

Mecco EtherMark Object Model

Modbus Register Map

Version 2.3

July 23, 2018

Table 1

O2T (Output Block) *Outbound from Modbus Master to EtherMark*

Modbus Convention	(Write) Holding Registers	
	Address	End Address
41001	1000	1000
41002	1001	1001
41003	1002	1002
41004	1003	1003
41005	1004	1004
41006	1005	1005
41007	1006	1006
41008	1007	1007
41009	1008	1008
41010	1009	1009
41011	1010	1010
41012	1011	1011
41013	1012	1012
41014	1013	1013
41015	1014	1015
41017	1016	1017
41019	1018	1019
41021	1020	1021
41023	1022	1023
41025	1024	1025
41027	1026	1043
41045	1044	1061
41063	1062	1079
41081	1080	1097
41099	1098	1115
41117	1116	1133
41135	1134	1151
41153	1152	1217

# Bytes	Type	Source Code Struct Member	Description	Notes
2	int16	x_offset_mm	X offset (+/- 160mm X 100)	
2	int16	y_offset_mm	Y offset (+/- 160mm X 100)	
2	int16	rotation_deg	Rotation (0-360 deg x 10)	
2	int16	x_rot_center_mm	X center of rotation (mm x 100)	
2	int16	y_rot_center_mm	Y center of rotation (mm x 100)	
2	uint16	x_scale	X scale (scale x 1000)	8
2	uint16	y_scale	Y scale (scale x 1000)	8
2	uint16	reserved	Reserved (2 bytes) for 32-bit alignment	8
2	uint16	cmd_seq_num	Marker command	8
2	uint16	LEC_computed_job_field_num	LEC computed job field #	8
2	uint16	job_tmo_sec	Mark Timeout (seconds, 0 disable)	8
2	uint16	tcp_tmo_sec	TCP Timeout (seconds, 0 disable)	8
2	uint16	barcode_pass_grade	Pass grade for barcode verification	8
2	uint16	max_remarks	Maximum number of remarks (0, 1, 2 ..)	8
4	CIP_INDEX_UNIQUE	short_field_index_unique[0]	Short Field 1	2
4	CIP_INDEX_UNIQUE	short_field_index_unique[1]	Short Field 2	2
4	CIP_INDEX_UNIQUE	short_field_index_unique[2]	Short Field 3	2
4	CIP_INDEX_UNIQUE	short_field_index_unique[3]	Short Field 4	2
4	CIP_INDEX_UNIQUE	short_field_index_unique[4]	Short Field 5	2
4	CIP_INDEX_UNIQUE	long_field_index_unique[0]	Long Field 1	2
36	CIP_STR32	marker_job_file_name	Laser or Dot Peen Job File Name	3
36	CIP_STR32	camera_job_file_name	Camera (Cognex) Job File Name	3
36	CIP_STR32	short_field_string[0]	Short Field String 1	3
36	CIP_STR32	short_field_string[1]	Short Field String 2	3
36	CIP_STR32	short_field_string[2]	Short Field String 3	3
36	CIP_STR32	short_field_string[3]	Short Field String 4	3
36	CIP_STR32	short_field_string[4]	Short Field String 5	3
132	CIP_STR128	long_field_string	Long Field String	3

436	Total Bytes
218	Total Modbus Registers

Table 2

T20 (Input Block)

Inbound to Modbus Master from EtherMark

Modbus Convention	(Read) Holding Registers	
	Address	End Address
42001	2000	2000
42002	2001	2001
42003	2002	2003
42005	2004	2005
42007	2006	2006
42008	2007	2007
42009	2008	2009
42011	2010	2011
42013	2012	2029
42031	2030	2095
42097	2096	2129
42131	2130	2195

# Bytes	Type	Source Code Struct Member	Description	Notes
2	uint16	marker_state	Marker state number	8
2	uint16	reserved	Reserved (2 bytes) for 32-bit alignment	8
4	uint32	last_mark_cycle_time_ms	Elapsed and Last Mark Cycle Time (ms)	4
4	uint32	status_gpio	Status DWORD / GPIO	4
2	uint16	marker_error_code	Marker Error Code	8
2	uint16	controller_error_code	Controller Error Code	8
4	uint32	heart_beat	Heart Beat	4
4	uint32	input_check_sum	32-bit simple sum of all input field bytes	1, 4
36	CIP_STR32	echo_job_file_name	Echo Job File Name	3
132	CIP_STR128	results_field	LEC computed job field string, or general command results	3
68	CIP_STR64	marker_state_desc	Marker state description	3
132	CIP_STR128	marker_error_desc	Marker error description	3

392	Total Bytes
196	Total Modbus Registers

Table 3

Data Types

Read/Write Holding Registers		# Bytes	Description	Notes
Address	Length			
int16				
N	High Byte	1	High order 8 bits of 16 bit signed integer	5
N	Low Byte	1	Low order 8 bits of 16 bit signed integer	
uint16				
N	High Byte	1	High order 8 bits of 16 bit unsigned integer	5
N	Low Byte	1	Low order 8 bits of 16 bit unsigned integer	
uint32				
N	1	2	High order 16 bits of 32 bit unsigned integer	5, 3
N + 1	1	2	Low order 16 bits of 32 bit unsigned integer	

Table 4

Strings and Nested Data Structures

Read/Write Holding Registers		# Bytes	Type	Source Code Struct Member	Description	Notes
Address	Length					
CIP_STR32						
N	2	4	uint32	str_len	String Length	3, 5
N + 2	16	32	char	str[32]	String	5, 6
CIP_STR64						
N	2	4	uint32	str_len	String Length	3, 5
N + 2	32	64	char	str[64]	String	5, 6
CIP_STR128						
N	2	4	uint32	str_len	String Length	3, 5
N + 2	64	128	char	str[128]	String	5, 6
CIP_INDEX_UNIQUE						
N	High Byte	1	uint8	index	Index	5, 7
N	Low Byte	1	uint8	unique	Uniqueness flag	5
N + 1	High Byte	1	uint8	verify	Cognex flags (bit 0 = Verify barcode, bit 1 = Verify OCR)	5
N + 1	Low Byte	1	uint8	justify	Text justification (0 = Left, 1 = Center, 2 = Right)	5

Description

- * The EtherMark responds as a Modbus/TCP Slave, listening on TCP port 502.
- * 16 bit Modbus Holding Registers are used to transfer data and control.
- * The EtherMark object model defines a T2O input block, and an O2T output block.
 - * O2T Output Block to EtherMark: 218 registers (436 bytes) of marking parameters, and data to be marked.
Address: **1000**
 - * T2O Input Block from EtherMark: 196 registers (392 bytes) of marking status/error info.
Address: **2000**
- * Modbus function **03 (Read Holding Registers)**, is used to read the T2O input block, and can also read back the O2T output block.
- * Modbus functions **06 (Write Single Register)** and **16 (Write Multiple Registers)**, are used to write the O2T output block.
- * Data in 16 bit Modbus Holding Registers is stored in Big Endian byte order (high byte first).
- * Modbus convention refers to the Modbus Slave address as being offset from the following value:
Holding Register Address: **40001**

Notes

Modbus convention refers to the Modbus Slave address (or reference range) as being offset from the Holding Register value 40001. Thus, for example, in the Modbus Master device, the Slave address 1000 would be referred to as Holding Register 41001.

1. The checksum is the 32-bit simple sum of the following fields: job_file_name, camera_job_file_name, all short field strings, and the long field string. If concatenation is employed on the long string, the checksum is computed using the full length concatenated string. The checksum is updated by specific command request (see command listing), and at the start of every mark sequence.
2. The CIP_INDEX_UNIQUE data structure contains 4 bytes, and is described in **Table 4**.
3. The CIP_STR32, CIP_STR64, and CIP_STR128 data structures contain an ASCII string and length field, and are described in **Table 4**.
4. The uint32 data type represents an unsigned 32 bit integer, and is described in **Table 3**.
5. 'N' represents the Holding Register address of a data structure (CIP_STR32, CIP_STR64, CIP_STR128, CIP_INDEX_UNIQUE).
6. ASCII Strings are stored two characters per Holding register, first char in high byte, second in low byte. The str_len register defines the string length.
7. Index specifies the destination to send string content: for laser, the Winlase object number; for dot-peen, the MC2000-T2 line number
8. The uint16 data type represents an unsigned 16 bit integer, and is described in **Table 3**.