

SHAFT ENCODER PERIPHERAL INTERFACE IXSE502 / IXSE503

Features

- Direct (2 channel) Quadrature Inputs with Schmitt Trigger Circuitry
- x4 Quadrature Detection for High Resolution Counting
- EXTERNAL EVENT Detected and Latched For Absolute Positioning Needs
- Quadrature Pulse Rate up to a Frequency of CLOCK/2
- Easily Interfaces to Most Microprocessors and Microcomputers
- INDEX Pulse Decoding and Sampling into a Dedicated Latch
- Non-Intrusive Counter Latching
- 8-bit Counter with Automatic Reset on Read provides Reliable Incremental Information.
- Low Power High Speed LSI CMOS
- High Noise Immunity
- Digital Filtering of the PHASE and INDEX Input Signals
- 16-Bit Counter by Cascading Two IXSE503

Applications

- AC and DC Servo Systems
- Plotters and Printers, and Other Computer Peripherals
- Other Motion Control
- X-Y Tables and Positioning Systems
- Optical Sensing Instrumentation
- Rotary Pulse Generator to Microprocessor Interface
- Robotics

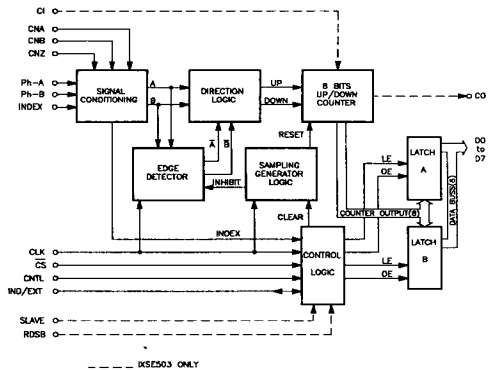
Description

IXSE502 and IXSE503 Shaft Encoder Peripheral Interface (SEPI) are CMOS LSI devices designed to interface a standard two-phase incremental encoder (either linear or rotary) with a microprocessor or microcomputer. The IXSE502 accepts two-phase incremental encoder signals directly, performs the quadrature detection, and counts the number of incremental encoder pulses in an internal 8-bit up/down counter. The IXSE503 provides the option to increase the counter to a 16-bit or larger up/down counter by cascading 2 or more IXSE503's. A separate hardware latch latches the exact count in the 8-bit up/down counter when either an INDEX pulse or an EXTERNAL EVENT pulse occurs, providing a means for very accurate absolute positioning.

Direction detection and up/down count accumulation result in a 2's complement number that is directly readable by the host microprocessor. The information provided by the IXSE502/503 aids in calculating the acceleration, velocity, and position of an incremental encoder; thus significantly reducing microprocessor overhead. Reduction of microprocessor overhead is particularly important in real-time motion control applications, such as servo motor velocity and position control systems.

The hysteresis built into the encoder inputs, together with digital filtering of the incoming data, provides very reliable electrical conditioning and high noise immunity, especially in electrically noisy industrial environments.

Block Diagram



Absolute Maximum Ratings (above which useful life may be impaired)							
Supply Voltage	-0.3 to +7.0 V						
Input Voltage (1)	-0.3 to +7.0 V						
Output Voltage	-0.3 to +7.0 V						
Maximum Power Dissipation	500 mW						
Storage Temperature Range	-55 to +150°C						
Recommended Operating Range				Commercial			
Parameter	Description	Min	Max	Unit			
V _{CC}	Supply Voltage	4.5	5.5	V			
T _A	Commercial Operating Temperature	0	70	°C			
T _A	Industrial Operating Temperature	-40	85	°C			
DC Characteristics (V _{CC} = 5V; T _A = 25°C)							
Parameter	Description	Applies To	Test Condition	Min	Typ	Max	Unit
V _{T+}	Schmitt Trigger Positive Threshold	Ph-A, Ph-B, INDEX		3.8	2.8		V
V _{T-}	Schmitt Trigger Negative Threshold	Ph-A, Ph-B, INDEX			2.2	1.3	V
V _H	Input Hysteresis	Ph-A, Ph-B, INDEX	V _{CC} = 5 V	0.3	0.6		V
V _{IH}	Input High Voltage	Note 1		2.0		V _{CC} + 0.3	V
V _{IL}	Input Low Voltage	Note 1				0.8	V
V _{OH}	Output High Voltage	DO - D7, CO	I _{OH} = -2mA	2.4			V
V _{OL}	Output Low Voltage	DO - D7, CO	I _{OL} = 4mA			0.4	V
I _{LI}	Input Leakage Current	Note 2	0 ≤ V _I ≤ V _{CC}	-10		10	μA
I _{LO}	Output Leakage Current	DO - D7	0 ≤ V _I ≤ V _{CC}	-10		10	μA
V _{OL}	IND/EXT Low Output Volt	IND/EXT	I _{OL} = 4mA			0.4	V
I _{PU}	IND/EXT Pull-Up Current	IND/EXT	V _O = 0V	-200		-30	μA
I _{CC}	Power Supply Current		f _{clk} = 4MHz; V _{IL} = 0, V _{IH} = V _{CC}		0.5	10	mA

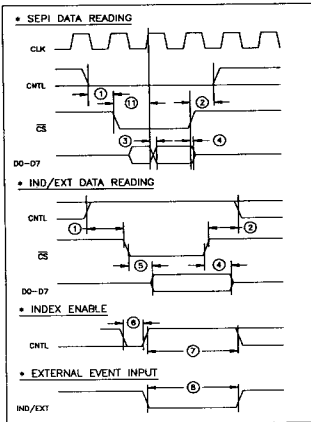
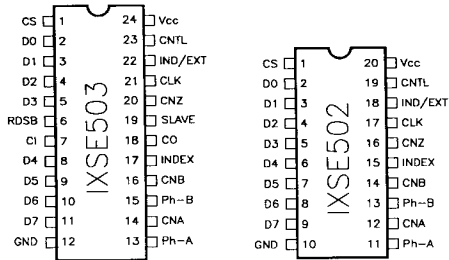
Note 1: These values apply to CS, CNTL, IND/EXT, CLK, CI, RDSB, SLAVE
 Note 2: These values apply to CS, CNTL, CLK, CI, RDSB, SLAVE

AC Characteristics (over operating range)

No.	Parameter	Description	Min	Typ	Max	Unit
1	tCNVCSL	CNTL to CS Low, Setup Time	10			ns
2	tCSHCNX	CNTL after CS High, Hold Time	10			ns
3	tCHDV	Latch A Data Valid after CLK High ¹		35	150	ns
4	tCSHDX	Data Valid after CS High	0	15	60	ns
5	tCSLDV	Latch B Data Valid after CS Low ¹		20	150	ns
6	tCNVCNH	CNTL Low to INDEX Enable, Setup Time	100			ns
7	tCNHCNX	CNTL High after INDEX Enable, Hold Time	100			ns
8	tELEX	EXT Low After EXTERNAL EVENT, Hold Time	100			ns
9	fCLK	CLK (Clock) Frequency		6	12	MHz
10	fA, fB	Ph-A and Ph-B Frequency ²		.5	1	MHz
11	tCSLCH	CS Low to CLK High, Setup Time	10			ns

Note 1: Test Condition is with a 100pF load

Note 2: Test Condition is f_A, f_B less than or equal to f_{CLK}/12

Timing Diagram

Pin Configuration


PIN DESCRIPTION

502 Pin	503 Pin	Symbol	Description
20	24	VCC	VOLTAGE SUPPLY - +5 volts \pm 10%.
10	12	GND	CIRCUIT GROUND - 0 Volts
1	1	CS	CHIP SELECT - When CNTL is "low", a "low" on this input loads data into LATCH A and enables the data onto the external data bus, pins D0 through D7. When CNTL is "high", a "low" on this input enables the data in LATCH B onto the external data bus.
19	23	CNTL	CONTROL - (A-B select) a "low" on this input selects Latch A for reading. A "high" on this input selects Latch B for reading. For proper operation CNTL must be stable before CS goes "low", for the duration of CS "low", and after CS goes "high". This input also determines how Latch B will be triggered. The Latch B Trigger has three states: triggered, armed for trigger on the IND/EXT input only, and armed for trigger by either the IND/EXT input or the INDEX input. Once Latch B has been triggered, additional trigger pulses will not cause any latching to occur. Latch B must be read to allow it to be triggered again. Reading Latch B (CNTL "high" and CS "low") arms Latch B for trigger by the IND/EXT input only. If Latch B has not been triggered a low-to-high transition on the CNTL input will arm Latch B for trigger on either the IND/EXT input or the INDEX input.
11 13	13 15	Ph-A, Ph-B	PHASE A and PHASE B - these Schmitt trigger inputs accept the quadrature signals from either a rotary or linear incremental encoder.
12 14	14 16	CNA CNB	CONTROL A and CONTROL B - These inputs control the effective polarity of PH-A and PH-B. Low level on these inputs will pass the phase inputs as presented on their inputs. High logic level will complement the corresponding phase input signal.
15	17	INDEX	INDEX - (sometimes called CHANNEL Z or MARKER PULSE) - this Schmitt trigger input has electrical characteristics identical to the PH-A and PH-B inputs. A low level signal on this input when CNZ, PH-A, CNA, PH-B and CNB are also low, will trigger Latch B if Latch B is armed to trigger on the INDEX input. If CNA, CNB and/or CNZ are high, the logic condition for the INDEX detection will change accordingly.
16	20	CNZ	CONTROL CHANNEL Z - This input controls the polarity of the INDEX input. Low level on this input will pass the INDEX input as presented. High logic level will complement the INDEX input signal.
17	21	CLK	CLOCK - external TTL clock.
2-5,6, 7,8,9	2-5,8, 9,10,11	D0-D3, D4,D5, D6,D7	DATA BUS - these outputs provide access to the Latch A and Latch B data. The data is available as long as CS is low. In the IXSE503 which is configured as a SLAVE, the data will be available only upon a negative signal at pin 6 (RDSB).
18	22	IND/EXT	INDEX/EXTERNAL EVENT - this pin is always active as both an input and an output. It should be sensed with a Schmitt trigger input and driven by an open-collector output. A high-to-low transition on this input when Latch B is armed causes the up/down counter's contents to be transferred to Latch B. The counter's contents cannot be transferred again to Latch B until Latch B has been read. Reading Latch B arms it for trigger on the IND/EXT input. When the counter's contents are transferred to Latch B, as a result of the INDEX input, this output generates a negative going pulse. The width of this pulse is half the clock period. When the EXTERNAL EVENT triggers Latch B, the source of this signal should be sent to the controlling microprocessor as well.
—	7	CI	CARRY IN - this signal should be wired to the 'CO' output of it's master when used in a cascading mode. If the IXSE503 is used as first master in a string or as a single 8-bit counter, this input pin can be left open or tied to GND (there is an internal pull down). A high (logic "1") on this input will disable the counter from advancing. A low (logic "0") on this input will enable the counter.
—	18	CO	CARRY OUT - this output indicates that the counter is on maximum state (0FFH), and it will overflow (go to 00H) during the next state. The 'CO' should be connected to the 'CI' of the successive cascaded IXSE503.
—	19	SLAVE	SLAVE - The logic level provided at this input pin selects the mode of operation of the IXSE503. A low ("0") or open will set the IXSE503 to the MASTER mode. A high ("1") will set it to SLAVE mode. Only the first IXSE503 in a chain should be set as MASTER.
—	6	RDSB	READ SLAVE - This input pin is used to access and read the data from an IXSE503 which is configured as a SLAVE device. When CNTL is low ("0"), a negative signal on RDSB will enable the output data buffers and present LATCH-A data (position count) on the bus. When CNTL is high ("1"), LATCH-B data (index position) will be directed to the output bus, and the IXSE503 will be armed for IND/EXT input exactly as in the IXSE502.

GENERAL FUNCTIONAL DESCRIPTION

The IXSE502 and IXSE503 are Shaft Encoder Peripheral Interface (SEPI) designed with monolithic CMOS technology. A single +5 volt supply and TTL level clock are required for operation. An internal 8-bit Up/Down Counter continuously provides the number of quadrature edges (encoder counts) in a two's complement binary format. Figure 1 is a block diagram of the IXSE502/503, it combines the quadrature detection circuitry and up/down counter of a typical incremental encoder feedback system into one chip. It includes the added feature of a latch on INDEX or EXTERNAL EVENT to aid in encoder error detection and accurate edge detection for "GO HOME" functions.

The IXSE502/503 can be viewed as a microprocessor peripheral with two read-only registers. Latch A and Latch B. Both latches are incremental position registers. Latch A tracks the position of the incremental encoder and is the register of primary interest. Latch B is used in conjunction with the latch on INDEX and EXTERNAL EVENT features. Both registers sample the internal Up/Down Counter which is always counting quadrature edges.

The IXSE502/503 consists of primarily two sections, described briefly below: the Signal Processing Logic and the Control and Output Buffer Logic.

Signal Processing Logic

The Signal Processing Logic is the center of quadrature detection and up/down counting. The Signal Processing Logic is represented by the Signal Conditioning, Direction Logic, Edge Detector, Sampling Generator Logic, and 8-bit Up/Down Counter blocks in figure 1.

Outputs from either a rotary or linear incremental encoder are connected directly to the Schmitt trigger phase inputs. These inputs are designed to provide maximum noise immunity by use of an internal digital filter.

Control and Output Buffer Logic

The Control and Output Buffer Logic consists of the Control Logic, Latch A, and Latch B blocks in figure 1. This logic is responsible for transferring data from the 8-bit Up/Down Counter to Latch A or Latch B and for driving the contents of the latches onto the Data Bus. When CS is low, data is transferred from the selected latch to the Data Bus. The CNTL input

determines which of the two latches is selected.

The IND/EXT line can be used as both an input and an output. An open-drain configuration, IND/EXT is pulled up internally by a resistor of approximately 50K Ω . This signal should be sensed with a hysteresis input and driven by an open-collector output. A low pulse on the IND/EXT line latches data into Latch B.

DETAILED DESCRIPTION

Latch A

This register is the incremental position register. It returns the change in position since the last read of Latch A. Each time Latch A is read the IXSE502/503's internal Up/Down Counter is reset to zero. The reset mechanism in the IXSE502/503 guarantees that no encoder pulses are left uncounted, or lost. Because Latch A returns an incremental position it can be used to quickly detect velocity, as well as position.

Reading Latch A

Figure 2 shows the Latch A read timing. The sequence of events required to read Latch A is as follows:

- 1) Set the CNTL input "low"
- 2) Set the \overline{CS} input "low"
- 3) Wait for the first low-to-high transition of CLK
- 4) Read the data on the data lines (D0 through D7)
- 5) Set the \overline{CS} input "high"

Reading Latch A is nearly identical to a typical microprocessor read except for one step, waiting for the low-to-high transition of the CLK input. The incremental nature of the IXSE502/503 requires that the CNTL and \overline{CS} inputs be active for at least one rising edge of CLK. The data being read will be available after the first rising edge of CLK (allowing for propagation delays). Thus, the time required to read Latch A is dependent on the frequency of CLK.

The IXSE502 data bus, D0 through D7, will drive the microprocessor's data bus lines as soon as CS is asserted (allowing for propagation delays). This is true even if the first rising edge of CLK has not yet occurred. During the first rising edge of CLK, following the assertion of CNTL and CS, the contents of the internal Up/Down Counter are transferred to

Latch A and the Up/Down Counter is reset to zero. After that first rising edge of CLK additional CLK edges can occur; these additional edges will have no effect on the state of the data being driven onto the data bus or on the contents of the internal Up/Down Counter. In asynchronous operation (when the IXSE502's CLK is not synchronized to processor read cycles), the worst case access time is one CLK period plus t_{CHDV} . If CLK is synchronized to processor read cycles the worst case access time will usually be shortened.

Three different types of clock edges are denoted in figure 2: the rising CLK edge that causes data to be transferred from the Up/Down Counter to Latch A (latching edge), the falling CLK edge that resets the Up/Down Counter (reset edge), and the falling CLK edges that sample the PH-A and PH-B encoder signals (sampling edges). The CLK edge that resets the counter always follows the CLK edge that transfers the counter's contents to Latch A. The Up/Down Counter's contents will not be transferred to Latch A and the counter will not be reset if a rising CLK edge does not occur when CNTL and CS are asserted. In that case the data in Latch A will be old data that was latched during the most recently successful read. Latch A is only refreshed on the first rising CLK edge following assertion of CNTL and CS. Failure to insure the occurrence of CLK during a Latch A read will result in incorrect information being read by the microprocessor.

Each time the Latch A register in the IXSE502 is read, the internal 8-bit Up/Down Counter is reset. This reset occurs on the falling edge of CLK which follows the rising edge that transferred data from the Up/Down Counter to Latch A (see figure 2). Like the "latching edge" of CLK, there is only one "reset edge" per read operation. Those falling edges of CLK that are not "reset edges" are "sampling edges". Normally, the PH-A and PH-B inputs are sampled on each falling edge of CLK. The only time this is not true is when Latch A is read and the Up/Down Counter is reset. When a "reset edge" occurs, the quadrature detection clock is suspended for one CLK cycle. The nature of the quadrature detection state machine guarantees that no encoder counts are sent to the Up/Down Counter during the suspended clock and no encoder counts are lost by the reset process.

The internal quadrature detection logic normally requires that a quadrature state persist for at least two CLK periods. However, because a "reset edge" suspends the quadrature detection clock for one CLK period a quadrature state must persist for at

least three CLK periods during a read of Latch A (when a reset of the Up/Down Counter occurs).

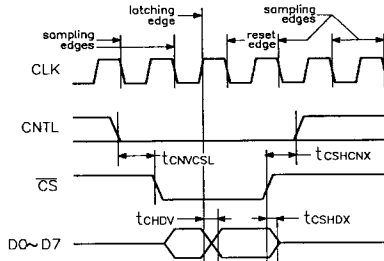


Fig. 2: Latch A Read Cycle Timing

Latch B

This register holds the position that was in the IC's internal Up/Down Counter when a Latch B trigger occurred. Latch B can be triggered by either an INDEX pulse from the incremental encoder or an EXTERNAL EVENT on the IND/EXT input. This latch can be used to detect encoder errors (by sensing the position of the once per revolution index pulse) or it can be used to provide an accurate HOME position for "GO HOME" functions.

Reading Latch B

Figure 3 shows the Latch B read timing. Reading Latch B is representative of a typical microprocessor system read. The sequence of events required to read Latch B is as follows:

- 1) Set the CNTL input "high"
- 2) Set the \overline{CS} input "low"
- 3) Read the data on the data lines (D0 through D7)
- 4) Set the \overline{CS} input "high"

As is apparent from the timing diagram in figure 3, reading Latch B is not dependent on the CLK input. This is different than reading Latch A. When reading Latch B, data is available as soon as the CNTL and CS inputs are asserted (allowing for the appropriate propagation delays). As long as CNTL and CS are asserted, the contents of Latch B will be available on

the data bus. The information in Latch B is guaranteed not to change while CNTL and \overline{CS} are asserted.

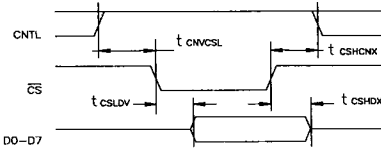


Fig 3: Latch B Read Cycle Timing

Latch B Trigger

Data can be transferred to Latch B in only two ways:

- 1) a negative going edge on the IND/EXT input
- 2) on the INDEX input detection.

CNTL and \overline{CS} determine which input, INDEX or IND/EXT, will cause data to be transferred from the 8-bit Up/Down Counter to Latch B. The Latch B trigger can be in one of three states: armed for trigger by an IND/EXT pulse only, armed for trigger by either an IND/EXT pulse or an INDEX pulse, or triggered (either an IND/EXT pulse or an INDEX pulse has transferred data to Latch B).

The various states controlled by IND/EXT, INDEX, CNTL, and \overline{CS} are defined as follows:

"FREE" -- armed for trigger on an IND/EXT pulse only

"SET" -- armed for trigger on either IND/EXT or INDEX

"LOCK" -- triggered by either IND/EXT or INDEX

These are the only states the Latch B trigger circuitry can be in. The CNTL and \overline{CS} inputs control movement to the "FREE" state and the "SET" state. The IND/EXT and INDEX inputs control movement to the "LOCK" state. The state of the Latch B trigger has no bearing on one's ability to read Latch A. Latch A can be read regardless of the state of the Latch B trigger.

The "FREE" state and the "SET" state are the only two states from which counter data can be transferred to Latch B (a "Latch B trigger"). When a Latch B trigger occurs, the Latch B trigger circuitry goes

immediately to the "LOCK" state. Notice that there is no state that arms Latch B for trigger on the INDEX input only. If the IXSE502 is armed for trigger on the INDEX input it is also armed for trigger on the IND/EXT input. The converse is not true, if the IXSE502 is armed for trigger on the IND/EXT input it is not necessarily armed for trigger on the INDEX input.

After data has been transferred to Latch B, additional pulses on either the INDEX or IND/EXT input are ignored. Reading the contents of Latch B always moves the Latch B trigger to the "FREE" state so that only pulses on the IND/EXT input will transfer data from the 8-bit Up/Down Counter to Latch B. A low-to-high transition on the CNTL input, when the Latch B trigger is in the "FREE" state, places the Latch B trigger in the "SET" state. A high-to-low transition on the CNTL input does not change the Latch B trigger state. That is, CNTL does not need to remain high for the Latch B trigger to remain in the "SET" state. Only a trigger pulse (on the IND/EXT input or the INDEX input) or reading the Latch B register will move the Latch B trigger from the "SET" state to another state.

CLK Input

CLK is a TTL compatible input that drives all the internal state machines, such as the quadrature detection logic and the "reset on read" logic. There is no minimum frequency, however, there is a maximum clock frequency. At the maximum frequency the clock must have a 50% duty-cycle. Frequencies less than the maximum frequency do not need to have a 50% duty-cycle, although their high-times and low-times must be greater than or equal to the high-time and low-time at the maximum frequency clock.

The considerations for selecting the clock frequency are as follows:

- Available clock in the system.
- The Latch A read access time is dependent on the frequency of CLK. If CLK is synchronous with respect to the microprocessor's read cycles, the worst case access time will be one CLK period plus tCHDV.
- Noise immunity is a function of clock period. Lower frequency clock provides better noise blockage.
- With an ideal shaft encoder (no phase

IXYS Corporation • 2355 Zanker Rd. San Jose, CA 95131-1109 • TEL: (408)435-1900 • FAX: (408) 435-0670

ABB-IXYS Semiconductor GmbH • POB 1180 • D-6840 Lampertheim, Germany • TEL: +49-6206-5030 • FAX: +49-6206-503627 • TLX: 662422602 abt

error), the minimum clock frequency should be 12 times the maximum channel input frequency to guarantee proper quadrature detection. For example, a maximum input frequency of 500KHz will require a minimum clock of 6MHz.

- The maximum phase error of a shaft encoder increases the minimum clock frequency requirement by the following ratio: $90^\circ / (90^\circ - \text{phase error} [^\circ])$
- It has been determined that a typical clock frequency of 4 MHz guarantees count integrity for most applications and provides an asynchronous maximum access time of 400ns. 4 MHz is easily generated from most system clocks.

CNZ	INDEX	CNA	CNB	PH-A	PH-B
0	0	0	0	0	0
0	0	0	1	0	1
0	0	1	0	1	0
0	0	1	1	1	1
1	1	0	0	0	0
1	1	0	1	0	1
1	1	1	0	1	0
1	1	1	1	1	1

Fig 4: INDEX Trigger Logic Combinations

PH-A, PH-B, CNA, CNB, INDEX and CNZ Inputs

PH-A, PH-B, and INDEX are the interface points between the IXSE502/503 and the incremental encoder. PH-A and PH-B are the two-phase quadrature inputs and INDEX is the once per revolution marker pulse input. All three inputs are Schmitt trigger inputs with a 0.6V typical hysteresis. In addition, all three signals are filtered by a 2 stage digital filter. Each of those inputs has a control input associated with it. The respective control input determines the polarity of the input signal. The logic state of CNA and CNB will determine the direction of the internal counter advance in relation to the quadrature input signal.

Figure 5 is an illustration of the PH-A, PH-B, and INDEX timing. The timing shown will cause the internal Up/Down Counter to increment each time a quadrature edge is detected. Reversing the PH-A and PH-B sequence will cause the Up/Down Counter to decrement each time a quadrature edge

is detected.

When CNZ is low, the IXSE502/503 expects the INDEX signal to be an active low pulse. If your encoder generates an active high pulse on its INDEX output, CNZ input should be high. This will invert the INDEX signal so that the IXSE502/503 sees the proper polarity. As shown in figure 5, the INDEX signal triggers the internal circuitry upon the quadrature edge following the selected logic combination of PH-A and PH-B inputs. This logic combination is determined by the control inputs as shown in the table of figure 4. This truth table depicts the 8 logic states of the input signals that will produce an INDEX trigger. Figure 5 represents the timing diagram for the first combination of the truth table.

Figure 5 defines three time periods that need to be considered when choosing the CLK frequency. The most important time period is "tSTATE". The other two time periods, "tCYCLE" and "tINDEX" are defined when "tSTATE" is defined; "tSTATE" is the minimum quadrature state time period. In order to guarantee proper detection of the quadrature states, "tSTATE" must be at least two CLK periods wide. One should take into account encoder signal tolerances when determining "tSTATE". "tINDEX" is the signal that will be available at the IND/EXT pin for microprocessor interrupt purposes, upon detection of an INDEX input pulse.

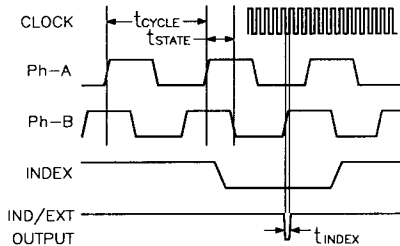


Fig 5: Encoder Inputs Timing Diagram

The polarity of the 2's complement incremental position output, and thus the up/down counter direction, is determined by the relative phase of the quadrature input signal (PH-A and PH-B) and the logic levels of the control inputs (CNA and CNB).

Case 1: If PH-B leads PH-A (as shown in figure 5), a positive count will be achieved by connecting CNA and CNB to opposite logic levels (CNA="0" and CNB="1" or CNA="1" and CNB="0"). A negative

count will be achieved by connecting both control inputs to the same logic level (CNA="1" and CNB="1" or CNA="0" and CNB="0"). **Case 2:** if PH-A leads PH-B, as shown in figure 7, the count direction will be opposite to case 1.

Figure 6 shows the phase input circuit as implemented in the IXSE502/503 and a sample circuit configuration for direct interface with a single ended optical shaft encoder.

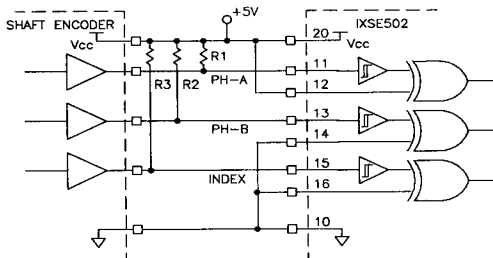


Fig 6: Single Ended Direct Input Connection

figure, "tCYCLE" and "tSTATE". There are always four tSTATE's for each tCYCLE. Each tCYCLE represents one "line" on the incremental encoder, or 360 electrical degrees. The inverse of tCYCLE is the encoder's line rate or line frequency. Each tSTATE represents one encoder pulse. The direction of the encoder pulse is determined by comparing the levels in the previous state to the levels in the current state. Thus, quadrature detection consists of detecting a change in quadrature states and noting the direction of the state change.

What is Quadrature Detection and How does it Work?

The signals generated by an incremental encoder are generally two identical periodic waveforms that are "in quadrature". By detecting the "quadrature states" one can determine both position and direction. Although many quadrature states can be defined (if the two periodic waveforms are continuous, or analog, in nature) only four are normally decoded. The IXSE502/503 internally identifies these four quadrature states and translates them into encoder counts and direction for the internal Up/Down Counter.

Quadrature detection refers to the process of detecting changes in the state of the PH-A and PH-B inputs. PH-A and PH-B are said to be "in quadrature" if there is a 90 degree phase difference between the two signals. Figure 7 represents the output of an ideal incremental encoder (either rotary or linear). This waveform is generated by moving the encoder at a constant velocity; the frequency of the waveform is proportional to the speed of the encoder.

Notice that two time periods are defined in the

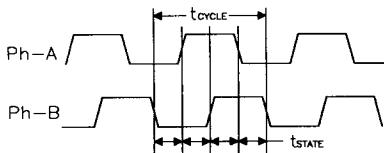


Fig 7: Ideal Encoder Output

Figure 8 represents the output of a real incremental encoder. In figure 7 each quadrature state is equal in size; 90 electrical degrees. The states in figure 8 are not equal in size, this is due primarily to inaccuracies in the encoder's line pattern and the electronics that sense the line pattern.

Most encoder manufacturers publish a number known as either the "quadrature state error" or the "quadrature alignment tolerance", usually stated in electrical degrees. As noted from figure 7, the ideal quadrature state size is 90 degrees. Typical encoders usually have worst case state errors of less than 45 degrees; thus, the minimum worst case state size would be 45 degrees. Some encoders can have a

worst case state error of 60 degrees, meaning the minimum worst case state size would be 30 degrees, less than half the ideal state size!

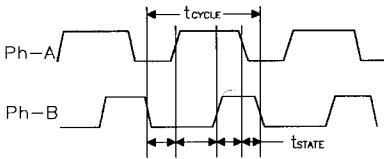


Figure 8: "Real" Encoder Output

APPLICATIONS INFORMATION

Relationship between the Encoder Rate and the Read Rate

The read rate is the frequency at which one reads the Latch A register. Latch A must be read often enough to guarantee the 8-bit Up/Down Counter does not overflow between reads. If the counter overflows, the number read from Latch A will be too small. Adding this incorrect number to an absolute position register will result in calculating an incorrect absolute position.

The rate at which the Latch A register must be read is a function of the highest expected line rate. Since each encoder line is divided into four quadrature states, and each quadrature state corresponds to one encoder count, the maximum count rate will be four times the maximum expected line rate. The Up/Down Counter accumulates an 8-bit two's complement number. Therefore, the counter will overflow if more than 127 counts accumulate in one direction. The only occasion during which this might occur is at maximum speed.

To determine what the read rate must be in order to avoid an overflow of the Up/Down Counter use the following formula:

$$\text{update rate} = \frac{(\text{max line rate}) \cdot 4}{127}$$

For example, if we are using a 1000 line rotary encoder (4000 encoder counts per revolution after quadrature detection) and the maximum encoder speed is 1800 rpm, the maximum line rate will be $(1800/60) \cdot 1000 = 30 \text{ kHz}$. Therefore, the minimum

update rate must be 945 times per second. The maximum time between reads at the maximum speed is the inverse of the minimum update rate; in this example the update period needs to be less than or equal to 1.06 milliseconds.

To determine a rotary encoder's maximum allowable speed in RPM as a function of the update rate (read rate in samples per second), use the following formula:

$$\text{encoder speed} = \frac{127 \cdot (\text{update rate}) \cdot 60}{4 \cdot (\text{lines per revolution})}$$

There is no relationship between the frequency of CLK and the rate at which one must read Latch A, even though both are determined by the maximum line rate. The CLK frequency is determined by the smallest time period in one quadrature state. The update rate is determined by the size of the Up/Down Counter. The limit, in both cases, is determined by the maximum line rate.

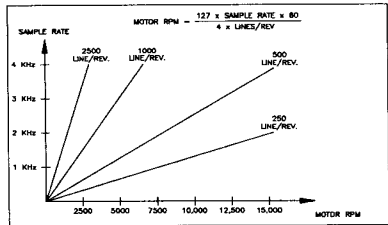


Figure 9: Motor Speed vs. Sample Rate - 8-bit Counter

Figure 9 illustrates the relationship between the update rate and a rotary encoder's speed as a function of the encoder's resolution for the IXSE502 (8-bit counter). There is no dependency on the quadrature state error.

Using 2 cascaded IXSE503 in a master slave configuration provides a 16 bit up/down counter for incremental shaft encoder application. In this case the relationship between the motor speed and the update rate is modified by a factor of 256. The formula to use for calculation of maximum speeds as a function of sample rate for a 16 bit counter is as follows:

$$\text{encoder speed} = \frac{32768 \cdot (\text{update rate}) \cdot 60}{4 \cdot (\text{lines per revolution})}$$

where encoder speed is in RPM and update rate in cycles per second.

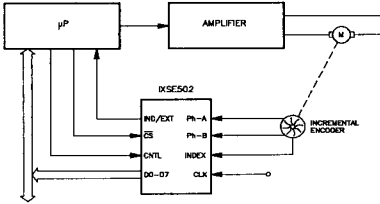


Figure 10: General Application Block Diagram

IXSE503 Circuit Configuration

The IXSE503 is mostly identical in operation and specification to the IXSE502. The main differences are in the package (24 pin plastic skinny DIP) and thus the pinout. The IXSE503 has 4 additional pins that allow cascading of those devices to form a 16-bit or more up/down counter. The cascadeable configuration is operating as an incremental interface, and thus is reset upon read the same way as the 8 bit counter in the IXSE502. This 16-bit counter configuration is not meant to operate as an absolute position counter. The absolute position should be kept in the microprocessor the same as in the case of the IXSE502. It expands the range of applications to higher motor speeds, with higher encoder resolutions than the 8-bit counter.

The IXSE503 should be configured in a MASTER/SLAVE configuration. Pin 19 (SLAVE) programs the part to be either a MASTER (logic "0") or SLAVE (logic "1"). Pin 18 (CO) of the MASTER configured IXSE503 provides a counter overflow or carry out signal and should be connected to pin 7 (C) of the SLAVE IXSE503. This pin is a Carry in pin that enables the counter. The count advance is performed as response to the phase inputs only during the time that C1 is held "low", which occurs only when the counter of the MASTER device overflows.

Pin 6 (RDSB) provides for orderly read operation of the 16-bit counter. In order to avoid bus contention, the CS signal will latch the contents of both 8 bit counters (MASTER and SLAVE) into Latch A simultaneously, but enable only the MASTER's output bus. After release of CS, the SLAVE's output bus could be enabled by asserting RDSB. The same procedure should be followed for reading Latch B.

The most significant byte of the 16-bit counter is in the IXSE503 which is configured as a SLAVE. The least significant byte is in the MASTER. To read the 16-bit counter using the 8-bit bus, read first the LSB using the CS signal followed by the MSB using the RDSB signal. Both the LSB and the MSB will be latched simultaneously upon CS.

The circuit diagram for a cascadeable configuration of two IXSE503 to form a 16-bit counter is shown in figure 11. Note that for proper operation both IXSE503 should be fed from an identical clock signal. Care should be taken in the layout to guarantee short lines of clock between the 2 IC's to eliminate any phase delay.

Initializing Latch A

Initializing Latch A is a very simple process. All that is required is to read Latch A. This will reset the Up/Down Counter and clear the sampling clock inhibit logic. The number that is read from Latch A should be ignored. Instead, set your internal absolute position counter to zero or some meaningful offset. After Latch A is initialized, the IXSE502/503 will immediately begin to count quadrature pulses. Thus, the next read of Latch A must occur soon enough to guarantee that the internal Up/Down Counter does not overflow.

Initializing Latch B

Initializing Latch B is also a very simple process. There is, however, a choice to be made; whether to set Latch B for trigger on an EXTERNAL EVENT pulse only, or for trigger on either an EXTERNAL EVENT pulse or an INDEX pulse. To initialize for trigger on an EXTERNAL EVENT pulse only simply read Latch B. This places the Latch B trigger in the

"FREE" state. Be sure the CNTL input does not go through a low-to-high transition after Latch B is in the "FREE" state, otherwise the Latch B trigger will be placed in the "SET" state. A high-to-low transition on the CNTL input does not affect the state of the Latch B trigger.

To initialize for trigger on either an INDEX pulse or an EXTERNAL EVENT pulse read Latch B (this moves the Latch B trigger to the "FREE" state) and then force a low-to-high transition on the CNTL input. The Latch B trigger will now be in the "SET" state and will remain in this state until Latch B is either triggered or read. A trigger of Latch B will place it in the "LOCK" state, and reading Latch B will place it in the "FREE" state. Once in the "SET" state, additional transitions on the CNTL input will have no

IXYS Corporation • 2355 Zanker Rd. San Jose, CA 95131-1109 • TEL: (408)435-1900 • FAX: (408) 435-0670

ABB-IXYS Semiconductor GmbH • POB 1180 • D-6840 Lampertheim, Germany • TEL: +49-6206-5030 • FAX: +49-6206-503627 • TLX: 662422602 abb

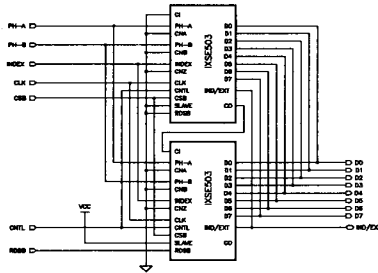


Fig 11: IXSE503 Cascaded Configuration - 16-bit Counter Initializing

affect on the state of the Latch B trigger.

Example IXSE502/503 to Microprocessor Interfaces

When designing a bus interface to go between the IXSE502/503 and a microprocessor or microcomputer it is important to keep in mind the fact that during a read of Latch A the CS and CNTL inputs must remain true until at least one rising edge of CLK occurs. This means that close attention must be paid to the read cycle timing diagrams provided in the microprocessor's data sheet if CLK is driven by the microprocessor's system clock. If CLK is driven by a clock source other than the system clock (an asynchronous source), there will be a time delay from CS true to valid data on the IXSE502 data bus. This time delay has a worst case time of $t_{CLK} + t_{CHDV}$, where t_{CLK} is the period of the asynchronous clock and t_{CHDV} is the worst case "CLK High to Data Valid" delay defined in the data sheet.

This section contains some examples illustrating synchronous IXSE502/503 to microprocessor interfaces for some popular general purpose microprocessors and microcomputers. These examples are called synchronous because the IXSE502/503 CLK input is driven by the system clock or some derivation of the system clock. In all cases the rising edge of CLK has a predictable relationship with respect to a read cycle. In some cases it is necessary to insert a wait state in the read cycle in order to guarantee the occurrence of CLK during a read and/or the appropriate delay between the rising edge of CLK and data valid from the IXSE502/503. As for the instruction sequences (software) provided with each example, only those instructions needed to read the IXSE502 are illustrated. It is assumed that the circuit initialization has been taken care of elsewhere.

IXYS Corporation • 2355 Zanker Rd. San Jose, CA 95131-1109 • TEL: (408)435-1900 • FAX: (408) 435-0670
 ABB-IXYS Semiconductor GmbH • POB 1180 • D-6840 Lampertheim, Germany • TEL: +49-6206-5030 • FAX: +49-6206-503627 • TLX: 662422602 abt

8051 to IXSE502 Interface

Figure 12 is an interface for the Intel 8051 single-chip microcomputer. The interface shown assumes that port 0 is to be used as a bi-directional data bus (AD0 through AD7) and, therefore, the MOVX instruction will be used to read the IXSE502. It is also assumed that port 2 holds the upper half of the address bus (the 16-bit addressing MOVX instruction will be used) and that linear address decoding is used rather than an external address decoder (such as a 74LS138).

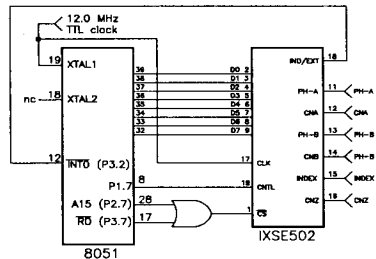


Figure 12: 8051 to IXSE502/503 Synchronous Interface

To read the IXSE502 the following instruction sequence could be used, (it is assumed that the carry flag is saved during an interrupt procedure):

```

IXSE502      equ 0EFFFH      ; address of the IXSE502

READ A:      setb c          ; set the carry, for
                                   ; remember EA
  
```

```

: jbc EA,ENABLED A : disable interrupts,
:                   : if enabled
:                   : clr carry if interrupt
:                   : disabled
ENABLED A:  : clr P17      : clear the CNTL
:                   : input
:                   : mov dptr,#IXSE502 : get the 502's
:                   : address
:                   : movx a,@dptr   : read the Latch A
:                   : register
:                   : jnc DISABLED A   : were interrupts
:                   : enabled?
:                   : setb EA        : yes, re-enable
:                   : interrupts
DISABLED A:  :
:
READ B:      : setb c        : set the carry, for
:                   : remember EA
:
:                   : jbc EA, ENABLED B : disable interrupts, if
:                   : enabled
:                   : clr c         : clr carry if interrupt
:                   : disabled
ENABLED B:   : setb P17     : set the CNTL input
:                   : mov dptr,#IXSE502 : get the 502's
:                   : address
:                   : movx a,@dptr   : read the Latch B
:                   : register
:                   : jnc DISABLED B   : were interrupts
:                   : enabled?
:                   : setb EA        : yes, re-enable
:                   : interrupts
DISABLED B:  :
:

```

MOVX instruction, a rising edge of CLK will occur that has sufficient setup time to guarantee valid data is read by the 8051. At 12 MHz this clock should have a 50% duty cycle.

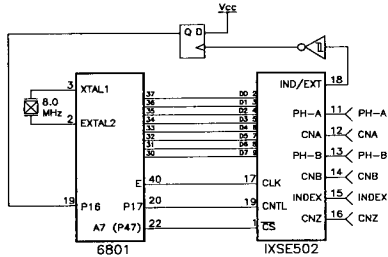


Figure 13: 6801 to IXSE502/503 Synchronous Interface

6801 to IXSE502/503 Interface

Figure 13 is an interface for the Motorola 6801 single-chip microcomputer. The interface shown assumes that port 3 is to be used as a bi-directional data bus (D0 through D7) in the Expanded Non-Multiplexed mode of operation (mode 5). Thus, port 4 holds the external address bus (A0 through A7). Linear address decoding is used rather than an external address decoder (such as a 74LS138).

To read the IXSE502/503, the following instruction sequence could be used (the following instructions assume that a hardware interrupt procedure will not return with interrupts disabled):

```

IXSE502      equ $017F      ; address of the
                          ; IXSE502
READ A      : tpa          : get the interrupt mask
:                   : bit
:                   : psha         : save the interrupt
:                   : mask bit
:                   : sei         : disable interrupts
:                   : ldab P1 IMAGE : get the port 1 image
:                   : andb #7FH    : clear P17 in the image
:                   : stab P1 IMAGE : replace the port 1
:                   : image
:                   : stab P1      : clear the CNTL input
:                   : ldab IXSE502 : read the Latch A
:                   : register
:                   : pula         : get the interrupt
:                   : mask bit
:                   : tap         : put the interrupt mask

```

The above instructions disable interrupts (this guarantees the CNTL bit is not modified before the IXSE502 is read). Set the CNTL bit to the appropriate state and read the selected register. Interrupts are enabled at the end of the procedure only if they were enabled before the IXSE502 was read. If it can be guaranteed that interrupt procedures will not be reading the IXSE502 and/or modifying the CNTL input, the instructions associated with disabling and enabling interrupts can be removed.

Notice that the IND/EXT output is tied to the 8051's INT0 interrupt input. It is assumed that this input is programmed to be edge-sensitive (high-to-low) and that the IND/EXT pulse will last at least 12 system clock cycles (one microsecond for a 12 MHz clock). If this is not true, the IND/EXT's falling edge should be latched.

The system clock is derived from an external TTL oscillator rather than using a crystal tied directly to the microprocessor. This is necessary because the 8051 does not provide a useful system clock for peripherals. During the read cycle generated by a

assume that a hardware interrupt procedure will not return with interrupts disabled):

```

IXSE502      equ 0XXXXH      ; I/O address of the
                .              ; IXSE502

READ A       pushf           ; save the interrupt flag
                cfi           ; disable interrupts
                mov al,0      ; clear the accumulator
                out IXSE502,al ; clear the CNTL input
                in al,IXSE502 ; read the Latch A
                popf          ; register
                .             ; put interrupt flag back
                .

READ B       pushf           ; save the interrupt flag
                cfi           ; disable interrupts
                mov al,1      ; set accumulator to
                .             ; a one
                out IXSE502,al ; set the CNTL input
                in al,IXSE502 ; read the Latch B
                popf          ; register
                .             ; enable interrupts
                .
    
```

The above instructions disable interrupts (this guarantees the CNTL bit is not modified before the IXSE502 is read), set the CNTL bit to the appropriate state, and read the selected IXSE502 register. Interrupts are enabled at the end of the procedure only if they were enabled before the IXSE502 was read. If it can be guaranteed that interrupt procedures will not be reading the IXSE502 and/or modifying the CNTL input, the instructions associated with disabling and enabling interrupts can be removed.

The IXSE502 could be placed in the memory space rather than the I/O space if at least one wait-state is generated during an IXSE502 read. The two D-Q flip-flops and the OR gate (NAND gate in the 8088 Minimum mode system) generate one wait-state whenever an I/O device is addressed (also when an interrupt acknowledge cycle occurs). The CNTL input can be connected to an address bit at the expense of decreased controllability over the state of the Latch B Trigger.

Notice that IND/EXT drives a flip-flop. The IND/EXT output can be polled through an I/O bit or interrupt driven through an interrupt controller, such as an 8259A. The Q output of the flip-flop indicates that a Latch B Trigger has occurred and the Latch B position should be calculated. This calculation should take place before Latch A is read.

The single wait-state is required to guarantee that

the latching edge of CLK occurs with sufficient setup time to insure that valid data is read by the 8086 or 8088. Since this is a synchronous system the system clock can be any frequency less than or equal to that shown in figures 14 and 15. The 24.0 MHz crystal generates an 8.0 MHz system clock that has a 33% duty cycle. The high time of this 33% duty cycle clock is equivalent to a 12.0 MHz clock with 50% duty cycle. This means that this is the maximum frequency 8086/88 system clock that can be driven into the IXSE502's CLK input.

SOFTWARE CONSIDERATIONS

Measuring Position and Velocity

As was mentioned previously, Latch A is an incremental position register. It returns the change in position since the last read of Latch A. The following example illustrates the difference between an incremental up/down counter and an absolute up/down counter.

Incremental Counter Absolute Counter

both counters are reset to zero	
encoder moves	
5 counts in the positive direction	
read counter	read counter
counter reads 5	counter reads 5
read counter	read counter
counter reads 0	counter reads 5
encoder moves	
6 counts in the positive direction	
read counter	read counter
counter reads 6	counter reads 11
encoder moves	
3 counts in the negative direction	
read counter	read counter
counter reads -3	counter reads 8

Each time the incremental up/down counter is read its contents are reset to zero. The absolute up/down counter's contents are not reset when the counter is read. Thus, in order to track the encoder's absolute position, when using an incremental up/down counter, one must add the change in encoder position to an absolute position variable stored in memory. This would also need to be done when using an absolute up/down counter, the absolute counter's word size was too small to guarantee no overflows of the counter within the range of the encoder's travel.

Because the IXSE502/503 returns an incremental count, it can be used to quickly estimate the velocity

of the encoder as well as track the encoder's position. If the time interval between reads of the IXSE502/503 is known, the velocity of the encoder is the Latch A number divided by the time interval. If the time interval is periodic (i.e., it is fixed), then the number read from Latch A will be proportional to the velocity of the encoder. In a system which monitors and displays the speed of an encoder, the incremental position number read from Latch A could be used as an index into a lookup table containing encoder velocity values in units of RPM.

Because the data returned by the IXSE502/503 is incremental, and not absolute, one must do some simple arithmetic in order to track the absolute position of an encoder. This means updating an internal absolute position variable each time the Latch A register is read. This absolute position variable would typically be stored in RAM or a dedicated register. Since the number returned by the IXSE502/503 is a two's complement number, keeping track of absolute position is a very simple process. Each time Latch A is read, the internal absolute position is updated by adding to it the incremental number read from the IXSE502/503. The result will be the absolute position of the encoder at the time the IXSE502/503 was read. (The incremental number must be sign extended before it is added to the absolute position.)

Using the Index Pulse for Error Detection

The "trigger Latch B on INDEX" feature can be used to great advantage to detect encoder errors. By monitoring the change in encoder position between successive INDEX pulses one can determine if encoder pulses have been lost or gained. There are many ways to write a routine that will check for encoder errors; the method outlined below assumes a rotary encoder with a once per revolution INDEX marker.

The key to detecting an error in position is tracking the absolute position of two successive INDEX pulses. Each time an INDEX pulse triggers Latch B the absolute position of the INDEX pulse must be calculated. This absolute position, NEW INDEX POS, is compared with the previous INDEX's absolute position, OLD INDEX POS. If the absolute value of the difference between the two marker positions is not equal to zero or the resolution of the encoder,

there has been an encoder error. Figure 16 is a flow chart illustrating this method of detecting encoder errors.

The variance in the difference between NEW INDEX POS and OLD INDEX POS, when an encoder error occurs, is an indication of the size of the encoder error. By calculating the variance one can delay the reporting of an error until its size is beyond the acceptable limits of the system's specifications. For example, this might be done to prevent shut-down of a system that has experienced a one count error when the acceptable accuracy is plus or minus five counts.

When an INDEX pulse occurs the Latch B Trigger goes to the "LOCK" state and remains there until Latch B is read. Reading the contents of Latch B places the Latch B Trigger in the "FREE" state. In order to trigger Latch B on an INDEX pulse the Latch B Trigger must be in the "SET" state. Obviously, there will be a non-zero time delay between the moment an INDEX pulse triggers Latch B and the moment the Latch B Trigger is again in the "SET" state. If the algorithm of figure 17 is to work properly, this time delay must not be longer than the shortest time between successive INDEX pulses, when the encoder is travelling at maximum speed. If this condition is not satisfied, the algorithm must check for differences equal to multiples of the encoder resolution. For example, if the maximum speed of the encoder is 3000 RPM (50 rps), the time between INDEX marks at maximum speed will be 20 milliseconds. In this example, the time from the INDEX pulse occurring to the IXSE502/503 being placed back into the "SET" state must be less than 20 milliseconds.

Satisfying the maximum time delay requirement should not be a problem for most systems. Comparing the difference between NEW INDEX POS and OLD INDEX POS to two revolutions, as well as one revolution, doubles the maximum time delay requirement. Thus, theoretically, the time delay can be as large as one requires. If the encoder is moving back and forth near the INDEX marker it is possible to encounter an INDEX marker before the IXSE502/503 has been placed into the "SET" state. This is not a problem if the calculated differences are based on absolute positions. As long as the maximum time delay requirement is satisfied, the algorithm of figure 16 should work well for most microprocessor systems.

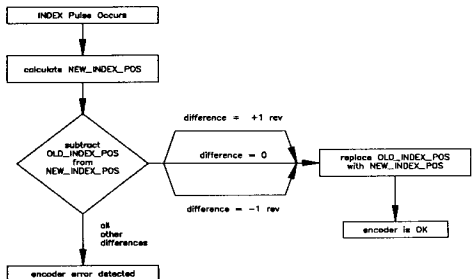
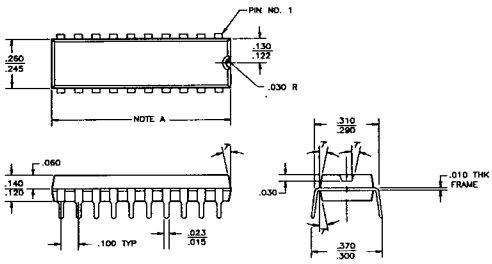


Fig 16: Flow Chart to Detect Encoder Errors

Ordering Information

PART NO.	DESCRIPTION	PACKAGE	TEMPERATURE RANGE
IXSE502PC	8 BIT SHAFT ENCODER INTERFACE	20 PIN PDIP	COM 0 TO 70 °C
IXSE502PI	8 BIT SHAFT ENCODER INTERFACE	20 PIN PDIP	IND -40 TO 85 °C
IXSE503PC	CASCADABLE SHAFT ENCODER INT.	24 PIN PDIP (.3")	COM 0 TO 70 °C
IXSE503PI	CASCADABLE SHAFT ENCODER INT.	24 PIN PDIP (.3")	IND -40 TO 85 °C



NOTE A :	PACKAGE	DIM.
IXSE502	20 PIN	1.040" / 1.020"
IXSE503	24 PIN	1.240" / 1.220"

Note: All dimensions are in inches

IXYS Corporation • 2355 Zanker Rd. San Jose, CA 95131-1109 • TEL: (408)435-1900 • FAX: (408) 435-0670
 ABB-IXYS Semiconductor GmbH • POB 1180 • D-6840 Lampenheim, Germany • TEL: +49-6206-5030 • FAX: +49-6206-503627 • TLX: 662422602 abd