UDP PROTOCOL FOR THE SIMCARD SC-MB ETHERNET

BUFFER – RECEIVED FROM THE BOARD	**30 BYTES**
BUFFER – SENT TO THE BOARD	**28 BYTES**

SISMO SOLUCIONES

Plug & Fly

BUFFER - RECEIVING

B0	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	B12	B13	B14	B15	B16	B17	B18	B19	B20	B21	B22	B23	B24	B25	B26	B27	B28	B29
83	67	MAC1	MAC2	TYPE REC	DAUGHTERS	PORT H	PORT L	DATA.............																					

83,67 “SC” init string

MAC1 The 2 most significant bits (in decimal) of the 4 last digits of the MacAddress of the Board. E.g.:MAC:00:4f:32:02:01 where MAC1 = 02

MAC2 The 2 least significant bits (in decimal) of the 4 last digits of the MacAddress of the Board. E.g.:MAC:00:4f:32:02:01 where MAC1 = 01

TYPE REC The received value says what it means. It can be 0, 1, 2 or 3: 0 = Inputs and Analog Inputs from the Simcard Mother Board. 1 = Inputs from Simcard Daughter Input1. 2 = Inputs from Simcard Daughter Input2. 3 = Analog Inputs from the Simcard Daughter ADC.

DAUGHTERS Bits of the Simcards Daughter which are activated. The Simcards Daughter Input does not notifies anything, we know that they are because we receive data from them.

BIT0 = Simcard Daughter OUT1 activated. BIT1 = Simcard Daughter OUT2 activated. BIT2 = Simcard Daughter SERVO activated.

BIT3 = Simcard Daughter DYSPLAY1 activated. BIT4 = Simcard Daughter DYSPLAY2 activated. BIT5 = Simcard Daughter ADC activated. BIT6 = NONE.

PORT H High part of the UPD Port where the Simcard is listening. E.g.: Port 1025 = 0401 in hex -> we take the high part 04 which in decimal is 4, where PORT H = 4

PORT L Low part of the UPD Port where the Simcard is listening. E.g.: Port 1025 = 0401 in hex -> we take the high part 01 which in decimal is 1, where PORT L = 1

PORT = PORTH * 256 + PORTL

E.g.: PORT = 4* 256 + 1 = 1025

DATOS MOTHER BOARD

Inputs are from the BYTE8 to BYTE15: 0100100100100011111100... Each bit is an input. The total 8BYTES x 8BITS = 64 INPUTS

Analog Inputs of the Simcard Mother are from the BYTE16 to BYTE25: The adcs are of 10 bits. 2BYTES are used for each one of the 5 adcs. Values are from 0..1024 = BYTE17 * 256 + BYTE16..

DAUGHTER INPUTS BOARD1

Inputs are from the BYTE8 to BYTE15: 0100100100100011111100... Each bit is an input. The total 8BYTES x 8BITS = 64 INPUTS

DAUGHTER INPUTS BOARD2

Inputs are from the BYTE8 to BYTE15: 0100100100100011111100... Each bit is an input. The total 8BYTES x 8BITS = 64 INPUTS

DAUGHTER ADC BOARD

Analog Inputs are from the BYTE8 to BYTE29: The adcs are of 10 bits. 2BYTES are used for each one of the 11 adcs. Values are from 0..1024 = BYTE9 * 256 + BYTE8..

BUFFER – SENDING

B0	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	B12	B13	B14	B15	B16	B17	B18	B19	B20	B21	B22	B23	B24	B25	B26	B27
83	67	TYPE SEND	T. DATA	DATA…………										Reserved													

83,67 “SC” init string

TYPE SEND It means that the data are sent to the Mother or to a Daughter. 0 = Mother. 1 = Daughter OUT1. 2 = Daughter OUT2. 3 = Daughter SERVOS. 4 = Daughter DISPLAYS.

T.DATA This byte is only for the Simcard Mother and Simcard Daughter Servo.

For the Simcard Mother: 0 = outputs are sent. 1 = displays are sent.

For the Simcard Daughter Servo: 0 = first group of servos (the first 8). 1 = second group of servos.

DATOS MOTHER BOARD OUTPUTS (TYPE SEND=0, T.DATA=0)

Outputs are from the BYTE4 to BYTE11: 010010010010001111100… Each bit is an output. The total 8BYTES x 8BITS = 64 OUTPUTS

MOTHER BOARD DISPLAYS (TYPE SEND=0, T.DATA=1)

Displays are in 4 groups. The BYTE4 is the group. The digits are from the BYTE5 to the BYTE12. The BYTE13 is the brightness from 0 to 15. 4 Buffers must be sent in order to complete the 32 displays.

MOTHER BOARD DISPLAYS (TYPE SEND=4, T.DATA=0)

Displays are in 4 groups. The BYTE4 is the group. The digits are from the BYTE5 to the BYTE12. The BYTE13 is the brightness from 0 to 15. 4 Buffers must be sent in order to complete the 32 displays.

DAUGHTER OUTPUT1 (TYPE SEND=1, T.DATA=0)

Outputs are from the BYTE4 to BYTE1: 010010010010001111100… Each bit is an output. The total 8BYTES x 8BITS = 64 OUTPUTS

DAUGHTER OUTPUT2 (TYPE SEND=2, T.DATA=0)

Outputs are from the BYTE4 to BYTE1: 010010010010001111100… Each bit is an output. The total 8BYTES x 8BITS = 64 OUTPUTS

DAUGHTER SERVOS (TYPE SEND=3, T.DATA=0) GROUP 0 -> the first 8 servos

The BYTE4 has the activation bits: BIT0 = enable servo 1, BIT1 = enable servo 2,.., BIT7 = enable servo 8

The data of 0 to 255 of each servo is from the BYTE5 to BYTE12, BYTE5 = Data Servo 1, BYTE6 = Data Servo 2, …

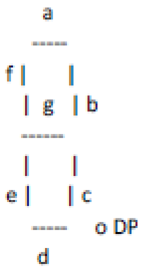
DAUGHTER SERVOS (TYPE SEND=3, T.DATA=1) GROUP 1 -> the second 8 servos

The BYTE4 has the activation bits: BIT0 = enable servo 1, BIT1 = enable servo 2,.., BIT7 = enable servo 8

The data of 0 to 255 of each servo is from the BYTE5 to BYTE12, BYTE5 = Data Servo 1, BYTE6 = Data Servo 2, …

CODE FOR DIGITS

MAC1



A bit6
B bit5
C bit4
D bit3
E bit2
F bit1
G bit0
DP bit7

‘0’=’01111110’
‘1’=’00110000’
‘2’=’01101101’
‘3’=’01111001’
‘4’=’00110011’
‘5’=’01011011’
‘6’=’01011111’
‘7’=’01110000’
‘8’=’01111111’
‘9’=’01111011’
‘-’=’00000001’

‘A’=’01110111’
‘B’=’00011111’
‘C’=’01001110’
‘D’=’00111101’
‘E’=’01001111’
‘F’=’01000111’
‘H’=’00010111’
‘J’=’00111100’
‘O’=’00011101’
‘P’=’01100111’
‘T’=’00001111’
‘’=’00000000’