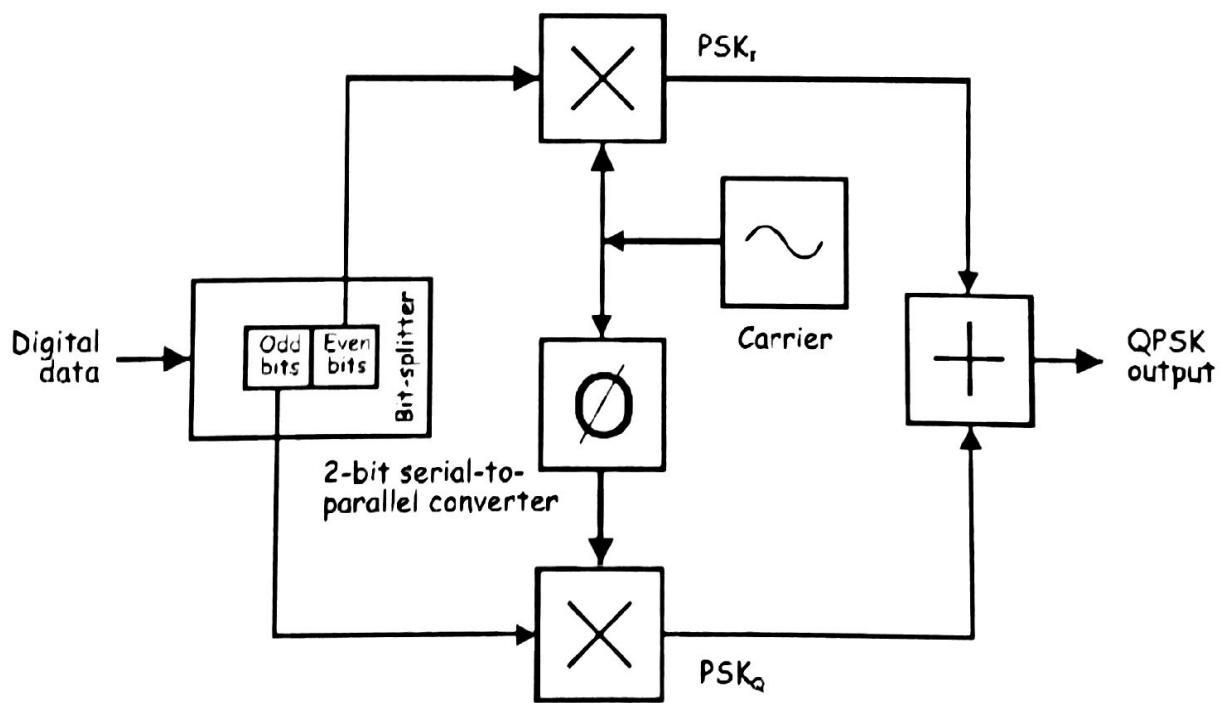


Figure 1

At the input to the modulator, the digital data's even bits (that is, bits 0, 2, 4 and so on) are stripped from the data stream by a "bit-splitter" and are multiplied with a carrier to generate a BPSK signal (called PSK_I). At the same time, the data's odd bits (that is, bits 1, 3, 5 and so on) are stripped from the data stream and are multiplied with the same carrier to generate a second BPSK signal (called PSK_Q). However, the PSK_Q signal's carrier is phase-shifted by 90° before being modulated. This is the secret to QPSK operation.

The two BPSK signals are then simply added together for transmission and, as they have the same carrier frequency, they occupy the same portion of the radio-frequency spectrum. While this suggests that the two sets of signals would be irretrievably mixed, the required 90° of phase separation between the carriers allows the sidebands to be separated by the receiver using phase discrimination (introduced in Experiment 8).

Figure 2 below shows the block diagram of the mathematical implementation of QPSK demodulation.

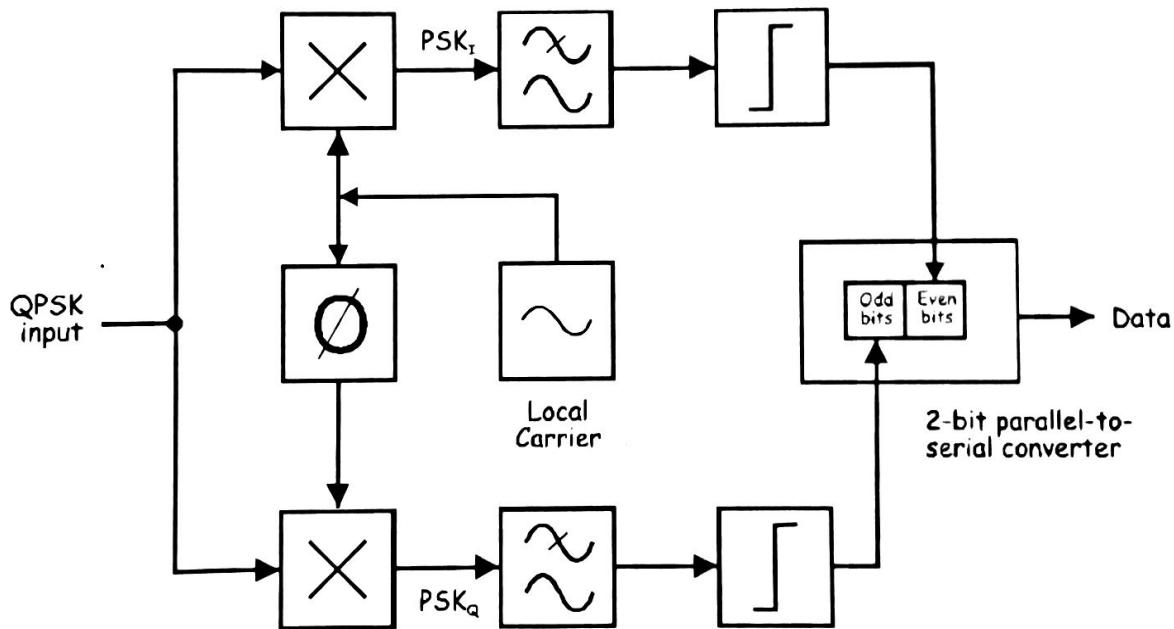


Figure 2

Notice the arrangement uses two product detectors to simultaneously demodulate the two BPSK signals. This simultaneously recovers the pairs of bits in the original data. The two signals are cleaned-up using a comparator or some other signal conditioner then the bits are put back in order using a 2-bit parallel-to-serial converter.

To understand how each detector picks out only one of the BPSK signals and not both of them, recall that the product detection of DSBSC signals is "phase sensitive". That is, recovery of the message is optimal if the transmitted and local carriers are in phase with each other. But the recovered message is attenuated if the two carriers are not exactly in phase. Importantly, if the phase error is 90° the amplitude of the recovered message is zero. In other words, the message is completely rejected (this issue is discussed in Part E of Experiment 7).

The QPSK demodulator takes advantage of this fact. Notice that the product detectors in Figure 2 share the carrier but one of them is phase shifted 90° . That being the case, once the phase of the local carrier for one of the product detectors matches the phase of the transmission carrier for one of the BPSK signals, there is automatically a 90° phase error between that detector's local carrier and the transmission carrier of the other BPSK signal. So, the detector recovers the data on the BPSK signal that it's matched to and rejects the other BPSK signal.

The experiment

In this experiment you'll use the Emona Telecoms-Trainer 101 to generate a QPSK signal by implementing the mathematical model of QPSK. Once generated, you'll examine the QPSK signal using the scope. Then, you'll examine how phase discrimination using a product detector can be used to pick-out the data on one BPSK signal or the other.

It should take you about 1 hour to complete this experiment.

Equipment

- Emona Telecoms-Trainer 101 (plus power-pack)
- Dual channel 20MHz oscilloscope
- three Emona Telecoms-Trainer 101 oscilloscope leads
- assorted Emona Telecoms-Trainer 101 patch leads

Procedure

Part A - Generating a QPSK signal

1. Gather a set of the equipment listed on the previous page.
2. Set up the scope per the instructions in Experiment 1.
3. Set the scope's *Trigger Source* control to the *EXT* position.
4. Set the scope's *Trigger Source Coupling* control to the *HF REJ* position.
5. Set the scope's Channel 1 and Channel 2 *Input Coupling* controls to the *DC* position.
6. Set the scope's *Timebase* control to the *0.5ms/div* position.
7. Locate the Divider module and set it up to divide by 2 by pushing the left-side switch up and the right-side switch down.

Tip: The Divider module is underneath the Sequence Generator module.

8. Connect the set-up shown in Figure 3 below.

Note: Insert the black plugs of the oscilloscope leads into a ground (*GND*) socket.

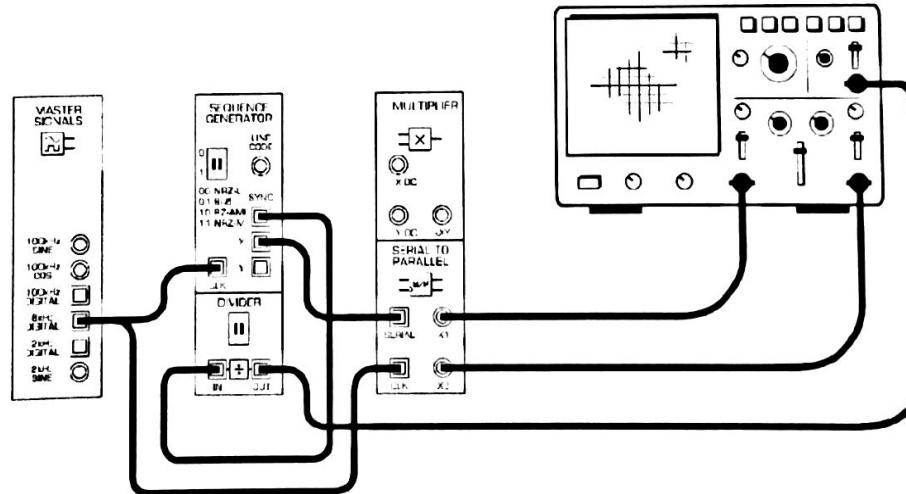


Figure 3

The set-up in Figure 3 can be represented by the block diagram in Figure 4 below. The Sequence Generator module is used to model digital data. The 2-bit Serial-to-Parallel Converter module is used to split the data bits up into a stream of even bit and odd bits.

As mentioned in the preliminary discussion, splitting the data up this way halves the bit rate of the two new sets of data. That being the case, the Sequence Generator module's SYNC output must be halved for it to be used as the triggering signal for the scope.

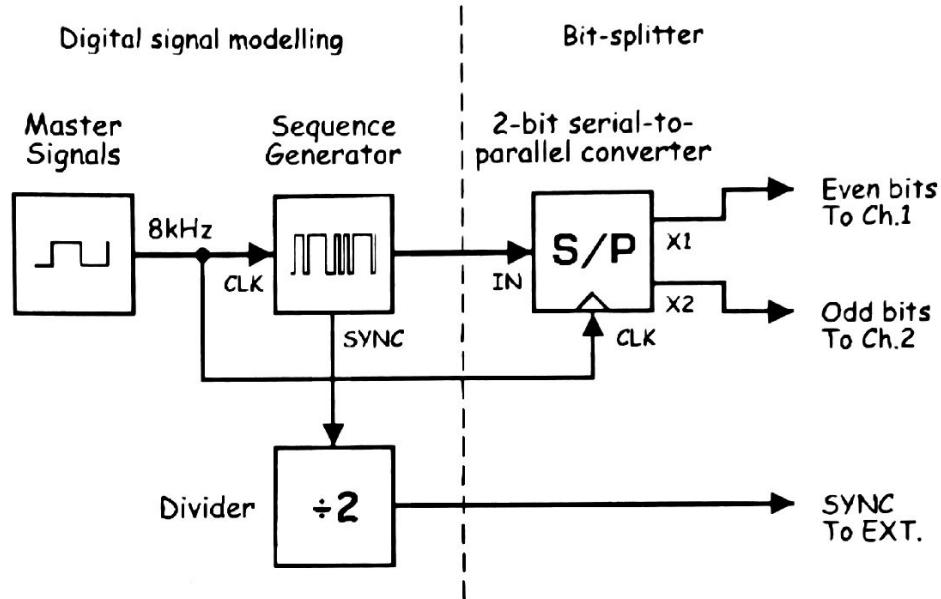


Figure 4

9. Set the scope's Mode control to the DUAL position to view the Serial-to-Parallel Converter module's two outputs.
10. Compare the signals. You should see two digital signals that are different to each other.

Question 1

What is the relationship between the bit rate of these two digital signals and the bit rate of the Sequence Generator module's output? Tip: If you're not sure, see the preliminary discussion.



Ask the instructor to check
your work before continuing.

11. Modify the set-up as shown in Figure 5 below.

Remember: Dotted lines show leads already in place.

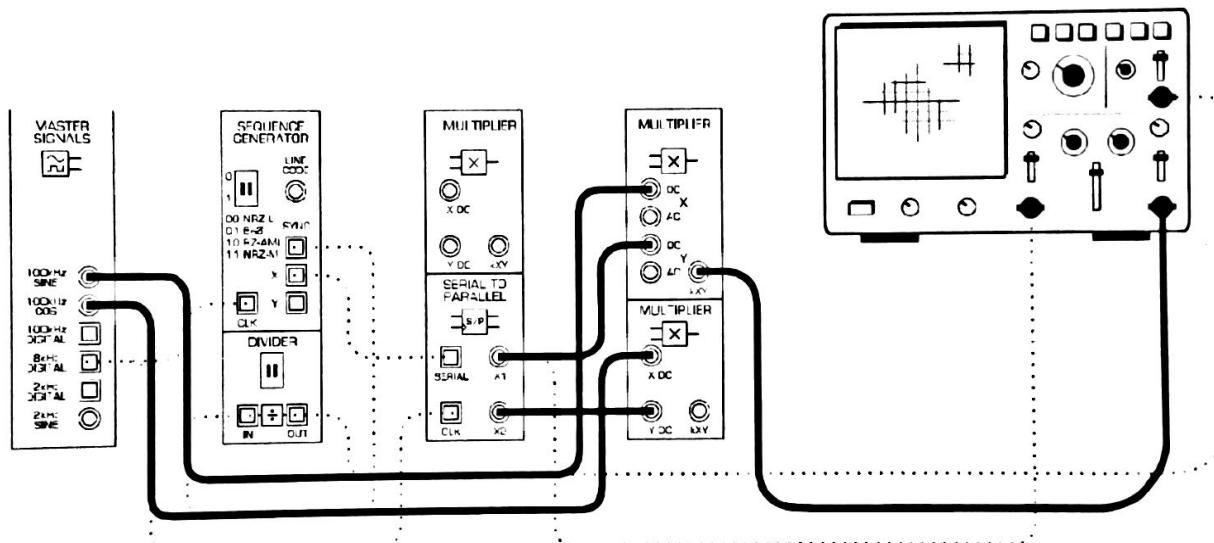


Figure 5

Excluding the digital data modelling, this set-up can be represented by the block diagram in Figure 6 on the next page. Notice that the bit-splitter's two outputs are connected to independent Multiplier modules. The other input to the Multiplier modules is a 100kHz sinewave. However, the signals are out of phase with each other by 90° which is a requirement of QPSK.

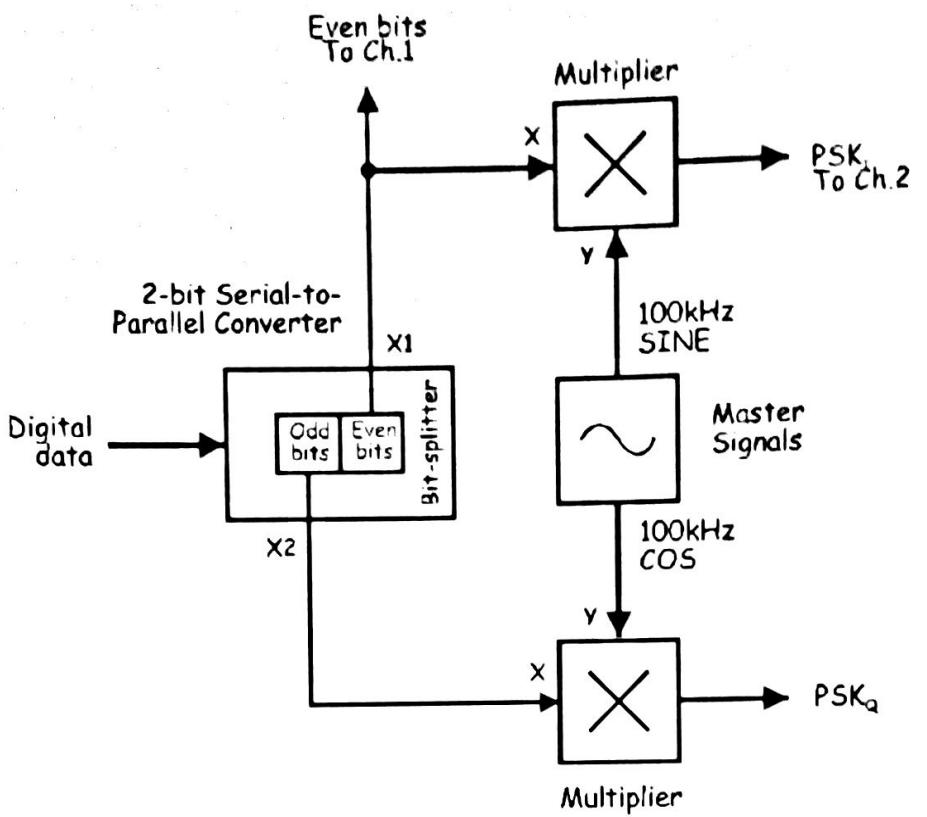


Figure 6

12. Compare the even bits of data with the Multiplier module's output (PSK_I).
13. Set the scope's *Timebase* control to the $0.2ms/div$ position.
14. Activate the scope's *Sweep Multiplier* to view the signals more closely.
15. Use the scope's *Horizontal Position* control to locate a transition in the data sequence.
16. Examine the carrier and note in what way it changes at the sequence's transition.

Question 2

What feature of the Multiplier's output suggests that it's a BPSK signal?

17. Deactivate the scope's *Sweep Multiplier*.
18. Move the scopes connections to the set-up as shown in Figure 7 on the next page.

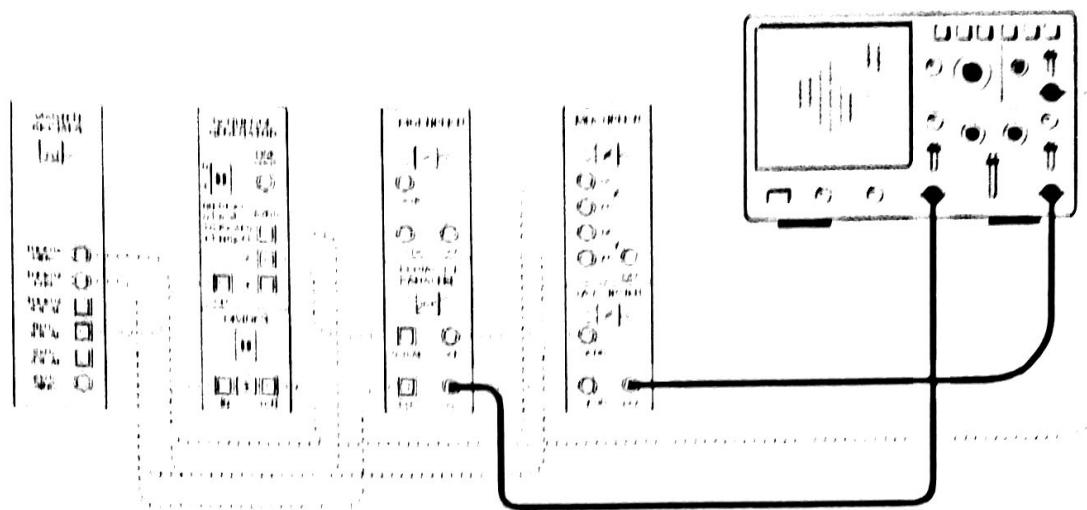


Figure 7

This change can be represented by the block diagram in Figure 8 below.

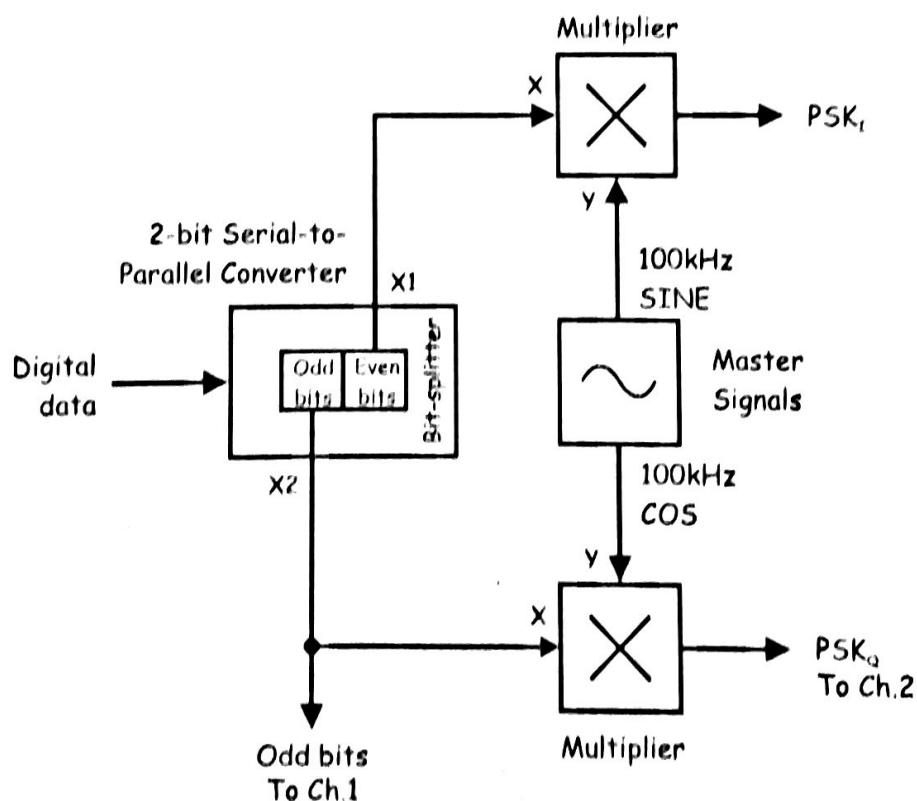


Figure 8

19. Activate the scope's *Sweep Multiplier* to view the signals more closely.
20. Use the scope's *Horizontal Position* control to locate a transition in the data sequence.
21. Examine the carrier and note in what way it changes at the sequence's transition.

Question 3

What type of signal is present on the Multiplier's output?



Ask the instructor to check
your work before continuing.

22. Deactivate the scope's *Sweep Multiplier* and return the scope's *Timebase* control to the *0.5ms/div* setting.
23. Modify the set-up as shown in Figure 9 below.

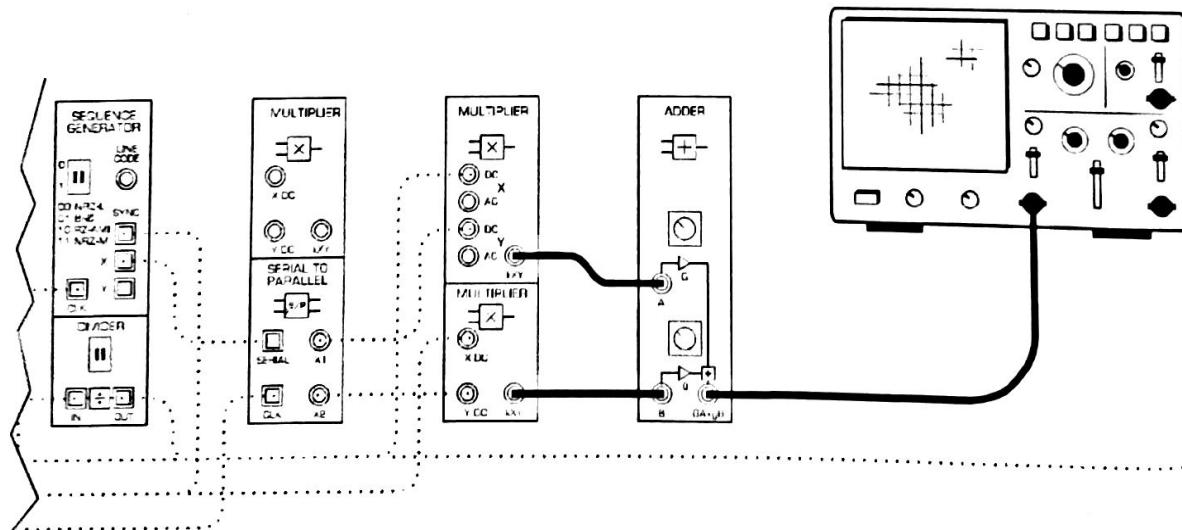


Figure 9

The set-up in Figure 9 can be represented by the block diagram in Figure 10 below. The Adder module is used to add the PSK_I and PSK_Q signals. This turns the set-up into a complete QPSK modulator.

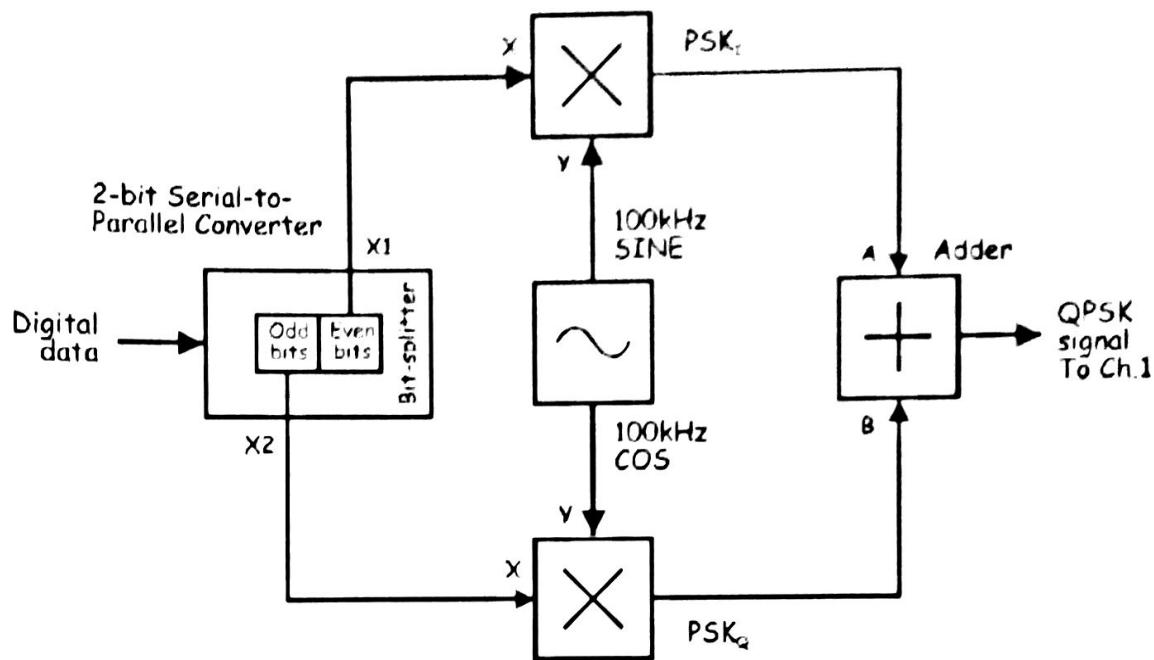


Figure 10

24. Turn the Adder module's *G* control fully anti-clockwise.

Note: This removes the BPSK_I signal from the signal on the Adder module's output.

25. Adjust the Adder's *g* control to obtain a 4Vp-p output.

26. Disconnect the patch lead to the Adder module's *B* input.

Note: This removes the BPSK_Q signal from the signal on the Adder module's output.

27. Adjust the Adder's *G* control to obtain a 4Vp-p output.

28. Reconnect the patch lead to the Adder's *B* input.

Question 4

According to the theory, what type of digital signal transmission is now present on the Adder's output?

29. Set the scope's Timebase control to the 0.2ms/div position
30. Activate the scope's Sweep Multiplier to view the signal more closely
31. Use the scope's Horizontal Position control to examine the signal from beginning to end

Question 5

Why is there only one sinewave when the QPSK signal is made up of two BPSK signals?

QPSK or OQPSK: What's the difference?

QPSK modulation is normally generated from a single data stream converted to two parallel data streams. In this particular experiment, the serial/parallel converter outputs the parallel streams such that the bits are offset from each other by one clock period. Therefore, in this experiment we are actually implementing a form of QPSK known as *Offset QPSK (OQPSK)*.



Ask the instructor to check
your work before continuing.

Part B - Using phase discrimination to pick-out one of the QPSK signal's BPSK signals
 It's not possible to implement both a QPSK modulator and demodulator with one Emona Telecoms-Trainer 101. However, it is possible to demonstrate how phase discrimination is used by a QPSK demodulator to pick-out one or other of the two BPSK signals that make up the QPSK signal. The next part of the experiment lets you do this.

32. Deactivate the scope's *Sweep Multiplier* and return the scope's *Timebase* control to the *1ms/div* setting.
33. Locate the *Tunable LPF* module and turn its *Cut-off Frequency Adjust* control fully clockwise.
34. Set the *Tunable LPF* module's *Gain* control to about the middle of its travel.
35. Locate the *Phase Shifter* module and set its *Phase Change* control to the 0° position.
36. Modify the set-up as shown in Figure 11 below.

Note: As there are a lot of connections, you may find it helpful to tick them off as you add them.

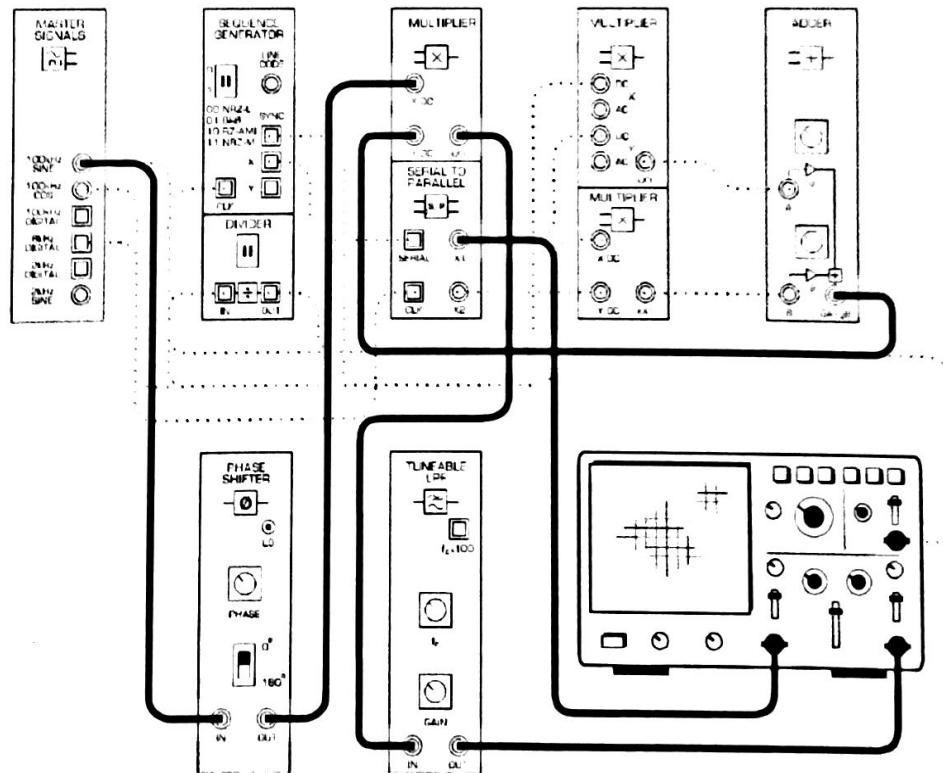


Figure 11

The additions to the set-up can be represented by the block diagram in Figure 12 below. If you compare it to Figure 2 in the preliminary discussion, you'll notice that it implements most of one arm of a QPSK demodulator (either I or Q).

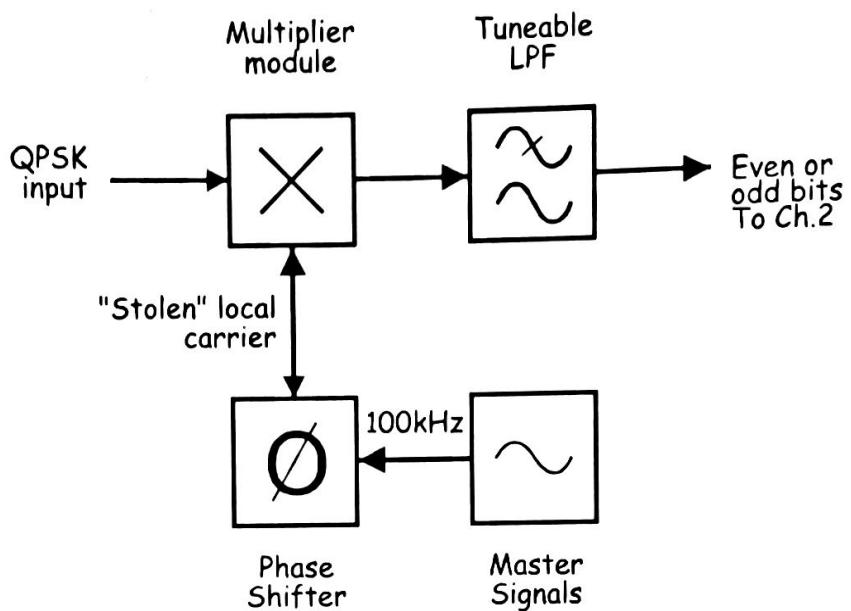


Figure 12

37. Compare the even data bits on the Serial-to-Parallel Converter module's X1 output with the Bandpass LPF module's output.
38. Vary the Phase Shifter module's *Phase Adjust* control left and right and observe the effect on the demodulated signal. You are aiming to recover a bipolar (2-level) signal like the original X1 or X2 signals from the Serial-to-Parallel Converter module.
39. Set the Phase Shifter module's *Phase Change* control to the 180° position and repeat step 36.

Question 6

What is the cause of the 3 and 4 level signals out of the Tunable LPF during the phase adjustments above? How many different *Phase Adjust* control positions will give you a bipolar signal?



Ask the instructor to check
your work before continuing.

40. Modify the set-up as shown in Figure 13 below.

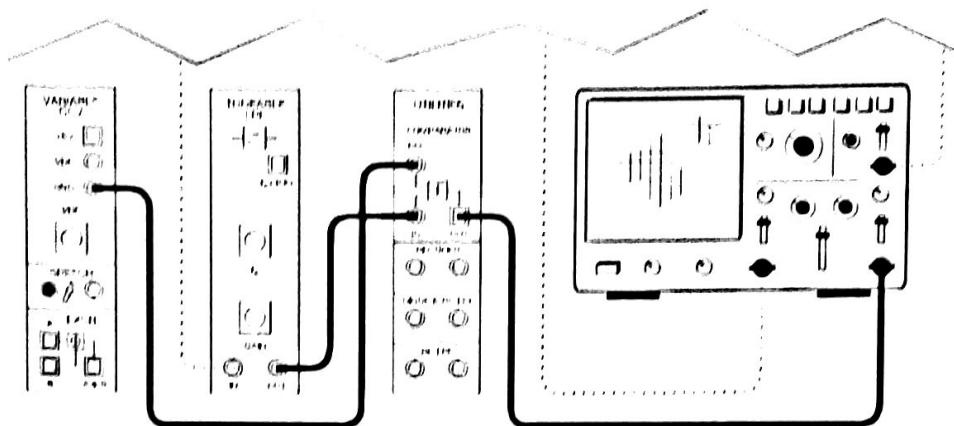


Figure 13

The addition of the Comparator on the Utilities module can be represented by the block diagram in Figure 14 below. If you compare this block diagram with Figure 2 in the preliminary discussion, you'll notice that this change completes one arm of a QPSK demodulator.

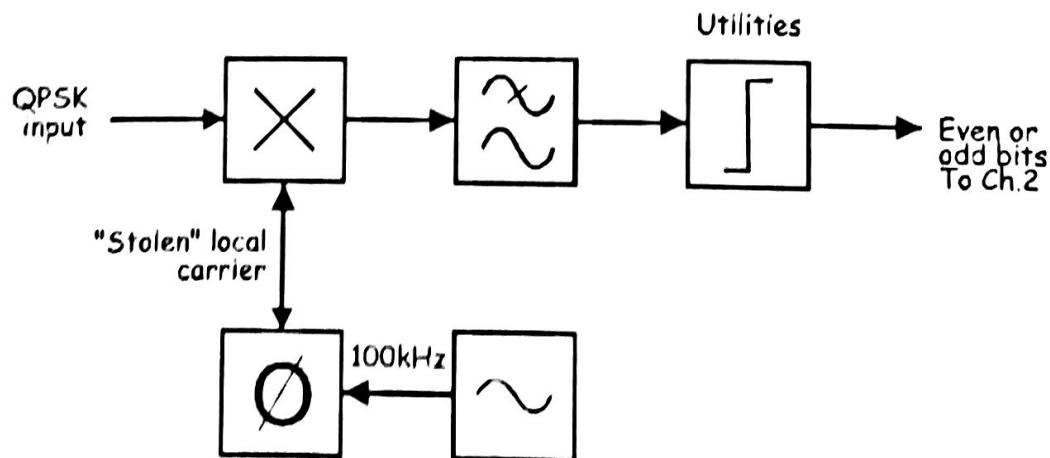


Figure 14

41. Set the Phase Shifter module's *Phase Change* control to the 0° position.
42. Compare the even data bits on the Serial-to-Parallel Converter module's *X1* output with the Bandpass LPF module's output.
43. Adjust the Phase Shifter module's *Phase Adjust* control until you have recovered the even data bits (ignoring any phase shift).

Question 7

What is the present phase relationship between the local carrier and the carrier signals used to generate the PSK_I and PSK_Q signals?



Ask the instructor to check your work before continuing.

44. Unplug the scope's Channel 1 input from the Serial-to-Parallel Converter module's *X1* output and connect it to its *X2* output to view the odd data bits.
45. Compare the odd data bits with the recovered data. They should be different.
46. Set the Phase Shifter module's *Phase Change* control to the 180° position.
47. Adjust the Phase Shifter module's *Phase Adjust* control until you have recovered the odd data bits (ignoring any phase shift).

Question 8

What is the new phase relationship between the local carrier and the carrier signals used to generate the PSK_I and PSK_Q signals?

Question 9

Why is your demodulator considered to be only one half of a full QPSK receiver?



Ask the instructor to check
your work before finishing.