



KOLLMORGEN • THOMSON • DOVER

# DeviceNet™ for SERVOSTAR® S300/S600/S700

## and DeviceNet HMS Simulator

Paul Coughlin



*Helping you build a better machine, faster.*

- DeviceNet is an 8-Byte Field Bus System for medium range industrial I/O control
- Originally created in 1996 for operation of sensors, switches, bar-code scanners, AC/DC drives.
- Currently maintained by independent organization ODVS (Open DeviceNet Vendor's Association)
- Bus connection Controller Area Network (CAN-standard ISO 11898) as base
- Features implemented for Servo Amplifiers
  - Setup and General Functions – homing, jogging, speed/torque control
  - Positioning Functions – execute motion tasks, absolute trajectory
  - Data Transfer Functions – read actual values, errors, inputs, set outputs



- 3 Baud rates available
  - 125k bps
  - 250k bps
  - 500k bps
  
- Distances
  - Up to 500 meters @ Baud rate of 125k
  - Up to 250 meters @ Baud rate of 250k
  - Up to 100 meters @ Baud rate of 500k

- Network Status LED
  - Device not powered, Allocated to Master, Recoverable / Non-recoverable faults
- 2 Rotary Switches for 0 – 63 Amplifier Address
  - Address 10 - MSD switch = 1, LSD switch = 0
- Single Rotary Switch for Baud Rate
  - Switch 0 = 125k bps, 1 = 250k bps, 2 = 500k bps

## Front view



## 2 Communication Methods

- **Explicit Messaging (Slower) – Used to Configure Drive**  
Communications: <50 mSec (Parameter Object <500 mSec)
  - OOT (Object Oriented Technique)
  - Parameters selected through:  
Class Object Number, Instance ID, Attribute ID, Explicit Message
  - Read or Write single parameter at a time
  - Homing can only be done via Explicit Messaging
  
- **Polled I/O (Fast) – Used to Control Movement Real-time**  
Communications: <10 mSec
  - 8 Byte – Control bits & Status bits
  - Command Assembly (write)
  - Response Assembly (read)
  - Limited Parameters (cannot Home via Polled I/O)

## Example of Object Oriented Technique:

- Set Jog Velocity to 500 rpm (01F4 hex)
  - Position Controller Object (class = 0x25)
  - Instance ID = 1
  - Attribute ID = 0x16 (dec 22)
  - Explicit Message = 1F4 (dec 500)
  - Data Type = Double Integer

- Polled I/O
  - Real-time response communications
  - Used to control motion
  - Combines Control & Status bits into 8-byte message
  - Command Assembly (send)
  - Response Assembly (read)
  - Only certain parameters assessable
    - Less versatile than Explicit Messaging



## ■ Command Assembly Structure

- 8 Bytes with 8 status bits
  - Byte 0 dedicated commands
  - Byte 1 identifies Block number (Motion Task number)
  - Byte 2 Command Axis / Command Assembly Type (send data)
  - Byte 3 Response Axis / Response Assembly Type (read data)
  - Bytes 4 – 7 Data Fields for data to be transmitted, such as velocity, position

(bit 5 is always 1 for Command & Response Axis – it is not Amp address)

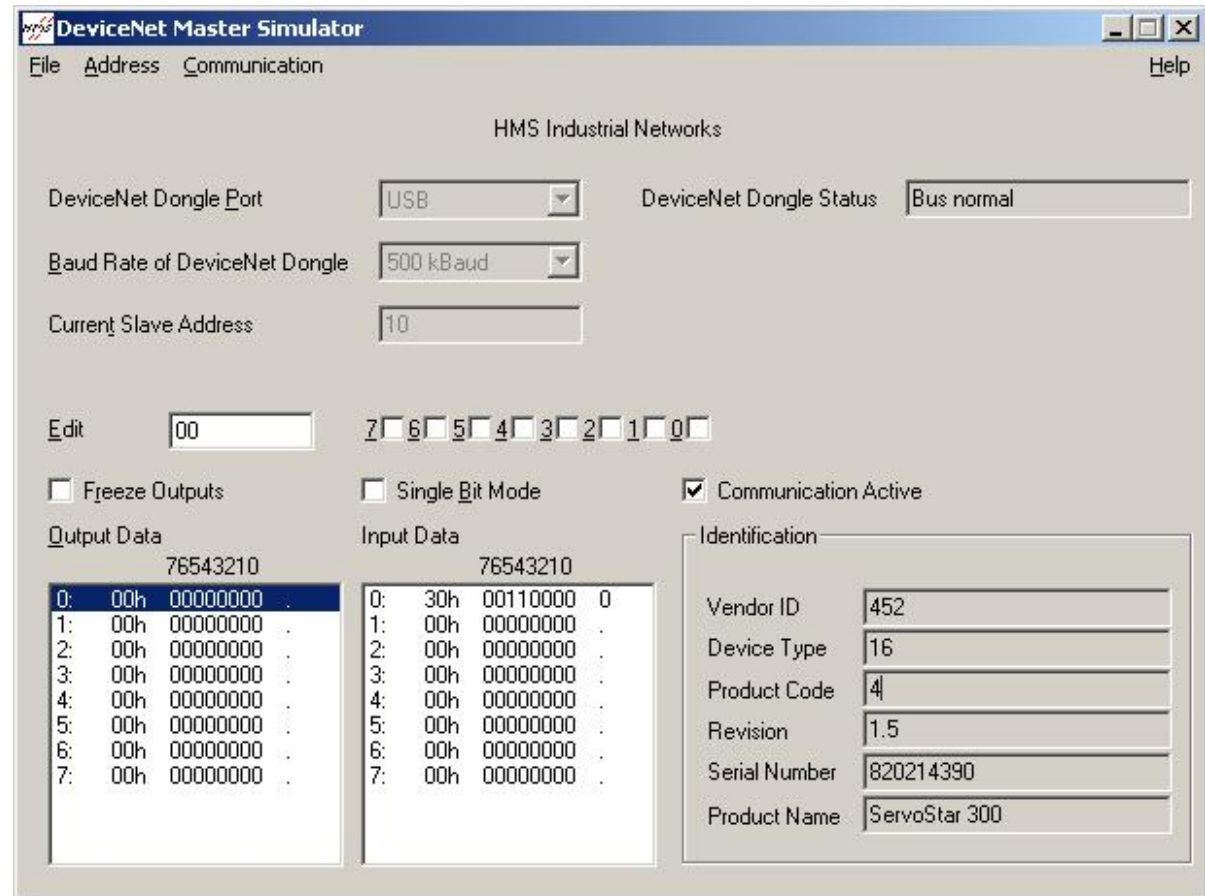
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	Enable	Reg Arm	Hard Stop	Smooth Stop	Dir	Relative	Start Block	Load / Start
1	Block Number							
2	Command axis = 001			Command Assembly Type (00001)				
3	Response axis = 001			Response Assembly Type				
4	Target Position Low Byte							
5	Target Position Low Middle Byte							
6	Target Position High Middle Byte							
7	Target Position High Byte							

## Example of Polled I/O: (need to be in Velocity Opmode)

- Set Jog Velocity to 20,000 counts per second
  - Byte 0 - bit 7 Enables drive, bit 0 Starts motion
  - Byte 1 – not use
  - Byte 2 – selects Command Axis (writes) & Command Assembly 0x02 Velocity
  - Byte 3 – selects Response Axis (reads) & Response Assembly 0x01 Actual Position
  - Bytes 4 & 5 loads data 0x00004E20 in hex (dec 20,000)

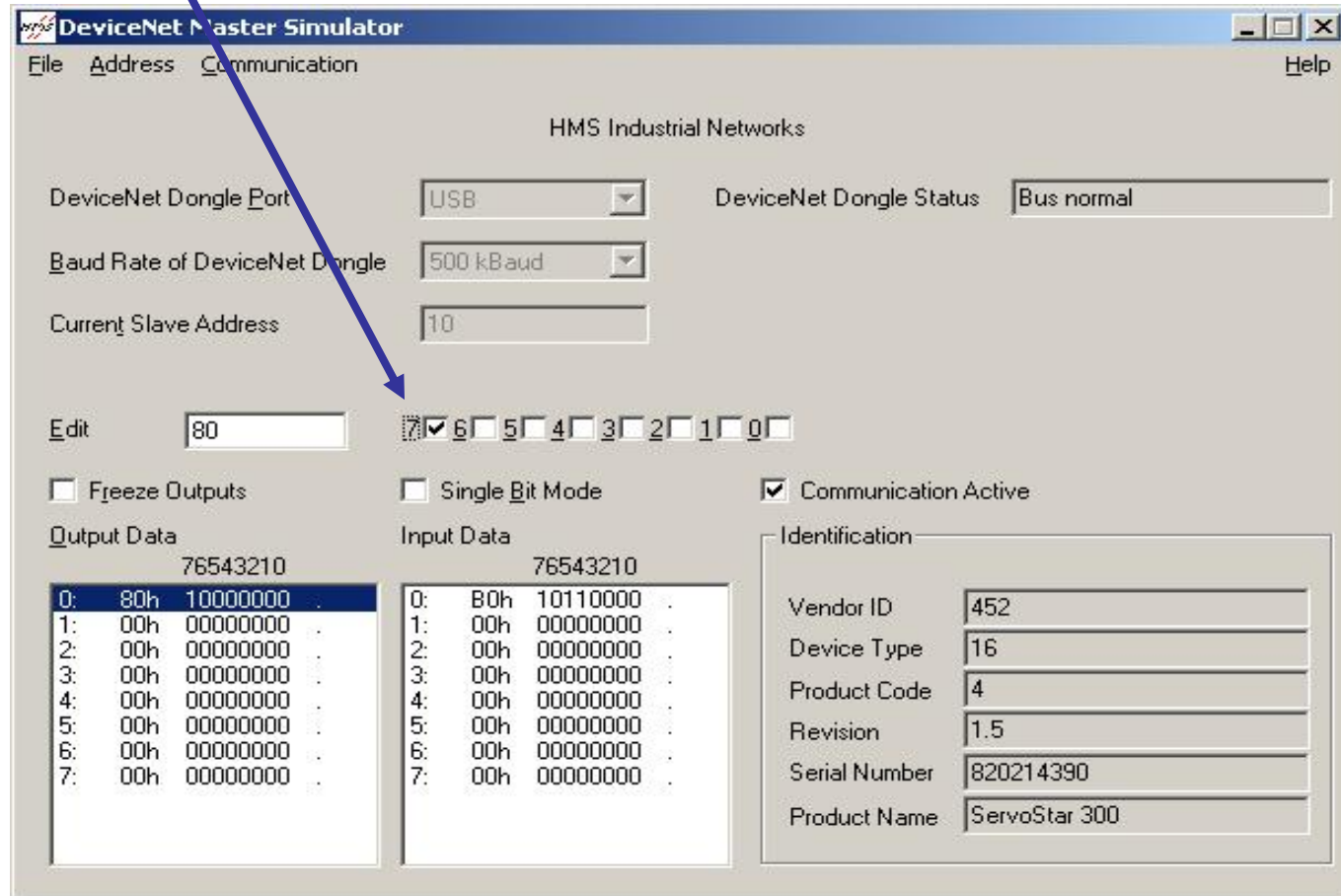
Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	1	0	0	0	0	0	0	1
1	0							
2	0	0	1	0	0	0	1	0
3	0	0	1	0	0	0	0	1
4	0x20							
5	0x4E							
6	0x00							
7	0x00							

- Polled I/O Screen
- Output Data
  - (Command Assembly)
- Input Data
  - (Response Assembly)
- Dongle Port
- Baud Rate
- Slave Address
  - Additional information
  - Vendor ID
  - Serial Number
  - Product Name



# Polled I/O – Enabling Drive

- Enable Drive = Byte 0, Bit 7 to 1

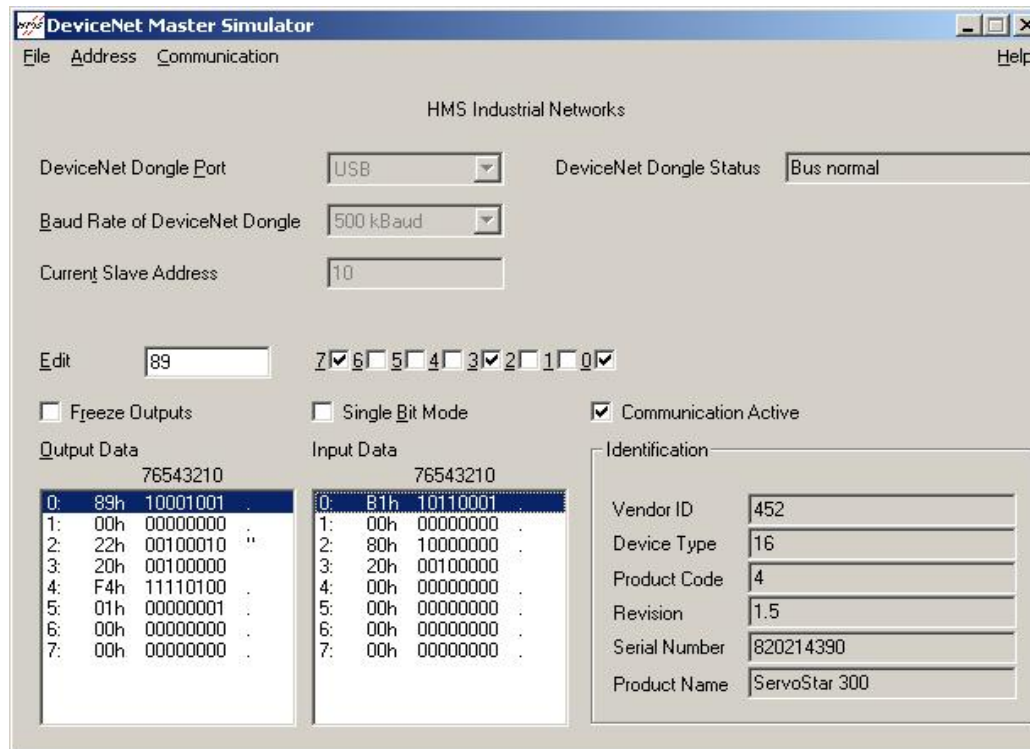


# Changing Velocity (Digital Velocity)

**Change Velocity and/or Direction on the fly using Polled I/O**

Opmode 0 (Digital Velocity)

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	1	0	0	0	1	0	0	1
1	0	0	0	0	0	0	0	0
2	0	0	1	0	0	0	1	0
3	0	0	1	0	0	0	0	0
4	1	1	1	1	0	1	0	0
5	0	0	0	0	0	0	0	1
6	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0



The screenshot shows the 'DeviceNet Master Simulator' window. The title bar includes 'File', 'Address', 'Communication', and 'Help'. The main area is titled 'HMS Industrial Networks' and contains several configuration fields: 'DeviceNet Dongle Port' (USB), 'DeviceNet Dongle Status' (Bus normal), 'Baud Rate of DeviceNet Dongle' (500 kBaud), and 'Current Slave Address' (10). Below these are bit control fields for bits 7 through 0, with checkboxes for each. The 'Communication Active' checkbox is checked. At the bottom, there are two data windows: 'Output Data' and 'Input Data', both showing hexadecimal and binary values for slave address 89h (10001001). The 'Identification' section on the right lists: Vendor ID (452), Device Type (16), Product Code (4), Revision (1.5), Serial Number (820214390), and Product Name (ServoStar 300).

# Monitoring Actual Position

## ■ Polled I/O

Actual Position

Read from GUI = 4998

Actual Position

Read from DeviceNet

00001387 hex

Input Data Bytes 4 & 5

The screenshot shows two software windows. The top window, 'Untitled - DriveGUI', displays motor parameters. The bottom window, 'DeviceNet Master Simulator', shows network configuration and data exchange.

Parameter	Value	Unit
Drive Pt Load	1	%
Motor Pt Load	0	%
Effective Current	0.029	A
- D Component	0.032	A
- Q Component	-0.016	A
Bus Voltage	180	V
Regen Power	0	W
Motor Thermistor Resistance	64	Ohms
Angle of Rotation	179.0	°mech
7Id a5	7Id a5	Counts
Actual Velocity	-0.125	rpm
Velocity Command	0.000	rpm
Position	4998	Counts

Byte	Hex	Binary
0	01h	00000001
1	00h	00000000
2	22h	00100010
3	21h	00100001
4	F4h	11110100
5	01h	00000001
6	00h	00000000
7	00h	00000000

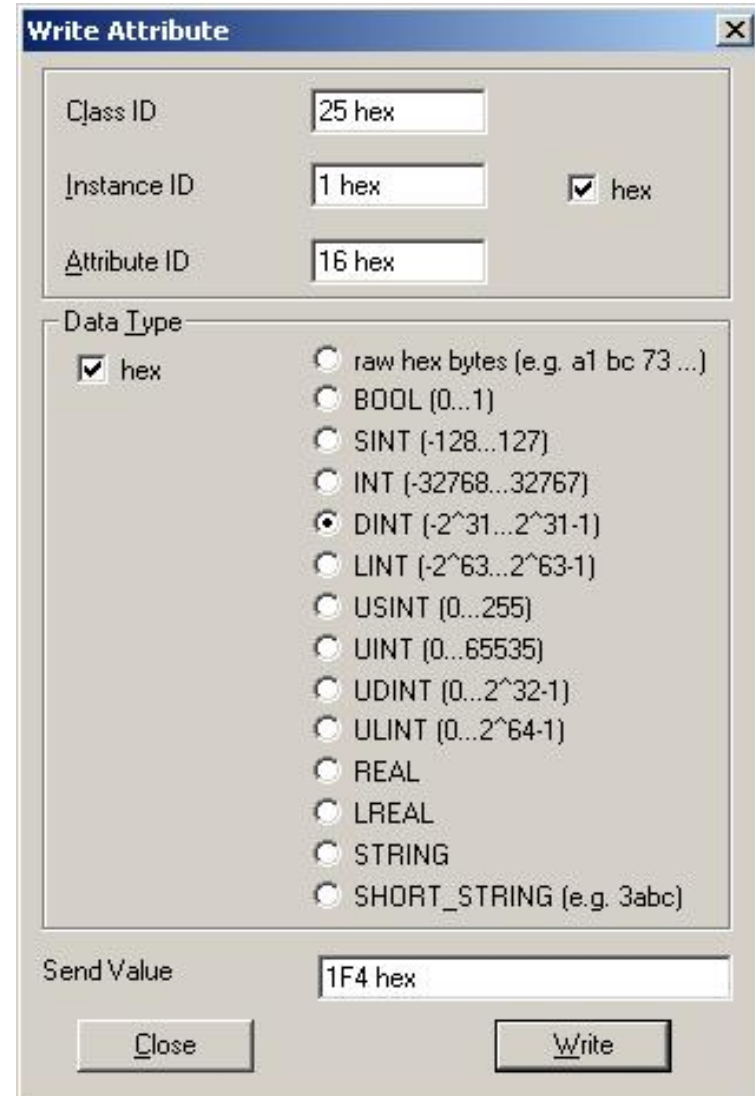
  

Field	Value
Vendor ID	452
Device Type	16
Product Code	4
Revision	1.5
Serial Number	820214390
Product Name	ServoStar 300

- Set Jog Velocity to 500 RPM  
Class 0x25  
Instance ID 1  
Attribute ID 0x16  
Explicit Message 0x01F4  
(Send Value)

Data Type Double Integer

- Write Attribute Screen
- Then set Attribute 0x0B to initiate the move.
  - Data field Boolean
  - Send Value 01.



Write Attribute

Class ID: 25 hex

Instance ID: 1 hex  hex

Attribute ID: 16 hex

Data Type

hex

raw hex bytes (e.g. a1 bc 73 ...)

BOOL (0..1)

SINT (-128...127)

INT (-32768...32767)

DINT (-2<sup>31</sup>...2<sup>31</sup>-1)

LINT (-2<sup>63</sup>...2<sup>63</sup>-1)

USINT (0...255)

UINT (0...65535)

UDINT (0...2<sup>32</sup>-1)

ULINT (0...2<sup>64</sup>-1)

REAL

LREAL

STRING

SHORT\_STRING (e.g. 3abc)

Send Value: 1F4 hex

Close Write

- Read Jog Velocity

Class 0x25

Instance ID 1

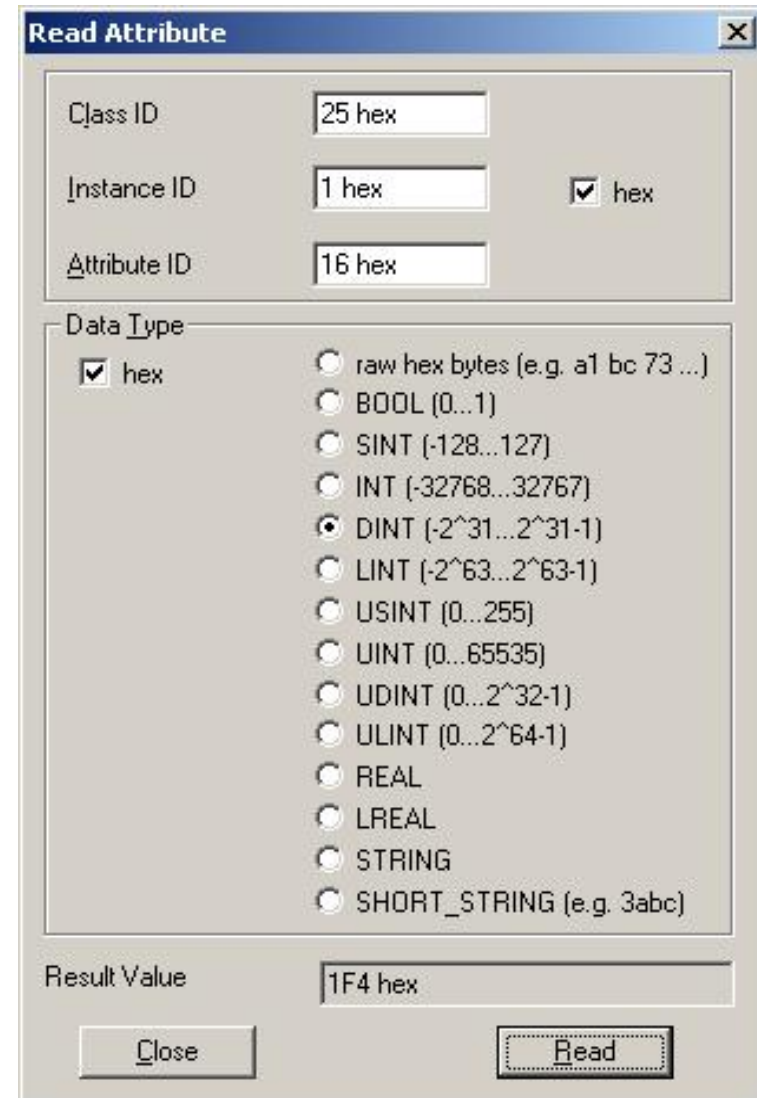
Attribute ID 0x16

Explicit Message 0x01F4  
(Result Value)

Data Type Double Integer

- Read Attribute Screen

- Result Value should be 1F4 hex
- (Value of Attribute 0x16)



**Read Attribute**

Class ID: 25 hex

Instance ID: 1 hex  hex

Attribute ID: 16 hex

Data Type

hex

raw hex bytes (e.g. a1 bc 73 ...)

BOOL (0..1)

SINT (-128..127)

INT (-32768..32767)

DINT (-2<sup>31</sup>..2<sup>31</sup>-1)

LINT (-2<sup>63</sup>..2<sup>63</sup>-1)

USINT (0..255)

UINT (0..65535)

UDINT (0..2<sup>32</sup>-1)

ULINT (0..2<sup>64</sup>-1)

REAL

LREAL

STRING

SHORT\_STRING (e.g. 3abc)

Result Value: 1F4 hex

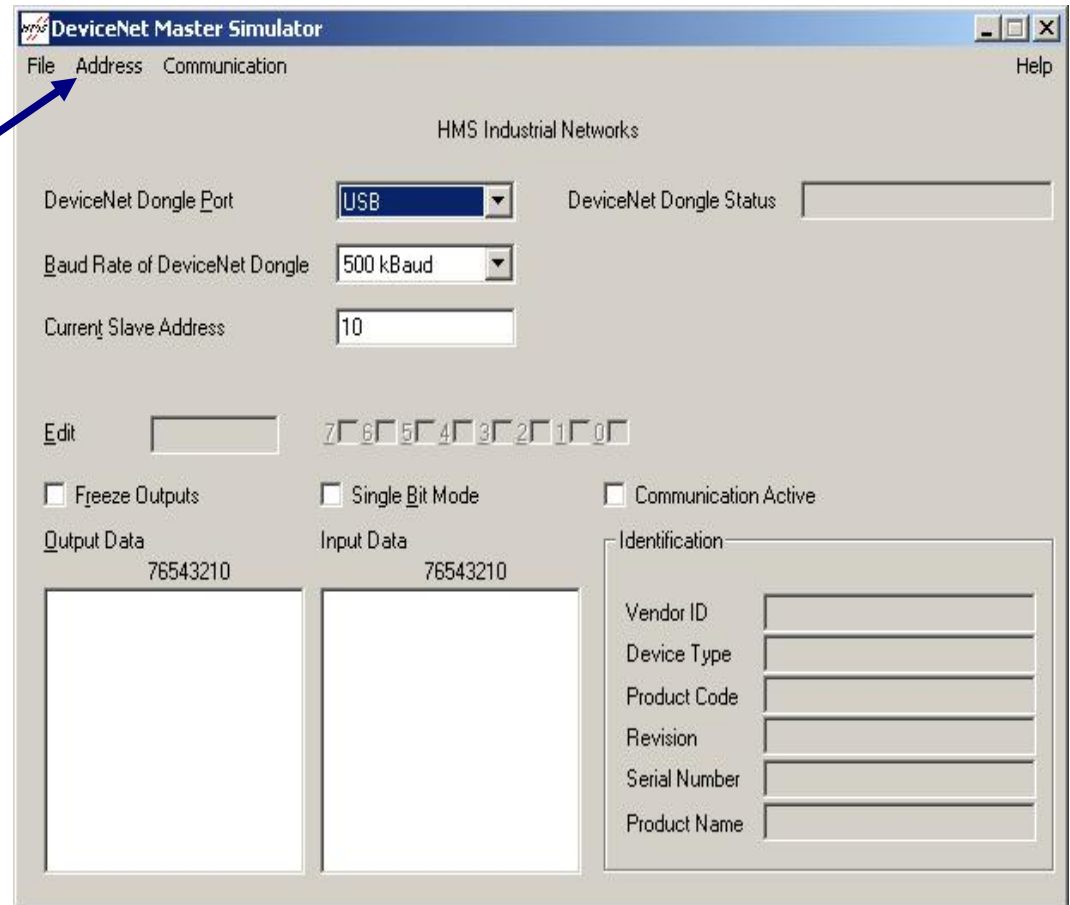
Close Read



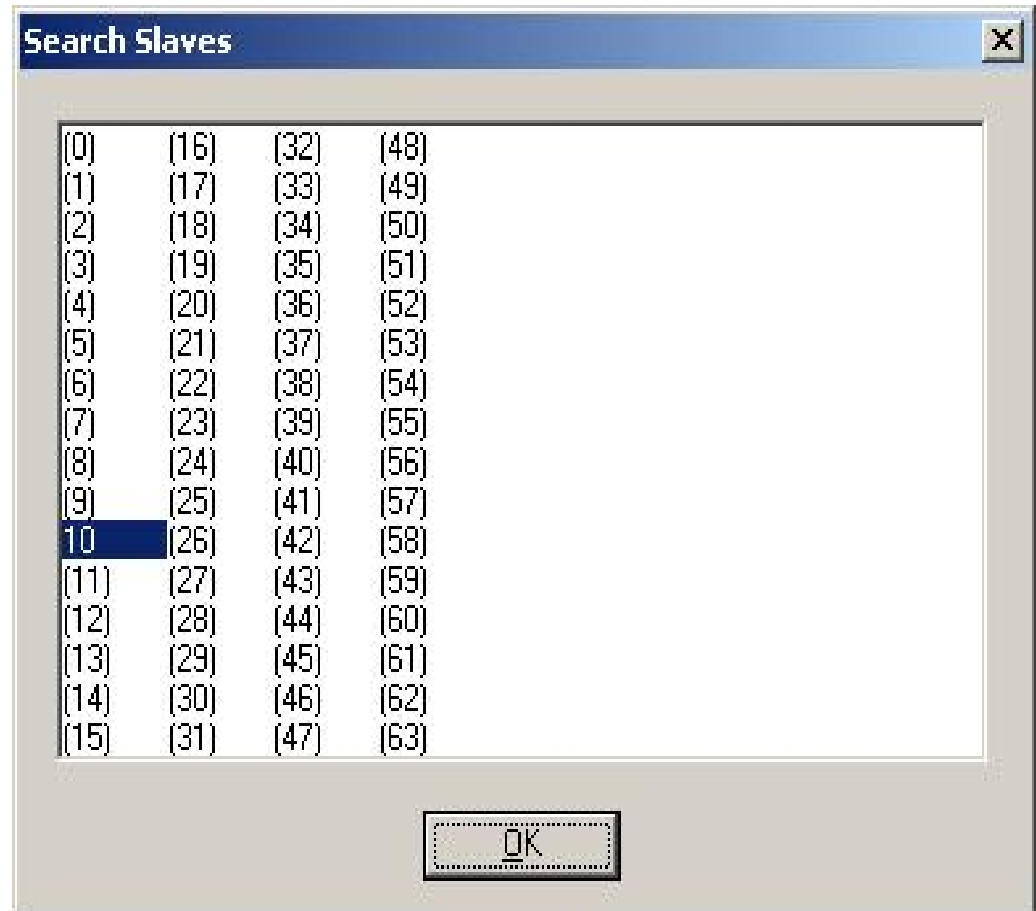
- **Position Controller Object: Class 0x25**
  - Most commonly used for Operation Modes (Torque, Velocity & Position) and configuring Motion
  
- **Parameter Object: Class 0x0F**
  - Vendor defined object for amplifier configuration. Uses DPR (dual port RAM) number from ASCII reference guide
  
- **Additional Objects**
  - Command Block : Class 0x27 – creating and storing Motion Tasks
  - Block Sequencer: Class 0x26 – executing Motion Tasks
  - Position Object: Class 0x24 – handles errors of amplifiers

- Start DeviceNet Master Simulator
- Click on Address Search for slaves

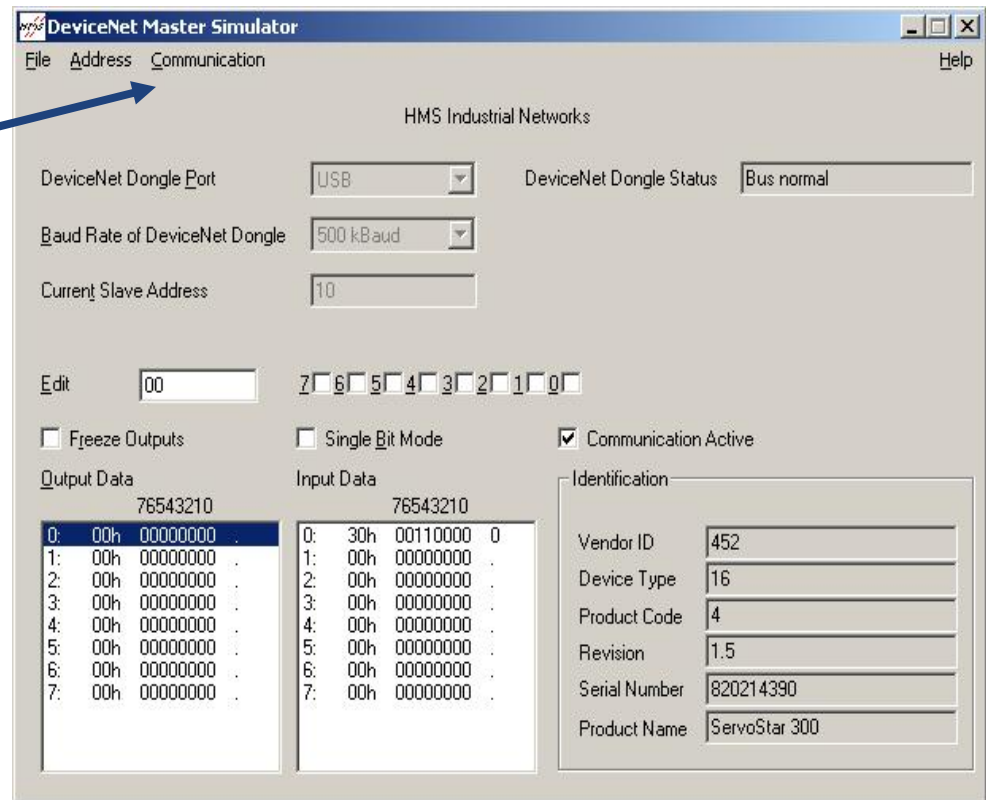
Note:  
CommPort  
Baud Rate  
Node Address



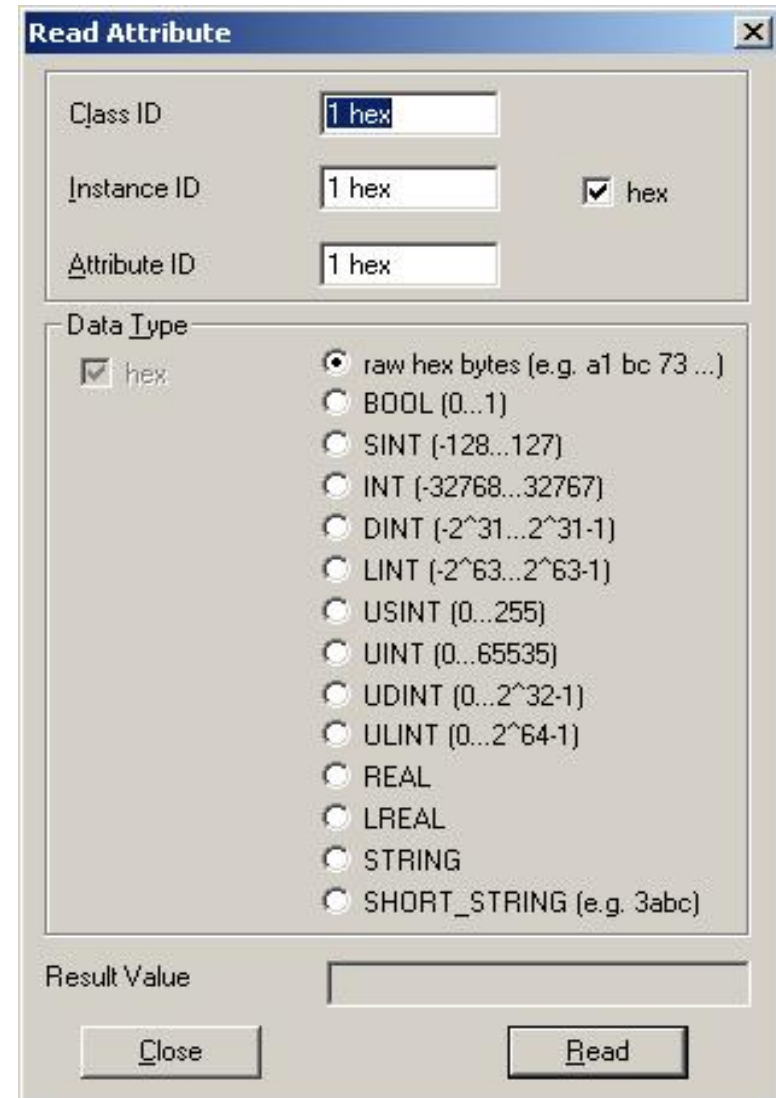
- Address 10 is identified  
(no parenthesis)



- Establish Communications
- Click on Communications
  - Start
  - Stop
  - Read Attribute
  - Write Attribute
- Polled I/O screen shown



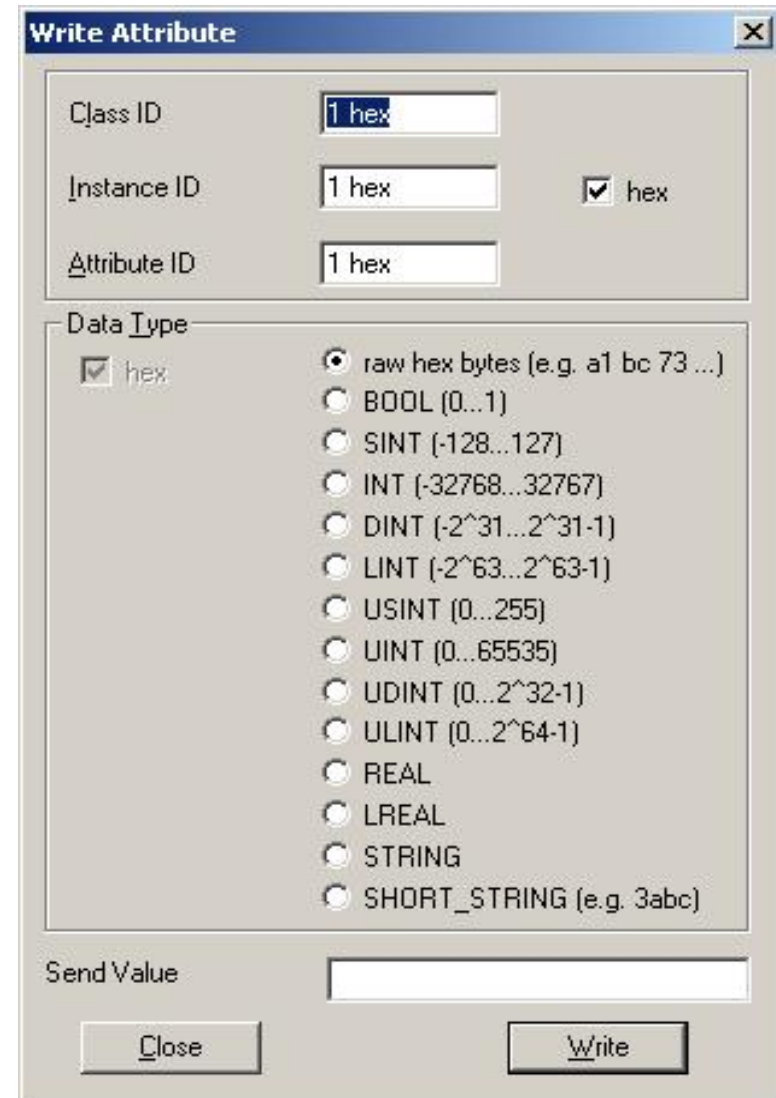
- Read Attribute screen shown



The image shows a 'Read Attribute' dialog box with the following fields and options:

- Class ID:** 1 hex
- Instance ID:** 1 hex  hex
- Attribute ID:** 1 hex
- Data Type:**
  - hex
  - raw hex bytes (e.g. a1 bc 73 ...)
  - BOOL (0...1)
  - SINT (-128...127)
  - INT (-32768...32767)
  - DINT (-2<sup>31</sup>...2<sup>31</sup>-1)
  - LINT (-2<sup>63</sup>...2<sup>63</sup>-1)
  - USINT (0...255)
  - UINT (0...65535)
  - UDINT (0...2<sup>32</sup>-1)
  - ULINT (0...2<sup>64</sup>-1)
  - REAL
  - LREAL
  - STRING
  - SHORT\_STRING (e.g. 3abc)
- Result Value:** [Empty text box]
- Buttons:** Close, Read

- Write Attribute screen shown



**Write Attribute**

Class ID: 1 hex

Instance ID: 1 hex  hex

Attribute ID: 1 hex

Data Type

hex

- raw hex bytes (e.g. a1 bc 73 ...)
- BOOL (0..1)
- SINT (-128..127)
- INT (-32768..32767)
- DINT (-2<sup>31</sup>..2<sup>31</sup>-1)
- LINT (-2<sup>63</sup>..2<sup>63</sup>-1)
- USINT (0..255)
- UINT (0..65535)
- UDINT (0..2<sup>32</sup>-1)
- ULINT (0..2<sup>64</sup>-1)
- REAL
- LREAL
- STRING
- SHORT\_STRING (e.g. 3abc)

Send Value:

## ■ Changing Opmodes

- Class 0x25
- Instance 0x01
- Attribute 0x03
- Explicit Message 0x00  
(Send Value)

## ■ Data Type Short Integer

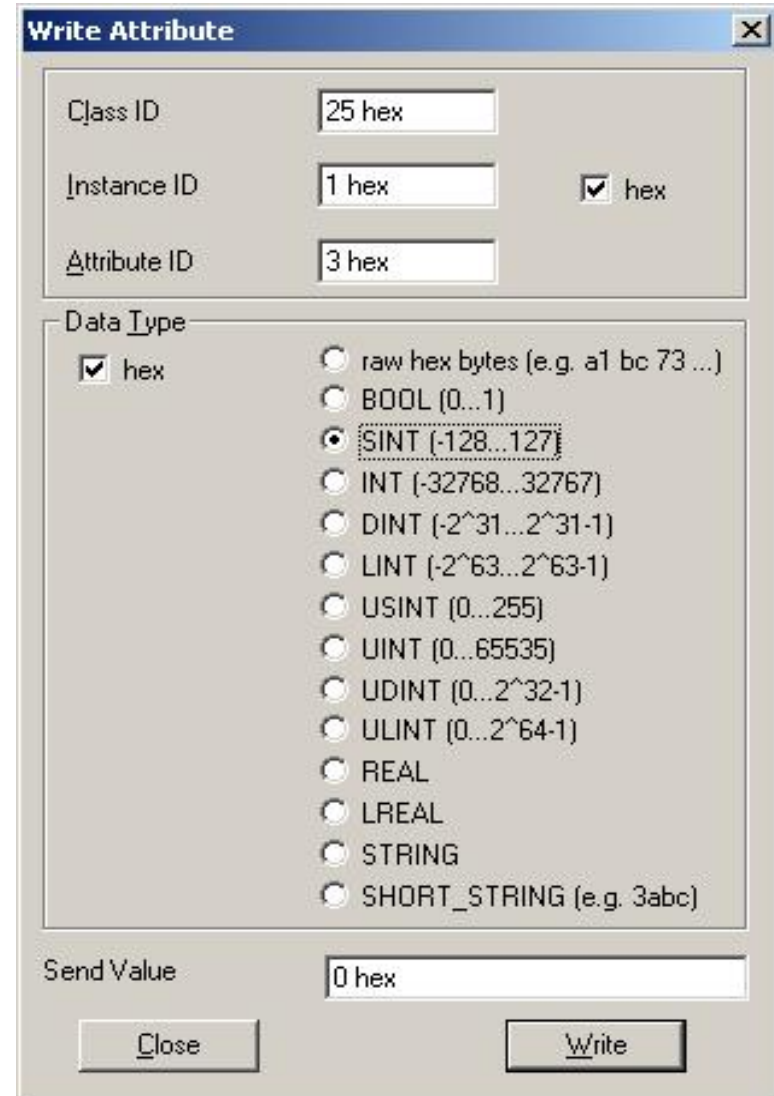
Position Mode = 0x00

Velocity Mode = 0x01

Torque Mode = 0x02

Read only = 0x03

View S300 GUI as you enter Send Values



**Write Attribute**

Class ID: 25 hex

Instance ID: 1 hex  hex

Attribute ID: 3 hex

Data Type

hex

raw hex bytes (e.g. a1 bc 73 ...)

BOOL (0..1)

SINT (-128..127)

INT (-32768..32767)

DINT (-2<sup>31</sup>...2<sup>31</sup>-1)

LINT (-2<sup>63</sup>...2<sup>63</sup>-1)

USINT (0..255)

UINT (0..65535)

UDINT (0...2<sup>32</sup>-1)

ULINT (0...2<sup>64</sup>-1)

REAL

LREAL

STRING

SHORT\_STRING (e.g. 3abc)

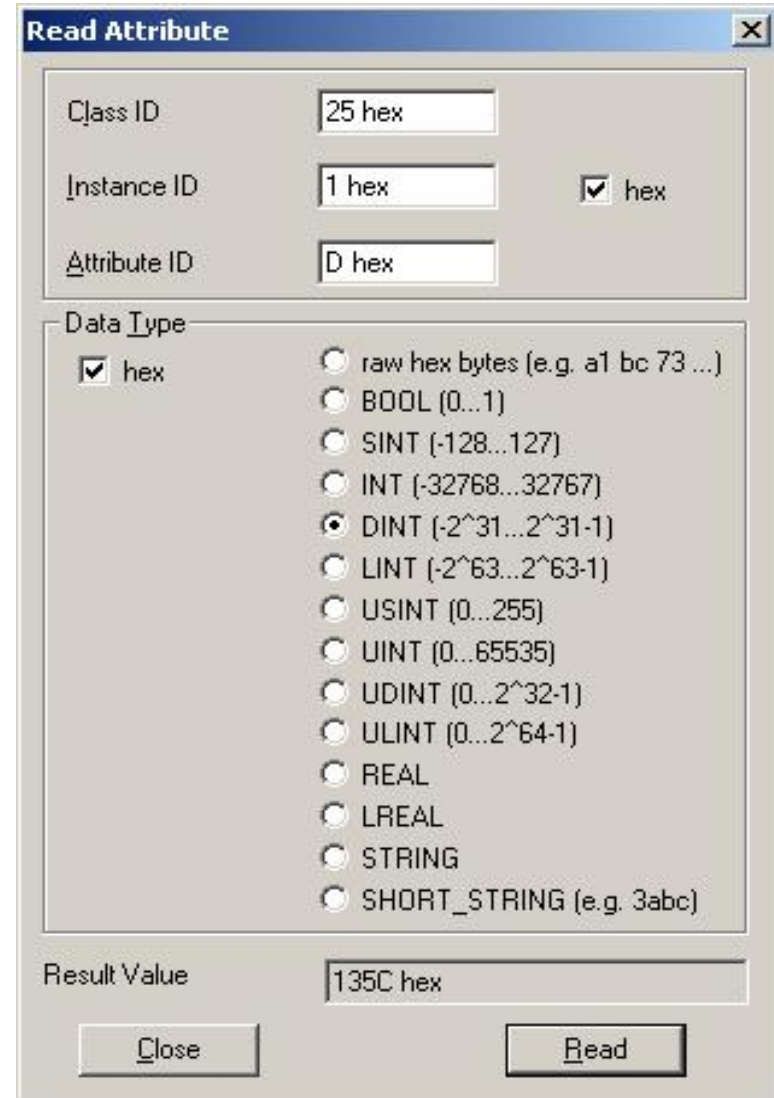
Send Value: 0 hex

## ■ Reading Actual Position

- Class 0x25
- Instance 0x01
- Attribute 0x0D
- Explicit Message  
(Result Value)

- Data Type Double Integer

Return Result is 0x135C  
= 4956 counts (verify with GUI)



Read Attribute

Class ID: 25 hex

Instance ID: 1 hex  hex

Attribute ID: D hex

Data Type

hex

raw hex bytes (e.g. a1 bc 73 ...)

BOOL (0..1)

SINT (-128...127)

INT (-32768...32767)

DINT (-2<sup>31</sup>...2<sup>31</sup>-1)

LINT (-2<sup>63</sup>...2<sup>63</sup>-1)

USINT (0...255)

UINT (0...65535)

UDINT (0...2<sup>32</sup>-1)

ULINT (0...2<sup>64</sup>-1)

REAL

LREAL

STRING

SHORT\_STRING (e.g. 3abc)

Result Value: 135C hex

Close Read



- Most drive parameters can be read or written using 0x0F
  - Instance ID = DPR from GUI
  - Attribute ID = 0x01

Note: Most Objects use Instance 0x01 and special Attribute number. Parameter Object uses Attribute 0x01 and Instance corresponding to DPR from GUI

- Only DPR 1 – 254 can be used in Instance, since Instance is 1 byte
  - For DPR numbers greater than 254 we use Instance 255, Attribute 0x64 (Instance 255 to be discussed later)
- 
- Response time < 500 mSec  
(all other objects < 50mSec)

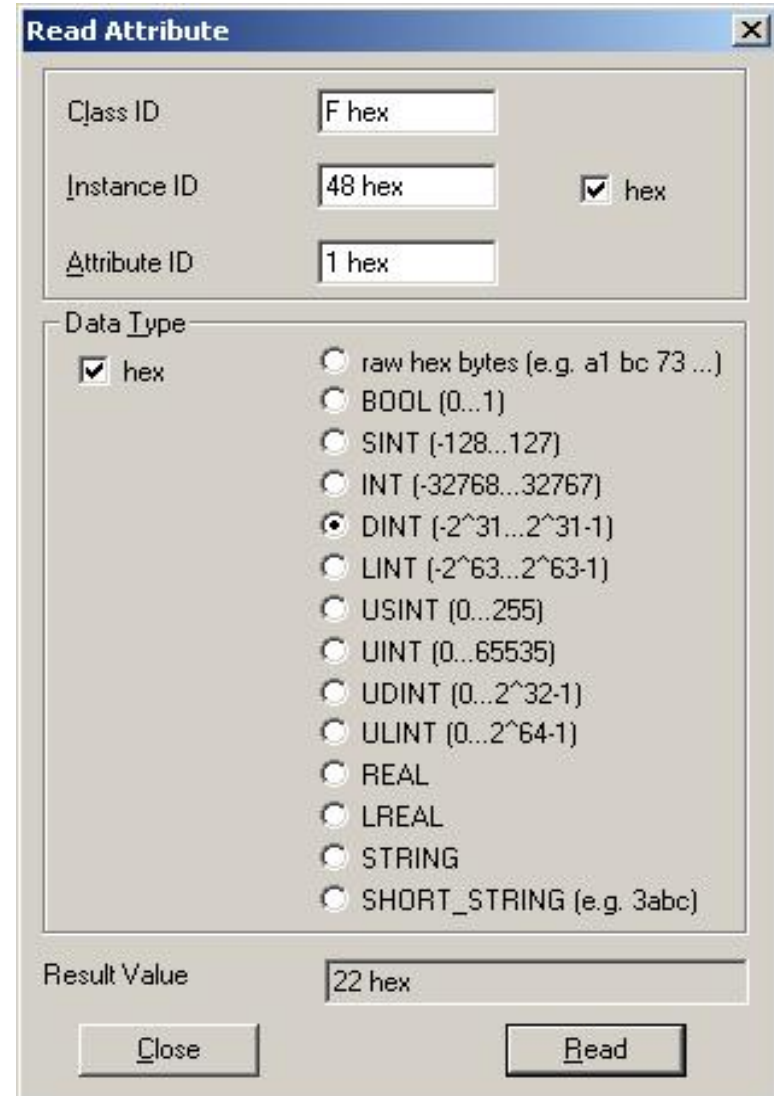
- Example:
  - Move Home (MH)  
DPR = 141 (dec)
  - Class ID = 0x0F
  - Instance ID = 0x8D
  - Attribute ID = 0x01
  - Explicit Message = 0x01  
(Send Value)

ASCII - Command	MH		
Syntax Transmit	MH		
Syntax Receive	MH	Available in	
Type	Command	MMI	Yes
ASCII Format	-	CAN (CoE) Object Number	358D (hex)
DIM	-	PROFIBUS PNU	1741 (dec) IND = 1 (dec)
Range	-	DPR	141 (dec)
Default	-	SERCOS IDN	
Opmode	8	Data Type Bus/DPR	-
Drive State	Enabled	Weighting	
Start Firmware	1.0		
Configuration	No	Last Change of this Object	1.0
Function Group	Position Controller	EEPROM	-
Short Description	Start Homing		

- Reading Velocity Gain GV
  - Class 0x0F
  - Instance 0x48 (DPR value 72 dec)
  - Attribute 0x1
  - Explicit Message (Result Value)
- Data Type Double Integer

Return Result is 0x22  
= 34 dec

Note: Divide Device Net result by  
1000 to obtain GUI result  
 $34 / 1000 = 0.034$   
(GUI has Weighting listed)



Read Attribute

Class ID: F hex

Instance ID: 48 hex  hex

Attribute ID: 1 hex

Data Type

hex

raw hex bytes (e.g. a1 bc 73 ...)

BOOL (0..1)

SINT (-128...127)

INT (-32768...32767)

DINT (-2<sup>31</sup>...2<sup>31</sup>-1)

LINT (-2<sup>63</sup>...2<sup>63</sup>-1)

USINT (0...255)

UINT (0...65535)

UDINT (0...2<sup>32</sup>-1)

ULINT (0...2<sup>64</sup>-1)

REAL

LREAL

STRING

SHORT\_STRING (e.g. 3abc)

Result Value: 22 hex

Close Read

- ASCII object reference
- GV Proportional gain Velocity Control Loop
  - DPR 72 (dec) 0x48 hex
  - Weighting = 1000

ASCII - Command	<b>GV</b>		
Syntax Transmit	GV [Data]		
Syntax Receive	GV <Data>	Available in	
Type	Variable rw	MMI	Yes
ASCII Format	Float	CAN (CoE) Object Number	3548 (hex)
DIM	-	PROFIBUS PNU	1672 (dec) IND = 1 (dec)
Range	0.001 ... 369.2	DPR	72 (dec)
Default	0.046	SERCOS IDN	100
Opmode	0, 1	Data Type Bus/DPR	Integer32
Drive State	-	Weighting	1000
Start Firmware	1.0		
Configuration	No	Last Change of this Object	1.0
Function Group	Velocity Controller	EEPROM	Yes
Short Description	Velocity Control Loop: Proportional Gain		

# Reading GV Proportional Gain

Note  
Displays:

GUI = 0.034

DeviceNet  
= 0x22  
(34 dec)

The screenshot shows the DriveGUI interface with a 'Read Attribute' dialog box open. The dialog box displays the following information:

- Class ID: F hex
- Instance ID: 48 hex (checked hex)
- Attribute ID: 1 hex
- Data Type:  hex,  raw hex bytes (e.g. a1 bc 73 ...),  BOOL (0...1),  SINT (-128...127),  INT (-32768...32767),  DINT (-2<sup>31</sup>...2<sup>31</sup>-1),  LINT (-2<sup>63</sup>...2<sup>63</sup>-1),  USINT (0...255),  UINT (0...65535),  UDINT (0...2<sup>32</sup>-1),  ULINT (0...2<sup>64</sup>-1),  REAL,  LREAL,  STRING,  SHORT\_STRING (e.g. 3abc)
- Result Value: 22 hex

Blue arrows point from the text on the left to the 'GV' entry in the 'User-defined Variables to monitor' table and the 'Result Value' field in the dialog box.

User-defined Variables to monitor			
ASCII Cmd.	Value	ASCII Cmd.	Value
ACC	293750	GV	0.034

Analog Inputs	
Input 1	Input 2
163 mV	622 mV

Digital Inputs / Outputs							
IN1	IN2	IN3	IN4	Enable	OUT1	OUT2	AS
<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	AS

- **Reading or Writing for DPR greater than 254 can be accomplished through Attribute 0x64: Parameter Number**
- **Load the desired DPR number into the Parameter Number Attribute, then use the Parameter Object Instance 255 (0xFF) to access the Parameter.**
- **Note since this is a Class Attribute (0x0F) use Instance 0 when setting the Parameter Number.**

**It really is easy. Just an extra step.**

# Attribute 0x64: Parameter Number

- Example to Read VLIMP:
  - VLIM DPR = 122 (290 DEC)

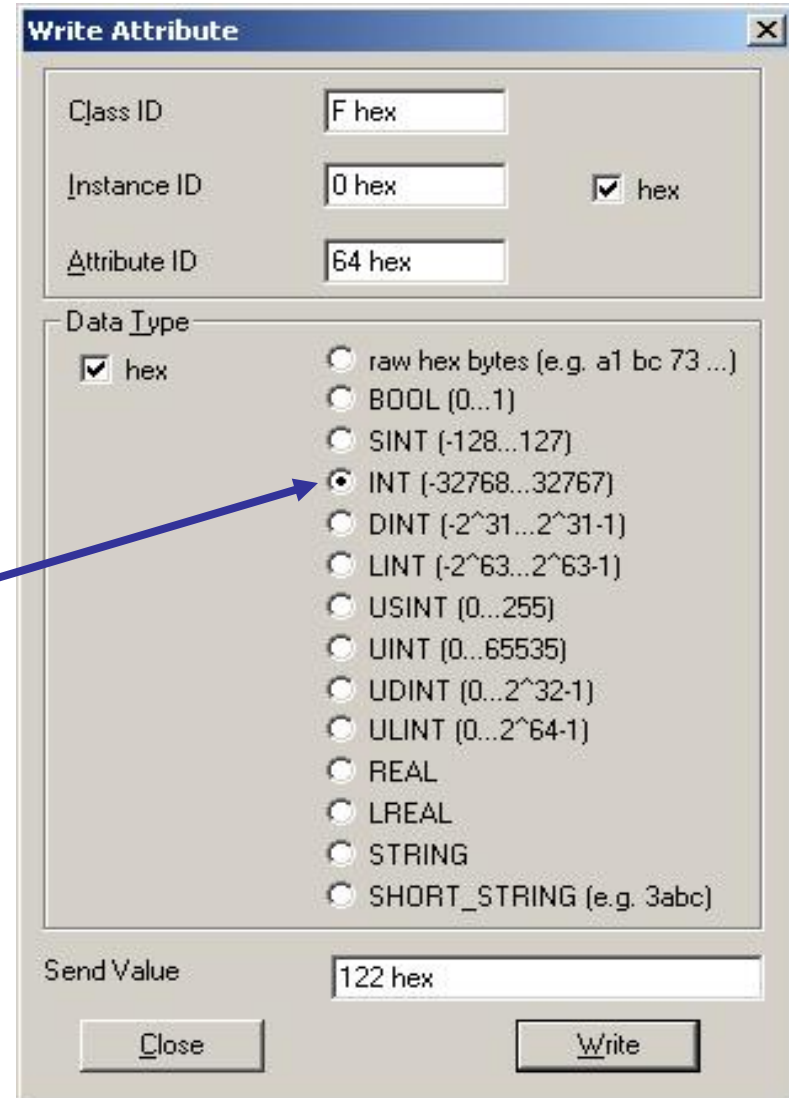
Set up to read value first

- Class ID = 0x0F
  - Instance ID = 0x00
  - Attribute ID = 0x64
  - Explicit Message = 0x122  
(Send Value)
- The Explicit Message tells the device that DPR 0x122 is the parameter of the drive we are interested in reading or writing.

ASCII - Command	<b>VLIMP</b>		
Syntax Transmit	VLIM [Data]		
Syntax Receive	VLIMP <Data>	Available in	
Type	Variable rw	MMI	Yes
ASCII Format	Float	CAN (CoE) Object Number	3622 (hex)
DIM	(> VUNIT)	PROFIBUS PNU	1890 (dec) IND = 1 (dec)
Range	0.0 .. MSPEED	DPR	290 (dec)
Default	3000	SERCOS IDN	38
Opmode	0, 1	Data Type Bus/DPR	Integer32
Drive State	-	Weighting	1000
Start Firmware	1.0		
Configuration	No	Last Change of this Object	1.0
Function Group	Velocity Controller	EEPROM	Yes
Short Description	Max. Velocity		

# Setting Parameter Number VLIMP

- Using Attribute 0x64 to set Parameter we are interested in viewing:
  - Class 0x0F
  - Instance 0x00
  - Attribute 0x64
  - Explicit Message 0x122
    - DPR 290 = VLIMP (Send Value)
- Data Type Integer



**Write Attribute**

Class ID: F hex

Instance ID: 0 hex  hex

Attribute ID: 64 hex

Data Type

hex

- raw hex bytes (e.g. a1 bc 73 ...)
- BOOL (0...1)
- SINT (-128...127)
- INT (-32768...32767)
- DINT (-2<sup>31</sup>...2<sup>31</sup>-1)
- LINT (-2<sup>63</sup>...2<sup>63</sup>-1)
- USINT (0...255)
- UINT (0...65535)
- UDINT (0...2<sup>32</sup>-1)
- ULINT (0...2<sup>64</sup>-1)
- REAL
- LREAL
- STRING
- SHORT\_STRING (e.g. 3abc)

Send Value: 122 hex

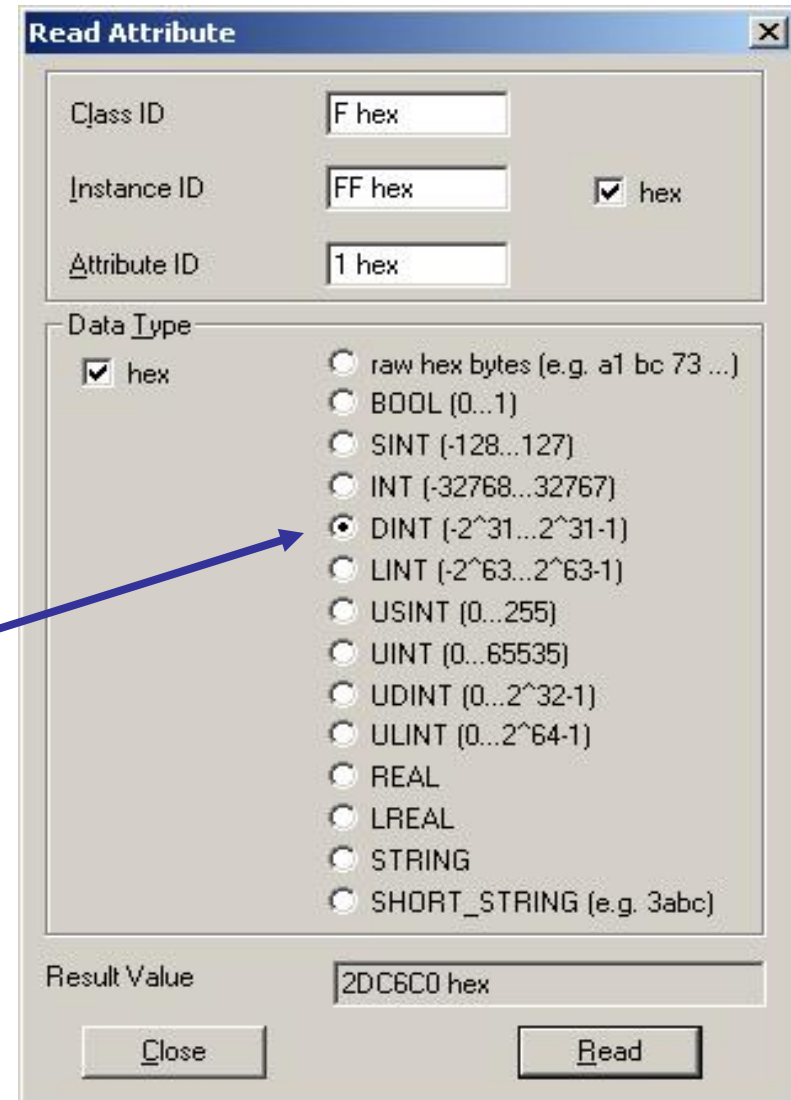


- Using Instance 255 (0xFF) read Parameter we are interested in viewing:

- Class 0x0F
- Instance 0xFF
- Attribute 0x01
- Explicit Message 0x2DC6C0  
Dec = 3,000,000  
(Result Value)

- Data Type Double Integer

Remember value is weighted by 1000,  
so VLIMP GUI READS 3000



**Read Attribute**

Class ID: F hex

Instance ID: FF hex  hex

Attribute ID: 1 hex

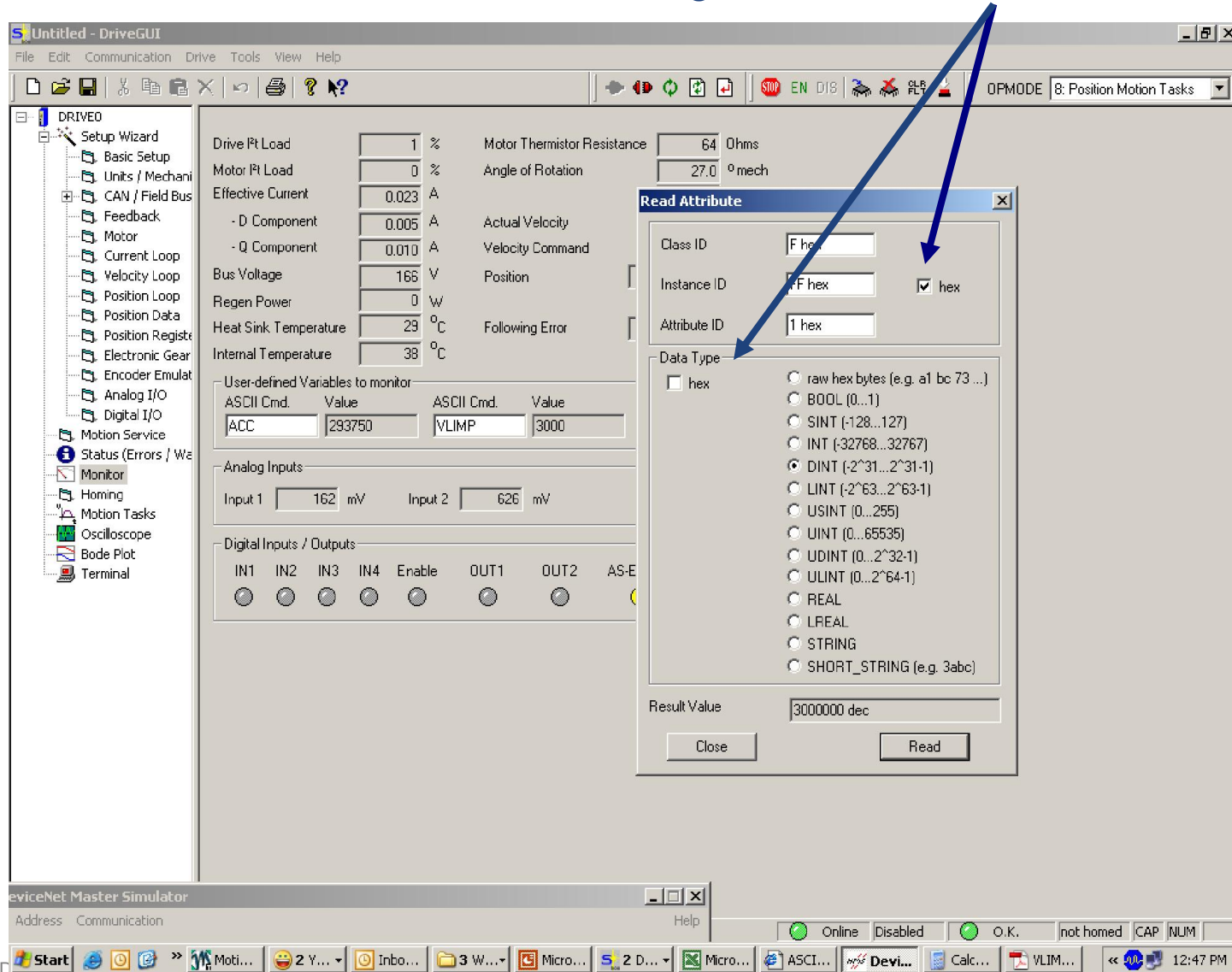
Data Type

hex

- raw hex bytes (e.g. a1 bc 73 ...)
- BOOL (0...1)
- SINT (-128...127)
- INT (-32768...32767)
- DINT (-2<sup>31</sup>...2<sup>31</sup>-1)
- LINT (-2<sup>63</sup>...2<sup>63</sup>-1)
- USINT (0...255)
- UINT (0...65535)
- UDINT (0...2<sup>32</sup>-1)
- ULINT (0...2<sup>64</sup>-1)
- REAL
- LREAL
- STRING
- SHORT\_STRING (e.g. 3abc)

Result Value: 2DC6C0 hex

- Note that DeviceNet Simulator allows to change from hex to dec



The screenshot shows the DriveGUI interface with a 'Read Attribute' dialog box open. The dialog box contains the following information:

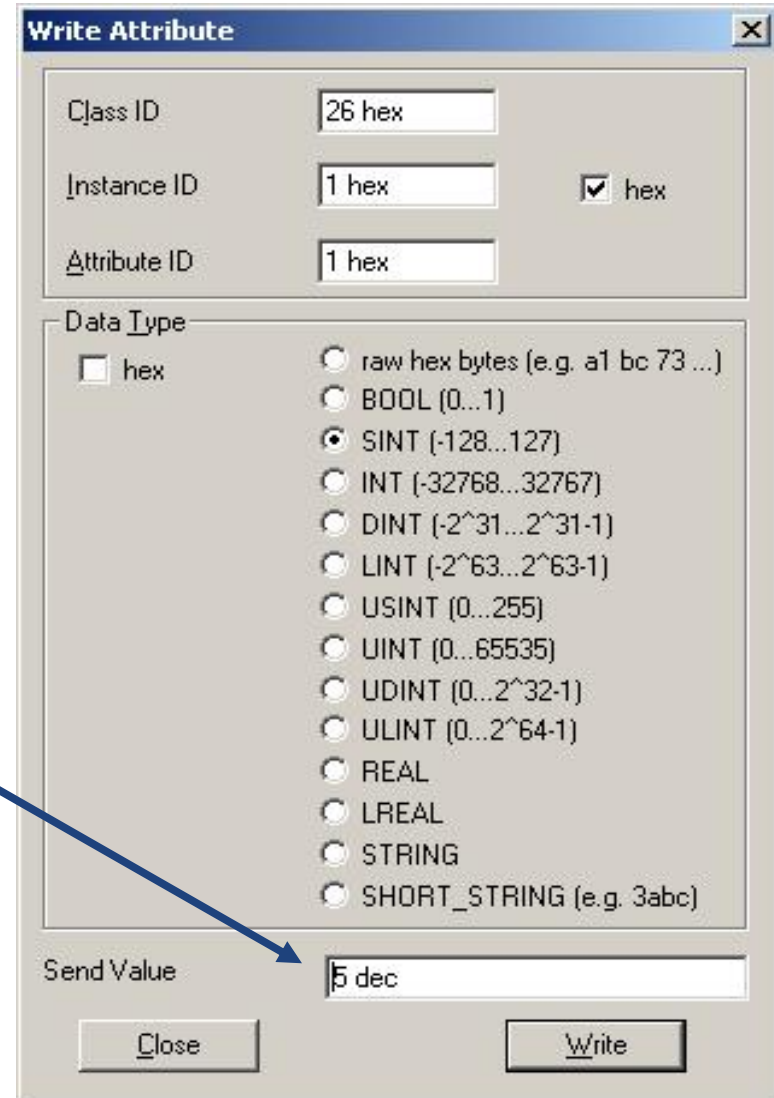
- Class ID: F hex
- Instance ID: FF hex
- Attribute ID: 1 hex
- Data Type:  hex
- Result Value: 3000000 dec

The background interface shows various motor parameters and a tree view on the left. A blue arrow points from the top right of the dialog box to the 'hex' checkbox.

## ■ Block Sequencer Object

- Class 0x26
- Instance 0x01
- Attribute 0x01
- Explicit Message 0x05 (Send Value)

This determines the Block Number (Motion Task) to Execute



**Write Attribute**

Class ID: 26 hex

Instance ID: 1 hex  hex

Attribute ID: 1 hex

Data Type

hex

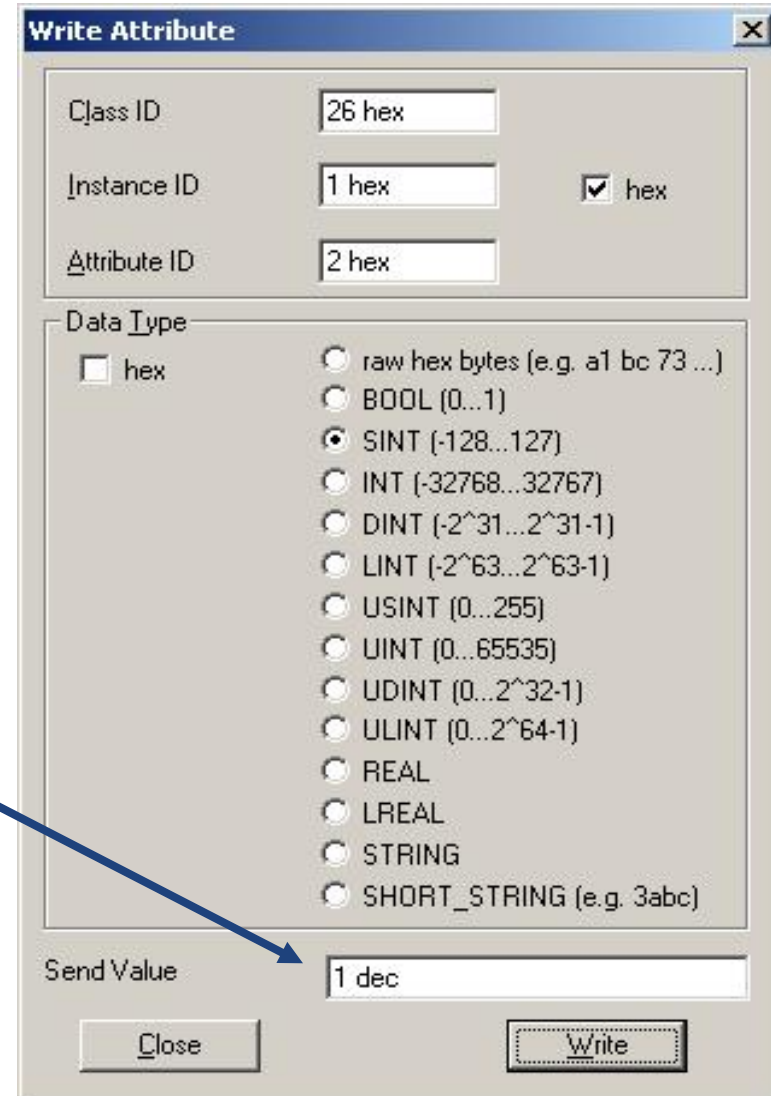
- raw hex bytes (e.g. a1 bc 73 ...)
- BOOL (0...1)
- SINT (-128...127)
- INT (-32768...32767)
- DINT (-2<sup>31</sup>...2<sup>31</sup>-1)
- LINT (-2<sup>63</sup>...2<sup>63</sup>-1)
- USINT (0...255)
- UINT (0...65535)
- UDINT (0...2<sup>32</sup>-1)
- ULINT (0...2<sup>64</sup>-1)
- REAL
- LREAL
- STRING
- SHORT\_STRING (e.g. 3abc)

Send Value: 5 dec

## ■ Block Sequencer Object

- Class 0x26
- Instance 0x01
- Attribute 0x02
- Explicit Message 0x01 (Send Value)

This Executes the Block Number (Motion Task) identified from Attribute 0x01

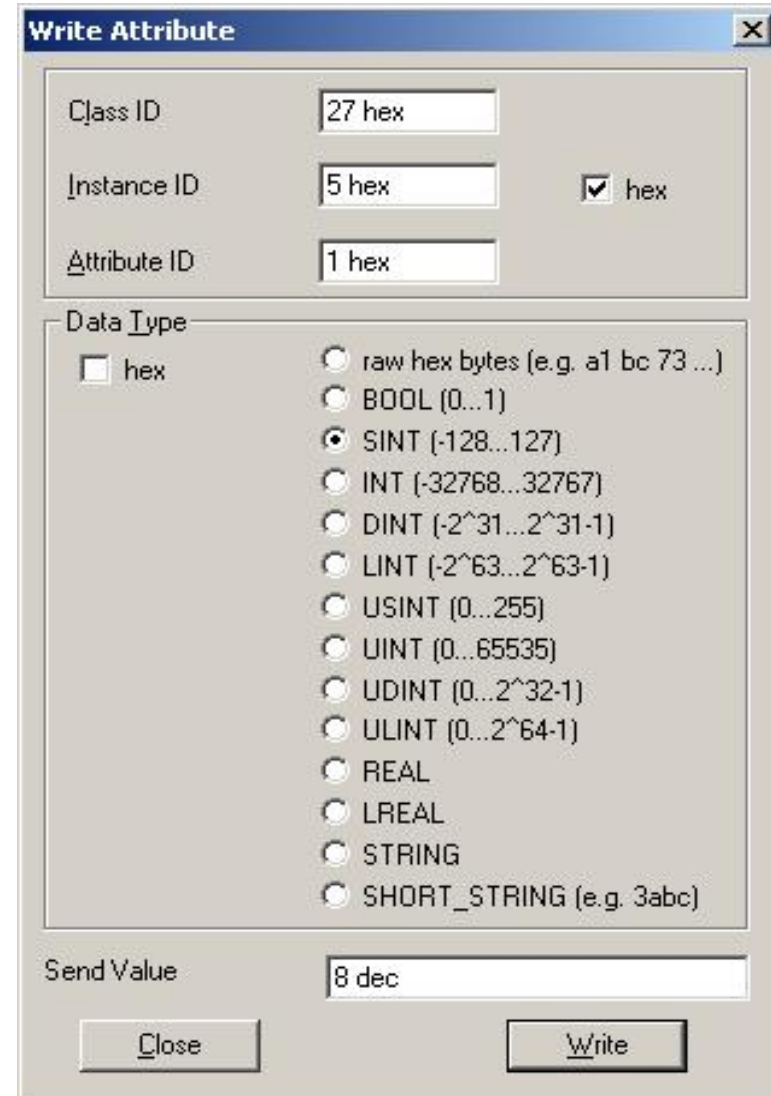


- Command Block Object
- Class 0x27
- Command 0x08 Motion Task

**REMEMBER TO DISABLE  
DRIVE!!**

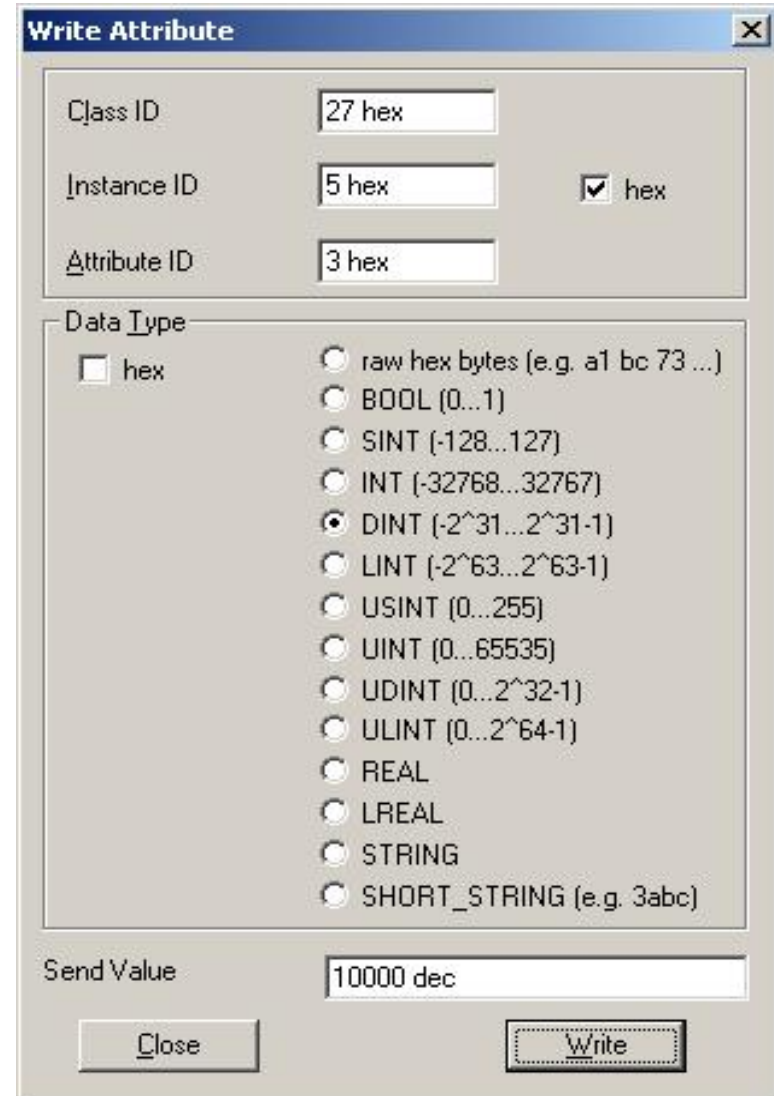
- Class 0x27
- Instance 0x05
- Attribute 0x01
- Explicit Message 0x08  
(Send Value)

Instance = Block No. (Motion Task)  
0x08 is Command Block – Motion  
Task



- Command Block Object Class 0x27
  - Keep Class 0x27
  - Instance = Block Number
  - Attributes for Profile
  - Explicit Message = Values for Task
- Class 0x27
- Instance 0x05
- Attribute 0x03 (Target Position)
- Explicit Message (Send Value)

**REMEMBER TO DISABLE DRIVE!!**



Write Attribute

Class ID: 27 hex

Instance ID: 5 hex  hex

Attribute ID: 3 hex

Data Type

hex

raw hex bytes (e.g. a1 bc 73 ...)

BOOL (0...1)

SINT (-128...127)

INT (-32768...32767)

DINT (-2<sup>31</sup>...2<sup>31</sup>-1)

LINT (-2<sup>63</sup>...2<sup>63</sup>-1)

USINT (0...255)

UINT (0...65535)

UDINT (0...2<sup>32</sup>-1)

ULINT (0...2<sup>64</sup>-1)

REAL

LREAL

STRING

SHORT\_STRING (e.g. 3abc)

Send Value: 10000 dec

Close Write

- Command Block Object  
Class 0x27

- Attributes

0x03 = Target Position

0x04 = Target Velocity

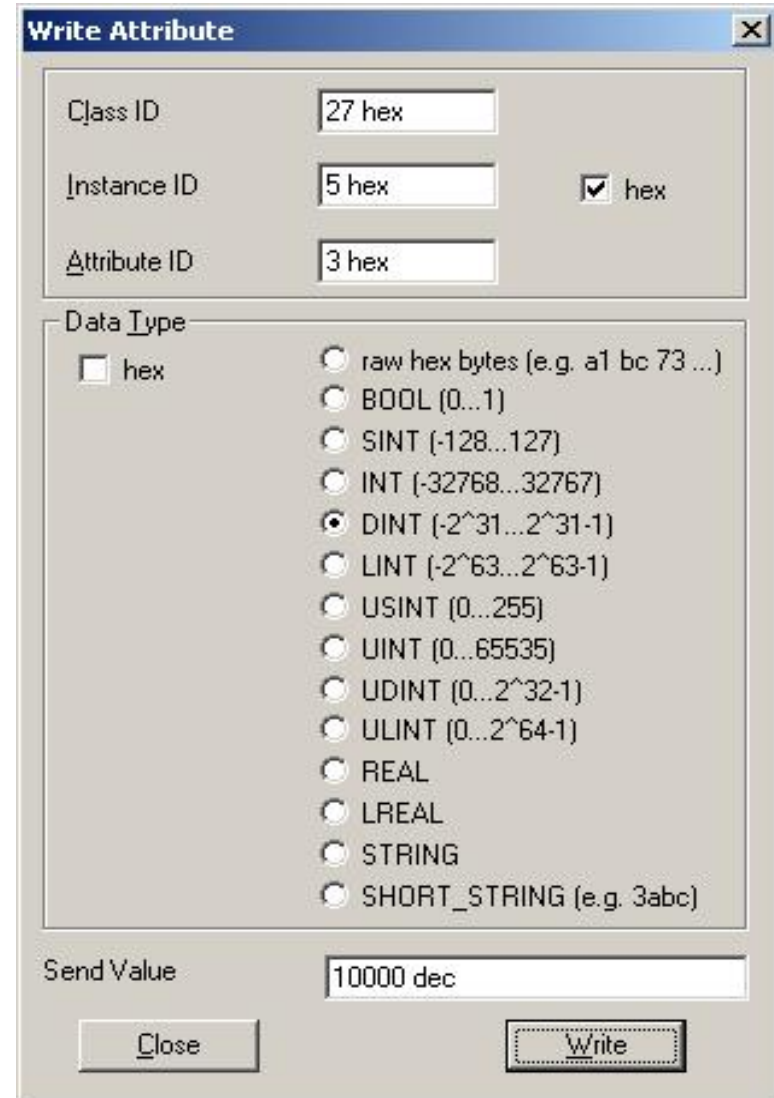
0x05 = Incremental Move

0x65 = Acceleration

0x66 = Deceleration

Note: 0x65 & 0x66 not supported in  
S600. Use Class 0x25 Attributes  
0x08 & 0x09

**REMEMBER TO DISABLE DRIVE!!**



Write Attribute

Class ID: 27 hex

Instance ID: 5 hex  hex

Attribute ID: 3 hex

Data Type

hex

raw hex bytes (e.g. a1 bc 73 ...)

BOOL (0...1)

SINT (-128...127)

INT (-32768...32767)

DINT (-2<sup>31</sup>...2<sup>31</sup>-1)

LINT (-2<sup>63</sup>...2<sup>63</sup>-1)

USINT (0...255)

UINT (0...65535)

UDINT (0...2<sup>32</sup>-1)

ULINT (0...2<sup>64</sup>-1)

REAL

LREAL

STRING

SHORT\_STRING (e.g. 3abc)

Send Value: 10000 dec

Close Write

- Take a look at the Motion Task Screen as you enter Values
- Now take a look at Terminal Screen
  - Enter Order # (Motion Task number)

What is the response?

- Now go back to Motion Task Table

What values appear in the Table?

- Now try initiating the Motion Task with Block Sequencer



## ■ Set up Scan List






**Dnsw32 - Interface Setup**

CardName **5136-DNP-PCI-2-1**

Overlap Enabled

Ok Cancel

**Dnsw32 - Edit Scan List**

**Device**

Mac Id  Vendor Id

Device Type (hex)  Product Code (hex)

Explicit  G3 Explicit Only

**I/O Configuration**

Poll  Cyclic  Ack Suppress

Strobe  Change-of-State

**Explicit Buffer**

Explicit Size  Exp. Offset (hex)

**I/O Connection 1**

Input Size  Input Offset (hex)

Output Size  Output Offset (hex)

**I/O Connection 2**

Input Size  Input Offset (hex)

Output Size  Output Offset (hex)

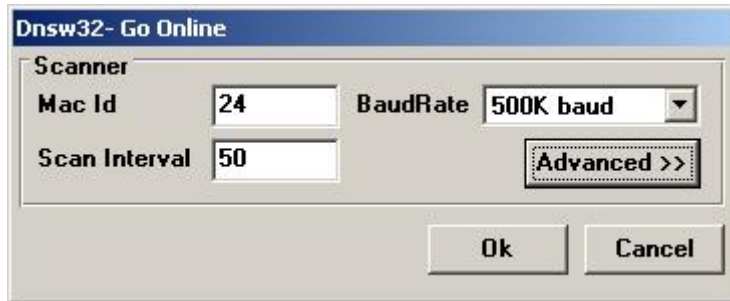
**COS Setup**

Production Inhibit Time (ms)

I/O 1 Interval  I/O 2 Interval

Ok Cancel

## ■ On Line

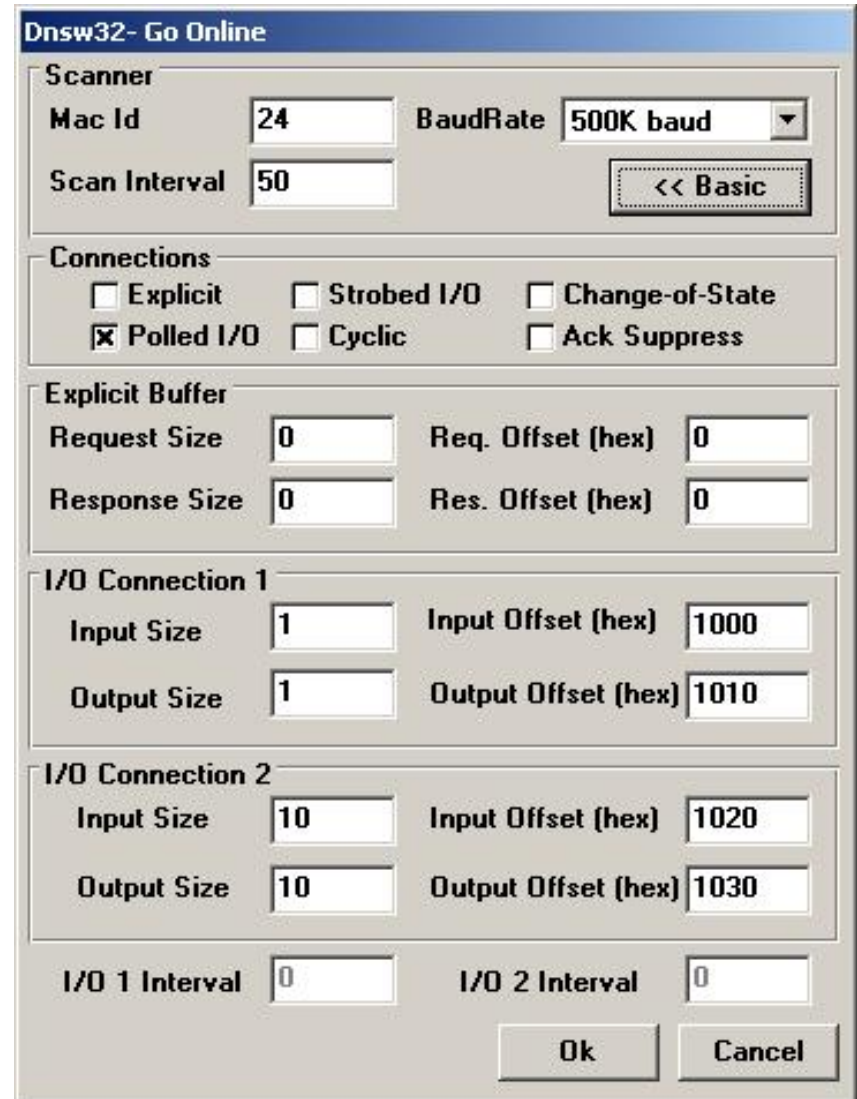


**Dns32- Go Online**

Scanner

Mac Id  BaudRate

Scan Interval



**Dns32- Go Online**

Scanner

Mac Id  BaudRate

Scan Interval

Connections

Explicit  Strobed I/O  Change-of-State

Polled I/O  Cyclic  Ack Suppress

Explicit Buffer

Request Size  Req. Offset (hex)

Response Size  Res. Offset (hex)

I/O Connection 1

Input Size  Input Offset (hex)

Output Size  Output Offset (hex)

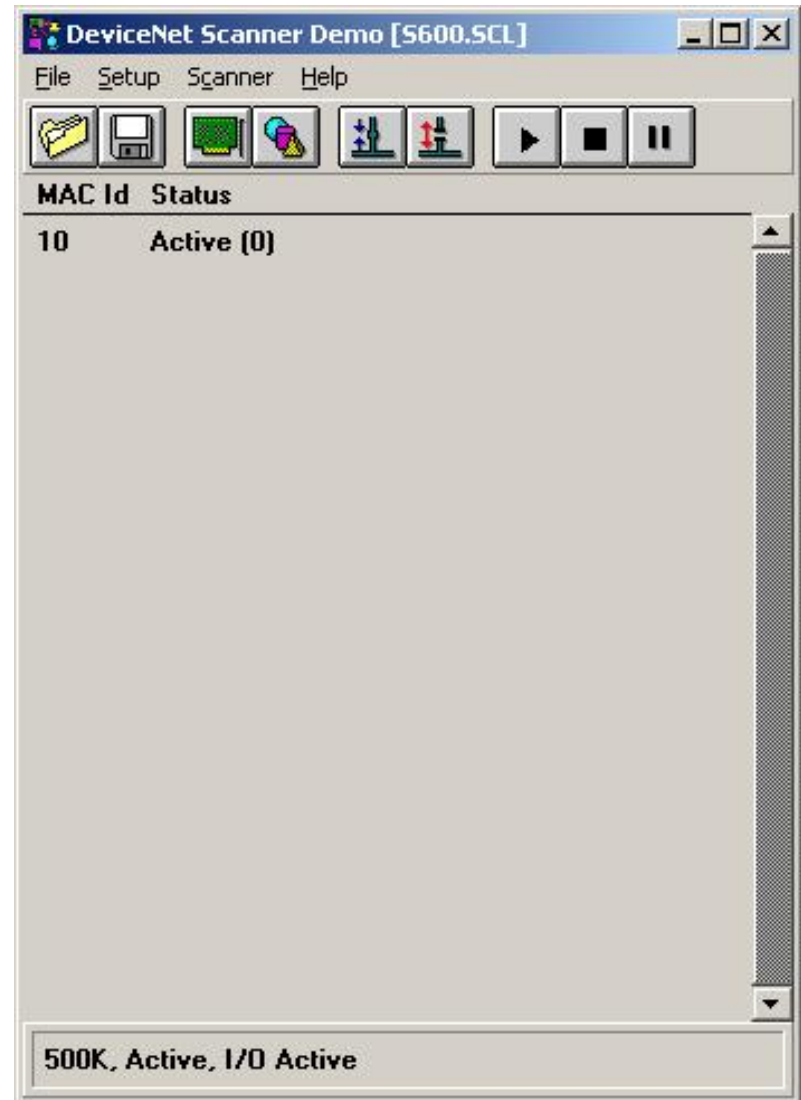
I/O Connection 2

Input Size  Input Offset (hex)

Output Size  Output Offset (hex)

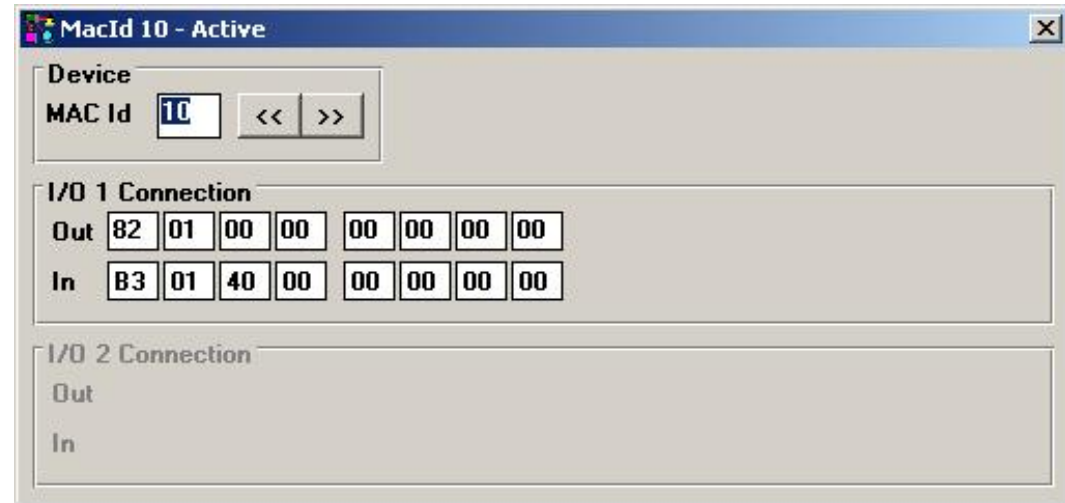
I/O 1 Interval  I/O 2 Interval

## ■ Node Active



## ■ Polled I/O

- Node address (MAC Id)
- Output to drive
  - Byte 0 – 7
  - Bits 0 - 7
- Reply from drive
  - Byte 0 – 7
  - Bits 0 - 7



## ■ Explicit Message Screen

- Set GV to 5000 dec (1388 hex)

- Node address (MAC Id)

- Service Code

- Class ID

- Data Size

- Instance ID

- Attribute

- Explicit Message  
(LSD to MSD)

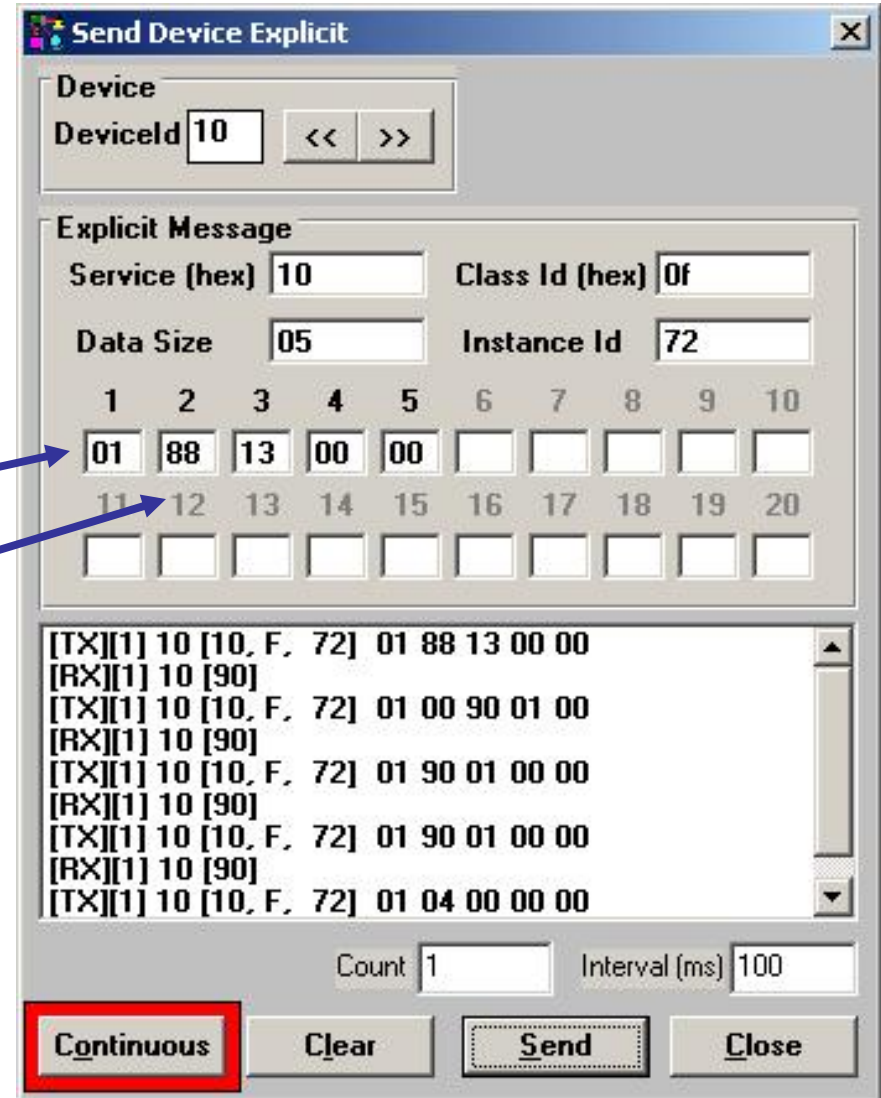
13 88 = 5000 dec (weight 1000)

- Transmit = 1

- Address = 10

- [Service, Class, Instance]

- Attribute, Explicit Message



The dialog box 'Send Device Explicit' contains the following fields and controls:

- Device**: DeviceId  with left and right arrow buttons.
- Explicit Message**:
  - Service (hex)  Class Id (hex)
  - Data Size  Instance Id
  - A 2x10 grid of hex input fields. The first row contains 01, 88, 13, 00, 00, and the second row contains 11, 12, 13, 14, 15, 16, 17, 18, 19, 20.
- A list box showing a sequence of TX and RX messages with their corresponding data bytes.
- Count  Interval (ms)
- Buttons: **Continuous** (highlighted with a red box), **Clear**, **Send**, and **Close**.