

FB63 - <offline>

"RdParm" Read Parameter

Nom : rdparm

Famille : MAC

Auteur : arp

Version : 1.0

Version de bloc : 2

Horodatage Code : 15/06/2006 22:09:07

Interface : 06/06/2006 22:23:10

Longueur (bloc/code /données locales) : 00274 00152 00000

Nom	Type de données	Adresse	Valeur initiale	Commentaire
IN		0.0		
RegNumber	Int	0.0	0	Registernumber to read
Is32Bit	Bool	2.0	FALSE	Value read as 16 or 32 bit (word or Dword)
NodeAdr	Int	4.0	0	Start address of the servo node wanted
OUT		0.0		
Value	DInt	6.0	L#0	Requested Value to be returned
IN_OUT		0.0		
STAT		0.0		
TEMP		0.0		

Bloc : FB63 Reads parameter value from register number in servo
--

Reads 1 parameter from a specific register in a MAC drive, per call.
--

Réseau : 1 pre handling order

Test for order allready sent, and if not sent make relative addresspointer
--

```

L   #NodeAdr      //Get nodeaddress number to read
LAR1                //load AR1 from accumulator

```

```

U   "RdOrderSent" //read order sent for handling M99.7      -- RdParm
                                control flag
U   "RdOrderSent" //read order sent for handling M99.7      -- RdParm
                                control flag

```

```

SPB  blok          //jump over order sending

```

Réseau : 2 Select register number in servo
--

Set read parameters, fro register and specify if 16 or 32 bit read
--

```

L   #RegNumber
T   AB [AR1,P#5.0] //set registernumber

U   #Is32Bit      //test for 32 bit command
U   #Is32Bit
SPB  atb
UN   #Is32Bit     //case of 16 bit, then reset read32 flag
R   A [AR1,P#7.4]
SPA  ate
atb: U   #Is32Bit //case of 32 bit, then set read32 flag
      S   A [AR1,P#7.4]

```

```

ate: U      "Dummy"                M1.0
     =      "Dummy"                M1.0

```

Réseau : 3	Toggle Read toggle flag in command byte
------------	---

```

//toggle Q7.6 for servo 1
U      A [AR1,P#7.6] //read read toggle flag for level
U      A [AR1,P#7.6] //read read toggle flag for level
SPBN   res
U      A [AR1,P#7.6] //read read toggle flag for level
R      A [AR1,P#7.6] //if off then set on
SPA    blok
UN     A [AR1,P#7.6] //read read toggle flag for level
res:   S      A [AR1,P#7.6] //if on then set off
blok:  U      "ReadParmSub"                M0.1                -- Activa
                                           te sub
                                           M99.7                -- RdParm
                                           control flag

S      "RdOrderSent" //order has been sent

```

Réseau : 4	Transfer result and reset calling bit
------------	---------------------------------------

Respond from servo, comes after a while when read toggle in command status is equal to read toggle in command.
--

```

//wait for done signal = Q and I are equal
U      A [AR1,P#7.6] //read read toggle flag for level
X      E [AR1,P#7.6] //read read toggle flag for level
SPB    wait          //if not ready jump over function end

//Function end return the requested value received from the register
U      #Is32Bit
U      #Is32Bit
SPB    ttb
L      EW [AR1,P#2.0] //if parameter was an integer val
T      #Value
SPA    end
ttb:   L      ED [AR1,P#0.0] //if parameter was an double integer val
T      #Value

end:   U      "ReadParmSub"                M0.1                -- Activa
                                           te sub
R      "ReadParmSub" //reset call bit      M0.1                -- Activa
                                           te sub
R      "RdOrderSent" //reset order sent work bit
                                           M99.7                -- RdParm
                                           control flag

wait:  U      "Dummy"                M1.0
     =      "Dummy"                M1.0
BE

```