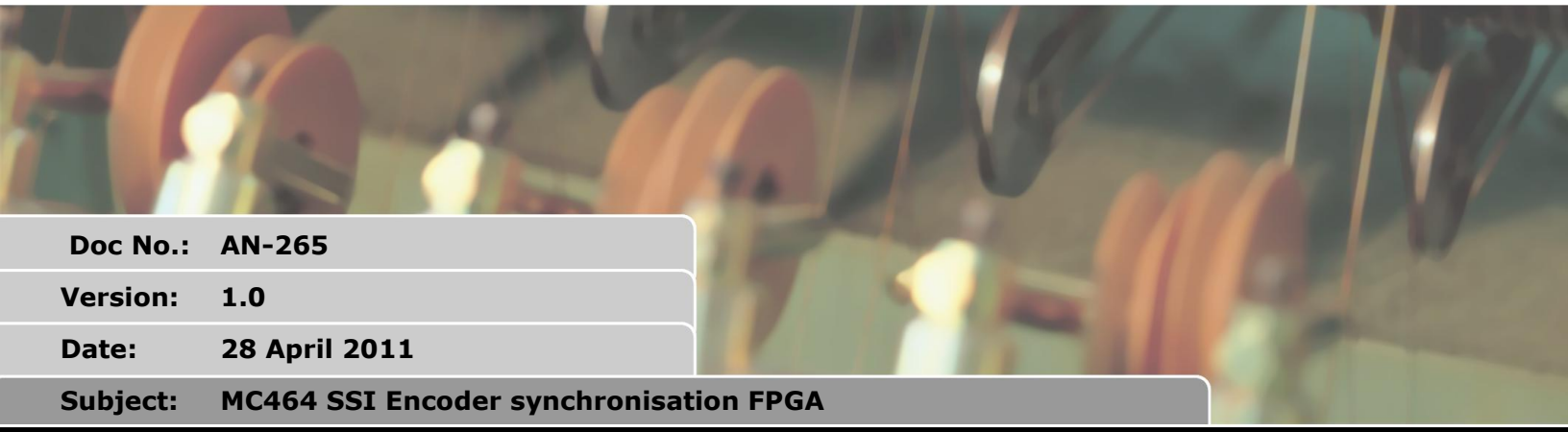
Trio Motion Technology Ltd.
Shannon Way, Tewkesbury,
Gloucestershire. GL20 8ND
United Kingdom
Tel: +44 (0)1684 292333
Fax: +44 (0)1684 297929

1000 Gamma Drive
Suite 206
Pittsburgh, PA 15238
United States of America
Tel: +1 412.968.9744
Fax: +1 412.968.9746

Tomson Centre
118 Zhang Yang Rd., B1701
Pudong New Area, Shanghai,
Postal code: 200122
P. R. CHINA
Tel/Fax: +86-21-58797659



Doc No.: AN-265

Version: 1.0

Date: 28 April 2011

Subject: MC464 SSI Encoder synchronisation FPGA

APPLICATION NOTE

www.triomotion.com

1. Contents

2. Introduction

This application note details the new features available on the MC464 (P860) when updated with a special version of FPGA. This update has been made available in response to an application requirement for multiple MC464's synchronising to a master SSI encoder. The FPGA is not field upgradable and must be programmed at Trio, therefore customers requiring this feature must specify it at the time of ordering.

3. System Requirements

The MC464 Motion Coordinators must be loaded with system software v2.0139 or higher and FPGA version MC464_SSI 1F. This feature uses the built-in encoder port on the MC464, so this will not be available for any other use.

Use of this feature in conjunction with pulse and direction or quadrature axis outputs may result in very occasional errors of +/-1 step output. Therefore this is not a recommended mode of operation.

4. Wiring Configuration

The SSI encoder connects directly to the "A", "/A", "Z" and "/Z" connections of all the MC464 encoder ports on the modules to be synchronised. The "B" and "/B" channels must be connected between all of the modules but not to the encoder. This synchronises the servo periods of all the controllers so that they read the SSI encoder at the same time. The following table gives the encoder connections to the MC464 encoder port:

SSI Encoder Signal	D-type Pin Number	MC464 (0) Encoder Port	MC464 (1) Encoder Port	....	MC464 (n) Encoder Port
CLOCK	1	A	A		A
/CLOCK	2	/A	/A		/A
	3	B	B		B
	4	/B	/B		/B
DATA	6	Z	Z		Z
/DATA	7	/Z	/Z		/Z
0V (Signal Gnd)	5	0V	0V		0V

The encoder must be connected to its power supply. Often this is +24V and separate power wires are contained in the encoder cable for +24V power and 0V power. If the encoder runs from 5V and requires less than 150mA the power can be taken from ONE of the MC464 encoder ports (pin 8). ***Do not link the 5V outputs from multiple MC464's.***

The clock and data signals are ALWAYS low voltage (5V) RS422 level. Do not connect signals which are above this level. Most SSI encoders have signals at 5V level, even when powered from 24V dc. Please confirm this before connecting.

5. Configuring the Motion Coordinators

At power up the MC464 encoder port will behave as an *incremental* encoder input. One of the MC464's must be set as the SSI master and all of the others are set as slaves. The master module will generate the SSI clock and read the data, whereas the slaves read in the clock and data. It is necessary to set the configuration on all the controllers by changing the axis ATYPE using a BASIC program:

Axis ATYPE	**Description**
48	SSI Absolute (Master)
75	SSI Absolute (Slave)

The axis number of the MC464 encoder port will be determined by the number and type of modules fitted to the MC464 and the setting of the AXIS_OFFSET parameter. See the MC464 User Manual for more information.

Example:

Assuming a system with two FlexAxis8 Interfaces fitted and AXIS_OFFSET at default, the master module would be set as follows:

```
ENCODER_BITS AXIS(16)=24 ' 24 bit SSI encoder
ATYPE AXIS(16) = 48 ' set up axis 16 as SSI Absolute Master
```

The slave module(s) would be set as follows:

```
ENCODER_BITS AXIS(16)=24 ' 24 bit SSI encoder
ATYPE AXIS(16) = 75 ' set up axis 16 as SSI Absolute Slave
```

6. Relevant Trio BASIC Commands

The BASIC has a set of keywords to give the user access to the encoder data. More information on these keywords can be found in the MC464 User Manual.

6.1. ENCODER_BITS

This axis parameter configures the interface for the number of encoder bits and the data type, e.g. Binary or Gray code. The parameter should be set to the total number of encoder bits, maximum 32. For example: A 24 bit Gray coded encoder has 12 multi-turn and 12 bits/turn resolution. The BASIC program initialises with:

```
BASE(axis_number)
ENCODER_BITS=24
ATYPE=48
```

Note that the parameter must be set for each axis and the value MUST be the same on all controllers linked to the SSI encoder.

6.2. ENCODER

This axis register contains the actual absolute value being returned by the encoder. Whereas MPOS can be changed by the DEFPOS, OFFPOS and REP_DIST functions, the ENCODER value will always show the true encoder absolute value.

Note that MPOS is the value used by the servo software to show the axis position. ENCODER is only for service diagnostic use.

6.3. FPGA_VERSION

To confirm that the MC464 has the updated FPGA it is possible to interrogate the version using the FPGA_VERSION parameter. The version needs to be 31 or greater to support SSI.

```
IF FPGA_VERSION<31 THEN
  PRINT "THIS MC464 DOES NOT SUPPORT SSI ENCODER"
  STOP
END IF
```

7. Error Handling

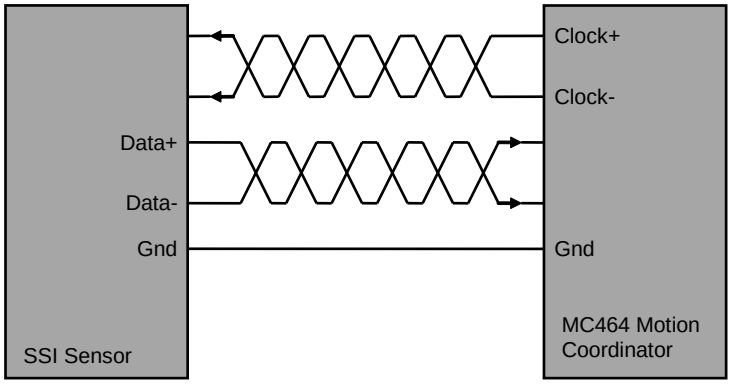
SSI data packets are sent as plain Gray Code or Binary numbers. There is no checksum or CRC included in the SSI format so error detection is minimal. If incorrect data is sent which results in the position going outside the following error limit (FE_LIMIT) then a servo error will occur. Otherwise the controller will have no knowledge of the error at all.

Where a SSI encoder axis is used in a non servo application, it is up to the programmer to ensure that suitable precautions are taken to detect data errors should they occur.

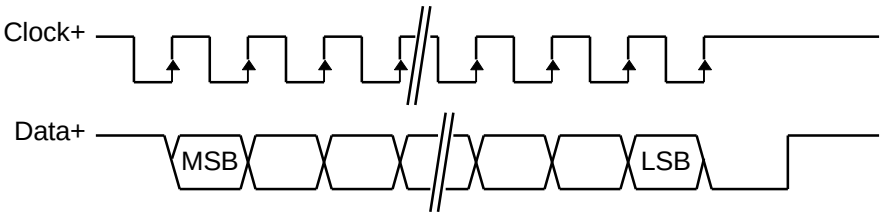
8. SSI Interface

SSI or Synchronous Serial Interface is a digital system for transferring data in serial form and is the most widely used serial interface between absolute sensors and controllers. The SSI convention uses a pulse train from the controller to clock out the data from the sensor.

Block Diagram



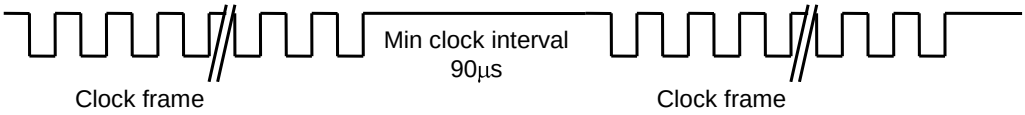
Timing Diagram



8.1. Data Interrogation

Once initialised by the ENCODER_BITS command the MC464 continually interrogates the sensor by sending clock pulses in "frames" of n+2 pulses where n is the bit count set. One clock frame is sent every servo period and the clock rate is fixed at 200KHz. The clock interval between frames depends on the number of bits and the servo period. With a servo period set to 250μs ENCODER_BITS set to 32 it is 90μs. There is a maximum cable length of 200m between the controller and the sensor.

Clock Sequence



8.2. Signal Format

When the data has been clocked in to the Trio *Motion Coordinator*, the firmware interprets the position value to produce a value for MPOS and hence a position error that is used to close the position control loop. ENCODER_BITS must be set to suit the encoder. (Maximum 32 bit)

9. Fault Finding

- Check the wiring and make sure the signal 0V is connected. SSI will not work if only the signal lines are connected with no 0V reference.

Symptom	Action
When ENCODER_BITS is set, the value of MPOS remains at 0.	Check that the Motion Coordinator is capable of running SSI. Check that the ATYPE has been set correctly. (`ATYPE=48 or ATYPE=75`)
When ENCODER_BITS is set, the value of MPOS goes to a number which does not change when the encoder is turned.	Check the clock and data connections between the MC464 and the encoder. In a system with multiple MC464's linked to one SSI encoder, ensure that only one MC464 is set as the SSI master and all the others are slaves. If possible use an oscilloscope to confirm that clock pulses are being transmitted from the MC464. Also check that data pulses are being transmitted by the encoder. See that the data pulses change as the encoder is turned.
The value of MPOS changes as the encoder is turned but the value jumps up **and** down when turning in one direction.	Check that bit 6 of the ENCODER_BITS parameter is set correctly for the data type. `Bit 6 of ENCODER_BITS=1` for Binary `Bit 6 of ENCODER_BITS=0` for Gray Code Check that the DATA lines are connected the right way around. Try swapping the DATA+ and DATA- lines.
The value of MPOS on the SSI master does not match the value of MPOS on the SSI slave(s).	Check that the ATYPE has been set correctly. (`ATYPE=48 or ATYPE=75`) Check that the encoder port B channel (B & /B) are linked between all the MC464's.

The value of MPOS goes outside the range of the encoder absolute position.	Set REP_DIST to the value of the encoder scale.
	`ENCODER_BITS=13`, use `REP_DIST=8192`
	`ENCODER_BITS=24`, use `REP_DIST=16777216`
	Set `REP_OPTION=ON`

After using DEFPOS or OFFPOS, the MPOS value is not the absolute value.	This is normal. The absolute value is set in MPOS when you set `ENCODER_BITS=nn`. After this, the program can change the value of MPOS if required. Use `DEFPOS(ENCODER)` to restore the absolute value when the axis is not moving.