

---

# **Handbuch der Windows DLL für den Messverstärker GSV-4**

Letzte Änderungen: 16.12.2013

## Inhaltsverzeichnis

Handbuch der Windows DLL für den Messverstärker GSV-4.....	1
1 Beschreibung.....	3
2. Versionen und Varianten.....	3
1.1. MEGSV4.DLL.....	3
1.2 MEGSV4w32.DLL.....	3
1.3 MEGSV4w64.DLL.....	3
3. Prinzipielle Vorgehensweise.....	3
3.1. Einbinden der DLL.....	4
3.2. Programmablauf.....	4
3.2.1. Initialisierung.....	4
3.2.2. Hauptschleife bzw. -thread.....	4
3.2.3. Beendigung.....	5
4. Die DLL Funktionen.....	5
4.1. GSV4actExt.....	5
4.2. GSV4activateExtended.....	6
4.3. GSV4release.....	7
4.4. GSV4dllVersion.....	7
4.5. GSV4getTXmode.....	8
4.6. GSV4setTXmode.....	8
4.7. GSV4startTX.....	9
4.8. GSV4stopTX.....	9
4.9. GSV4firmwareVersion.....	10
4.10. GSV4getMode.....	10
4.11. GSV4setMode.....	11
4.12. GSV4getLastError.....	11
4.13. GSV4readAll.....	12
4.14. GSV4readMultiple.....	13
4.15. GSV4read.....	15
4.16. GSV4received.....	16
4.17. GSV4getValue.....	16
4.18. GSV4clearDLLbuffer.....	17
4.19. GSV4setZero.....	17
4.20. GSV4readFreq.....	18
4.21. GSV4writeFreq.....	18
4.22 GSV4readSerialNo.....	19
4.23 GSV4DispGetNorm.....	20
4.24 GSV4DispSetNorm.....	21
4.25 GSV4getGain.....	21
4.26 GSV4setGain.....	22
4.27 GSV4getAnalogueType.....	23
4.28 GSV4setAnalogueType.....	24
4.29 GSV4saveAll.....	24
4.30 GSV4getAll.....	25
4.31 GSV4readThreshold.....	25
4.32 GSV4writeThreshold.....	26
4.33 GSV4getDigitalIOType.....	27
4.34 GSV4setDigitalIOType.....	29
4.35 GSV4getDigitalIOValue.....	30

4.36 GSV4setDigitalOvalue.....	31
4.37 GSV4getUnit.....	31
4.38 GSV4setUnit.....	33
4.39 GSV4getUnitText.....	34
4.40 GSV4setUnitText.....	34
4.41 GSV4getCanID.....	35
4.42 GSV4setCanID.....	36
Anhang A Fehlercodes.....	37

## 1 Beschreibung

Zur Programmierung der GSV-4-Baugruppe unter Windows (Warenzeichen der Microsoft Corp.) stehen Dynamic Link Librarys (DLL) zur Verfügung, jeweils eine für 32-bit und eine für 64-Bit. Beide enthalten dieselben Funktionen, die oft den Firmware-Befehlen entsprechen, zuweilen jedoch zusätzliche Funktionalität beinhalten oder mehrere Befehle kombinieren, um die Programmierung zu vereinfachen. Sie enthalten auch Funktionen zum Öffnen einer seriellen Schnittstelle mit gleichzeitigem Aufbau der Kommunikation mit dem GSV-4 und zum Schließen einer so geöffneten seriellen Schnittstelle. Die Funktionen der DLL können über ihren Index (Ordnungszahl) importiert werden.

Die aktivierten Schnittstellen und GSVs werden für jedes Programm, das diese DLL verwendet, separat verwaltet. Die Benutzung einer Schnittstelle (mit GSV), die von einem anderen Prozess aktiviert wurde, ist nicht möglich. Verschiedene Threads eines Prozesses können ihre aktivierten Schnittstellen gemeinsam parallel bzw. pseudo-parallel verwenden.

## 2. Versionen und Varianten

Die hier aufgeführten Varianten sind in Aufrufliste, Rückgabetyt und Funktionalität identisch. Für alle Varianten stehen Wrapper-VIs für die leichtere Einbindung in NI LabView zur Verfügung, sowie eine C-lib zur Verwendung mit C/C++.

### 1.1. MEGSV4.DLL

Die zuletzt ausgelieferte Version dieser 32-Bit DLL ist 1.10. Sie wird ab November 2013 nicht weiter gepflegt. Die Aufrufkonvention ist cdecl.

### 1.2 MEGSV4w32.DLL

Diese 32-Bit DLL wird ab November 2013 in der Version 1.13.00.32 ausgeliefert. Die Aufrufkonvention ist \_\_stdcall. Daher ist diese Variante zur Verwendung mit VisualBasic und vielen anderen Programmiersprachen geeignet. Eine BAS-Datei zur Einbindung in VB steht zur Verfügung. Ein WinSxS Manifest ist eingebettet.

### 1.3 MEGSV4w64.DLL

Diese 64-Bit DLL wird ab Dezember 2013 in der Version 1.13.00.64 ausgeliefert. Abgesehen davon gilt dasselbe wie für die MEGSV4w32.DLL.

### 3. Prinzipielle Vorgehensweise

Zum Erstellen eines Anwendungsprogramms, das die DLL verwendet, wird in der Regel folgendermaßen vorgegangen:

#### 3.1. Einbinden der DLL

Die DLL und ihre Deklarationsdateien (Header) müssen in das Anwendungsprogramm, das die DLL-Funktionen verwendet, eingebunden werden. Wie dies genau ausgeführt wird, hängt von der Programmiersprache und der Entwicklungsumgebung ab. Im allgemeinen werden DLLs zur Programmlaufzeit, d.h. dynamisch, eingebunden. Für statisches Einbinden steht zur Verwendung mit C/C++ Programmen eine LIB-Datei zur Verfügung.

#### 3.2. Programmablauf

Die meisten Anwendungsprogramme, die Messwerte auslesen und ggf. auswerten, können grob in drei Teile unterteilt werden: Initialisierung, Hauptschleife und Beendigung. Einfache Beispielprogramme in C und LabView stehen zum download auf [www.me-systeme.de](http://www.me-systeme.de) zur Verfügung oder sind in dem Archiv, das dieses Handbuch enthält, vorhanden.

##### 3.2.1. Initialisierung

Die Initialisierung wird bei einem Programmdurchlauf i.d.R. entweder nur einmalig ausgeführt, oder durch eine Benutzereingabe. Sie kann in sich in zwei oder drei Teile unterteilt werden: Einbinden der DLL (nur nötig bei dynamischem Binden), Aktivierung der Schnittstelle und Prüfen und ggf. Anpassen relevanter Einstellungen des Messverstärkers. Beispiel:

Sei die DLL bereits statisch eingebunden. Dann muss als erstes die Schnittstelle geöffnet werden, und zwar durch Aufruf von **GSV4actExt** oder **GSV4activateExtended**.

Anschließend ist es oft zweckmäßig, den Status der ständigen Messwertübertragung mit **GSV4getTXmode** zu prüfen und ggf. anzupassen. Auch wird die Anzeigenormierung (Lesen mit **GSV4DispGetNorm**) oft dazu verwendet, um die später zu lesenden Messwerte in der physikalisch korrekten Darstellung auszuwerten.

##### Hinweis:

Es wird nicht empfohlen, Betriebsparameter, die der GSV-4 selbst nichtflüchtig in seinem EEPROM speichert, pauschal zu setzen - erst recht nicht innerhalb einer Programmschleife -, da das EEPROM durch Schreibzugriffe abgenutzt wird (spezifiziert auf 1 Mio. Schreibvorgänge).

##### 3.2.2. Hauptschleife bzw. -thread

Wenn der Messverstärker ständig selbsttätig Messwerte sendet, ist es zweckmäßig und effizient, bei sich wiederholendem Messwertelesen- und auswerten folgendermaßen vorzugehen:

- Abfragen, ob sich Messwerte im Puffer befinden. Das kann mithilfe der Rückgabe einer der Messwertlesefunktionen geschehen (**GSV4read**, **GSV4readAll** oder **GSV4readMultiple**) oder aber durch die Funktion **GSV4received**.

- Wenn ja: Auswerten des/der Messwert(e), durch Verwendung des Inhalts des Zeigers, der der vorher aufgerufenen Lesefunktion übergeben wurde oder bei Verwendung von **GSV4received**: Aufruf der Lesefunktion(en). Anschließend wird der rohe Messwert oft mit

der in der Initialisierung gelesenen Anzeigeskalierung multipliziert, um physikalisch korrekt dargestellte Messwerte zu erhalten.

- Wenn nein: Warten bzw. Verzögern des Messwertverarbeitungsthreads, damit das Betriebssystem nicht unnötig ausgelastet wird (abhängig von der Programmierumgebung). Es ist effizient, diese Wartezeit etwas kleiner als die Messdatensendeperiode des GSV-4 zu wählen.

Wenn der Messverstärker nicht selbsttätig Messwerte sendet, muss zunächst (also noch vor der Abfrage des Puffers) die Funktion **GSV4getValue** aufgerufen werden, um das Senden eines Messwertframes durch den GSV-4 auszulösen. Das Auslösen des Messwertframesendens ist auch durch eine digitale I/O-Leitung möglich, siehe **GSV4getDigitalIotype**.

### 3.2.3. Beendigung

Zur Beendigung des Programms bzw. der Kommunikation mit dem GSV-4 muss **GSV4release** aufgerufen werden.

## 4. Die DLL Funktionen

Die Funktionen sind hier in der Reihenfolge ihrer Relevanz aufgeführt.

Wenn die ausführliche Beschreibung eines Parameters weggelassen ist, ist davon auszugehen, dass die vorangehende Beschreibung des gleichnamigen Parameters genauso gilt.

Der Rückgabewert ist oft (auch) ein Fehlerstatuswert. Signalisiert dieser keinen Fehler, muss dies nicht bedeuten, dass die Firmware den Befehl korrekt ausgeführt hat (dies prüfen die DLL-Funktionen nicht), sondern es bedeutet nur, dass innerhalb der Funktion selbst kein Fehler aufgetreten ist. Das gilt insbesondere für Schreibbefehle. Daher wird bei Kommandos, die Betriebsparameter des Gerätes ändern, empfohlen, den Wert nach dem Schreiben mit der entsprechenden Lesefunktion wieder auszulesen, um ihn zu verifizieren.

### 4.1. GSV4actExt

Diese Funktion alloziert notwendige Speicherressourcen und versucht, die angegebene COM-Schnittstelle zu öffnen und anschließend die Kommunikation mit dem GSV-4 aufzunehmen.

**Verwendete GSV-4 Firmware Befehle:** get\_mode (0x27), ggf. set\_mode (0x26), get\_tx\_status (0x29), get\_firmware\_version (0x2B)

**EEPROM-Speicherung:** Nein

**Mindest-Firmwareversionsnummer:** 0x05

**DLL-Ordnungszahl (Index):** 1

Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des zu öffnenden COMports	1 bis 256
<b>Bitrate</b>	Signed Int32	Bitrate der COMport-Kommunikation.	i.d.R. Nur <b>115200</b>
<b>BufSize</b>	Signed Int32	Puffergröße in Messwerten.	1 bis $3 \cdot 10^8$

#### Beschreibung:

**ComNo:** Diese Nummer wird vom System oder von Kommunikationstreibern (z.B. Bluetooth, USB) vergeben. Sie kann im Gerätemanager der Systemsteuerung nachgesehen werden.

**Bitrate** Bei fast allen GSV-4-Geräten beträgt diese 115200 Bits/s. Da sie jedoch (mit Einschränkungen) auch verstellt werden kann, sind folgende abweichende Baudraten möglich: 9600, 19200, 38400, 57600 und 230400.

**BufSize:** Dieser Puffer wird für jeden der 4 Kanäle angelegt und darin werden vom GSV-4 eingehende Messwerte abgelegt, bis diese durch eine der Lesefunktionen abgeholt werden. Geschieht letzteres nicht, so werden ältere Werte überschrieben, so dass diese verlorengehen. Ein Pufferüberlauffehler wird dabei nicht ausgelöst, d.h. alles funktioniert auch dann normal weiter.

#### Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Fehlerstatus	-1, 0

#### Beschreibung:

Wenn die Funktion erfolgreich ausgeführt wurde, wird GSV\_OK =0 zurückgegeben, andernfalls, wenn ein Fehler aufgetreten ist, GSV\_ERROR =-1. Im Fehlerfall kann mit GSV4getLastError ein Fehlercode abgeholt werden, der als Hinweis auf eine mögliche Fehlerursache zu interpretieren ist. Siehe dazu Anhang A. Zusätzlich zu diesen dort aufgeführten Fehlercodes kann GSV4getLastError nach fehlgeschlagenem Öffnungsversuch auch einen der *Windows System Error Codes* zurückgeben.

## 4.2. GSV4activateExtended

Wie GSV4actExt, aber mit einem zusätzlichen Flag-Parameter, mit dem beim Öffnen alternative Einstellungen gewählt werden können.

**Verwendete GSV-4 Firmware Befehle:** get\_mode (0x27), ggf. set\_mode (0x26), get\_tx\_status (0x29), get\_firmware\_version (0x2B), ggf. stop\_transmission (0x23)

**EEPROM-Speicherung:** Nein

**Mindest-Firmwareversionsnummer:** 0x05

**DLL-Ordnungszahl (Index):** 36

#### Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des zu öffnenden COMports	1 bis 256
<b>Bitrate</b>	Signed Int32	Bitrate der COMport-Kommunikation.	i.d.R. nur <b>115200</b>
<b>BufSize</b>	Signed Int32	Puffergröße in Messwerten.	1 bis $3 \cdot 10^8$
<b>flags</b>	Signed Int32	Zusätzliche Einstellungen, s.u.	Flagwert: 0 bis 772

#### Beschreibung:

**ComNo:** Wie GSV4actExt

**Bitrate** Wie GSV4actExt

**BufSize:** Wie GSV4actExt

**flags:** Folgende Werte können einzeln oder durch bitweises ODER kombiniert übergeben werden:

**ACTEX\_FLAG\_HANDSHAKE Wert 0x004.** Hiermit wird Hardware-Handshake aktiviert. Diese Einstellung empfiehlt sich nur mit Bluetooth-Geräten GSV-4BT, die bis Ende 2012 ausgeliefert wurden. Schläft das Öffnen hiermit dennoch fehl, sollte versucht werden, die Funktion ohne Handshake aufzurufen.

**ACTEX\_FLAG\_WAIT\_EXTENDED Wert 0x0100.** Bei Geräten, die ihre Hardware erst durch Öffnen der Schnittstelle einschalten (z.B. GSV4-BT), wird hiermit länger gewartet, bis die Kommunikation mit dem Gerät ausgenommen wird (nämlich ca. 4 Sek. statt ohne ACTEX\_FLAG\_WAIT\_EXTENDED ca. 0,5 s). Bei Aufnahme der Kommunikation wird der erste Lesebefehl bis zu 5 mal ausgeführt, wenn der GSV-4 noch nicht antwortet.

**ACTEX\_FLAG\_STOP\_TX Wert 0x0200** Mit diesem Flag wird die ständige Messdatenübertragung bereits beim Öffnen pauschal gestoppt. Wenn viele Geräte, die einen gemeinsamen Kommunikationskanal verwenden (zB mehrere GSV-4BT) geöffnet werden oder wenn die Datenrate der ständigen Messwertübertragung relativ hoch ist (ab 125 Werte/s), kann das Öffnen der Schnittstellen hierdurch erleichtert werden.

Rückgabeparameter: Wie GSV4actExt

### 4.3. GSV4release

Mit dieser Funktion wird die Schnittstelle geschlossen und alle allozierten Ressourcen werden freigegeben. Der Aufruf ist vor Beendigung einer Anwendung, die mit GSV4actExt oder GSV4activateExtended eine Schnittstelle geöffnet hat, erforderlich.

**Verwendete GSV-4 Firmware Befehle:** Keine

**DLL-Ordnungszahl (Index):** 10

Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256

Beschreibung:

**ComNo** ist die Nummer des Kommunikationsports, der vorher durch GSV4actExt oder GSV4activateExtended geöffnet wurde.

Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Fehlerstatus	-1, 0

Beschreibung:

Wenn die Funktion erfolgreich ausgeführt wurde, wird GSV\_OK =0 zurückgegeben, andernfalls, wenn ein Fehler aufgetreten ist, GSV\_ERROR =-1. Im Fehlerfall kann mit GSV4getLastError ein Fehlercode abgeholt werden, der als Hinweis auf eine mögliche Fehlerursache zu interpretieren ist. Siehe dazu Anhang A.

### 4.4. GSV4dllVersion

Mit dieser Funktion wird die Versionsnummer der DLL-Version abgefragt.

**Verwendete GSV-4 Firmware Befehle:** Keine

**DLL-Ordnungszahl (Index):** 39

Übergabeparameter: Keine



#### Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	DLL-Version	65536..6553600

Beschreibung:

Der Rückgabeparameter ist: VER\_H x 65536 + VER\_L

Die Versionsressource der DLL enthält die folgenden 4 Nummern:

VER\_H . VER\_L . 00 . XX. Die letzte (XX) ist dabei entweder 32 bei der MEGSV4w32.DLL oder 64 bei der MEGSVw64.DLL.

#### **4.5. GSV4getTXmode**

Mit dieser Funktion wird der Status der ständigen Messwertübermittlung durch den GSV-4 ermittelt.

**Verwendete GSV-4 Firmware Befehle:** get\_tx\_status (0x29)

**EEPROM-Speicherung:** Nein

**Mindest-Firmwareversionsnummer:** 0x08

**DLL-Ordnungszahl (Index):** 51

#### Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
ComNo	Signed Int32	Nummer des geöffneten COMports	1 bis 256

#### Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	TX-Status oder Fehler	-1 bis 3

Beschreibung:

Der Rückgabewert ist im Fehlerfall -1 (dann kann mit GSV4getLastError ein Fehlercode abgeholt werden, s. Anhang A) oder er entspricht einem der folgenden Werte:

**0:** Selbstständiges Messdatensenden permanent (d.h. nichtflüchtig gespeichert) und aktuell aus

**1:** Selbstständiges Messdatensenden permanent (d.h. nichtflüchtig gespeichert) an, aber temporär (d.h. flüchtig) abgeschaltet, weil GSV4setTXmode oder GSV4stopTX dementsprechend gegeben wurde.

**2:** Selbstständiges Messdatensenden permanent (d.h. nichtflüchtig gespeichert) aus, aber temporär (d.h. flüchtig) eingeschaltet, weil GSV4setTXmode oder GSV4startTX dementsprechend gegeben wurde.

**3:** Selbstständiges Messdatensenden permanent (d.h. nichtflüchtig gespeichert) und aktuell an.

#### **4.6. GSV4setTXmode**

Mit dieser Funktion kann der Status der ständigen Messwertübermittlung des GSV-4 geändert werden.

**Verwendete GSV-4 Firmware Befehle:** set\_tx\_status (0x28)

**EEPROM-Speicherung:** Ja, wenn nichtflüchtiger Zustand geändert (s.o.)

**Mindest-Firmwareversionsnummer:** 0x08

**DLL-Ordnungszahl (Index):** 50



#### Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256
<b>val</b>	Signed Int32	Zu setzender TX status	Flagwert 0 bis 3

Beschreibung:

**val:** Siehe Beschreibung des Rückgabeparameters der GSV4getTXmode.

#### Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Fehlerstatus	-1, 0

Beschreibung:

Wenn die Funktion erfolgreich ausgeführt wurde, wird GSV\_OK =0 zurückgegeben, andernfalls, wenn ein Fehler aufgetreten ist, GSV\_ERROR =-1. Im Fehlerfall kann mit GSV4getLastError ein Fehlercode abgeholt werden, der als Hinweis auf eine mögliche Fehlerursache zu interpretieren ist. Siehe dazu Anhang A.

### 4.7. GSV4startTX

Mit dieser Funktion wird die selbstständige Messdatenübertragung eingeschaltet, so dass der GSV-4 anschließend mit der eingestellten Datenrate permanent Messwerte sendet.

**Verwendete GSV-4 Firmware Befehle:** start\_transmission (0x24)

**EEPROM-Speicherung:** Nein

**Mindest-Firmwareversionsnummer:** 0x04

**DLL-Ordnungszahl (Index):** 8

#### Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256

#### Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Fehlerstatus	-1, 0

Beschreibung: s.o.

### 4.8. GSV4stopTX

Mit dieser Funktion wird die selbstständige Messdatenübertragung ausgeschaltet, so dass der GSV-4 anschließend nicht selbstständig Messwerte sendet.

**Verwendete GSV-4 Firmware Befehle:** stop\_transmission (0x23)

**EEPROM-Speicherung:** Nein

**Mindest-Firmwareversionsnummer:** 0x04

**DLL-Ordnungszahl (Index):** 9

#### Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256

#### Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Fehlerstatus	-1, 0

Beschreibung: s.o.

### 4.9. GSV4firmwareVersion

Mit dieser Funktion wird die Versionsnummer der GSV-4 Firmware gelesen.

**Verwendete GSV-4 Firmware Befehle:** get\_firmware\_version (0x2B)

**EEPROM-Speicherung:** Nein

**Mindest-Firmwareversionsnummer:** 0x04

**DLL-Ordnungszahl (Index):** 22

#### Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
ComNo	Signed Int32	Nummer des geöffneten COMports	1 bis 256

#### Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Firmwareversion oder Fehler	-1 bis 255

Beschreibung:

Es ist diese Versionsnummer, die als „Mindest-Firmwareversionsnummer“ in den Befehlsbeschreibungen angegeben ist.

### 4.10. GSV4getMode

Mit dieser Funktion wird der Befehlsmodus des GSV-4 gelesen. Die meisten Befehle sind nur verwendbar, wenn der GSV-4 im Befehlsmodus 1 ist. Die Aktivierungsfunktionen GSV4actExt und GSV4activateExtended setzen den Befehlsmodus bereits auf 1, sofern die Schnittstelle geöffnet werden konnte.

Die einzigen Befehle, die auch im Befehlsmodus 0 verwendbar sind, sind folgende:

- GSV4getMode
- GSV4setMode
- GSV4getTXmode
- GSV4getValue
- GSV4firmwareVersion

**Verwendete GSV-4 Firmware Befehle:** get\_mode (0x27)

**EEPROM-Speicherung:** Nein

**Mindest-Firmwareversionsnummer:** 0x04

**DLL-Ordnungszahl (Index):** 21

#### Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
ComNo	Signed Int32	Nummer des geöffneten COMports	1 bis 256

#### Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Befehlsmodus oder Fehler	-1 bis 2

Beschreibung:

Modus =0 bedeutet, dass nur die drei oben aufgeführten Funktionen ausgeführt werden können. Im Modus 1 können alle für den Benutzer zugänglichen Funktionen ausgeführt werden.

#### **4.11. GSV4setMode**

Mit dieser Funktion kann der Befehlsmodus des GSV-4 gesetzt werden. Die meisten Befehle sind nur verwendbar, wenn der GSV-4 im Befehlsmodus 1 ist. Die Aktivierungsfunktionen GSV4actExt und GSV4activateExtended setzen den Befehlsmodus auf 1, sofern die Schnittstelle geöffnet werden konnte. Diese Funktion kann also dazu verwendet werden, den Befehlsmodus auf 0 zu setzen, um den Messverstärker gegen unbefugtes Verwenden zu sperren und um diesen Zustand wieder auf 1 zurückzusetzen.

Die einzigen Befehle, die auch im Befehlsmodus 0 verwendbar sind, sind folgende:

- GSV4getMode
- GSV4setMode
- GSV4getTXmode
- GSV4getValue
- GSV4firmwareVersion

**Verwendete GSV-4 Firmware Befehle:** set\_mode (0x26)

**EEPROM-Speicherung:** Nein

**Mindest-Firmwareversionsnummer:** 0x05

**DLL-Ordnungszahl (Index):** 20

#### Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256
<b>Mode</b>	Signed Int32	Befehlsmodus	0 bis 1
<b>pw</b>	Zeiger auf Byte-String (0x00 terminiert)	Passwort „berlin“	„berlin“

Beschreibung:

**ComNo:** S.o.

**Mode:** S.o.

**pw:** Das Passwort soll vor versehentlichem oder unbefugtem Ändern des Befehlsmodus schützen. Es wird ein Zeiger auf ein mit 0x00 terminiertes Array vom Datentyp Unsigned Int8 (Char) übergeben, das folgenden 7 Werte enthält:

0x62,0x65,0x72,0x6C,0x69,0x6E,0x00

#### Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Fehlerstatus	-1, 0

#### Beschreibung:

Wenn die Funktion erfolgreich ausgeführt wurde, wird GSV\_OK =0 zurückgegeben, andernfalls, wenn ein Fehler aufgetreten ist, GSV\_ERROR =-1. Im Fehlerfall kann mit GSV4getLastError ein Fehlercode abgeholt werden, der als Hinweis auf eine mögliche Fehlerursache zu interpretieren ist. Siehe dazu Anhang A.

### 4.12. GSV4getLastError

Mit dieser Funktion kann ein Fehlercode abgeholt werden, falls ein vorheriger Funktionsaufruf GSV\_ERROR = -1 zurückgegeben hatte.

**Verwendete GSV-4 Firmware Befehle:** Keine

**DLL-Ordnungszahl (Index):** 2

#### Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256

#### Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Fehlercode	0 bis 805310464

Beschreibung: Wenn der Rückgabewert 0 ist, so ist vorher kein Fehler aufgetreten. Siehe ansonsten Anhang 1 für eine Beschreibung der Fehlercodes.

### 4.13. GSV4readAll

Mit dieser Funktion kann jeweils ein Messwert aller 4 Kanäle aus dem DLL-Puffer gelesen werden.

**Verwendete GSV-4 Firmware Befehle:** Keine

**DLL-Ordnungszahl (Index):** 5

#### Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256
<b>out0</b>	Zeiger auf Double	Gelesener Messwert Kanal 1	-1,05 bis 1,05
<b>out1</b>	Zeiger auf Double	Gelesener Messwert Kanal 2	-1,05 bis 1,05
<b>out2</b>	Zeiger auf Double	Gelesener Messwert Kanal 3	-1,05 bis 1,05
<b>out3</b>	Zeiger auf Double	Gelesener Messwert Kanal 4	-1,05 bis 1,05

#### Beschreibung:

**out0** bis **out3** werden als Zeiger auf vorher allozierte Variablen vom Typ Double übergeben. Wenn der Rückgabewert GSV\_TRUE =1 ist, schreibt die Funktion hier die Rohwerte jeweils eines Messwertes aller 4 Kanäle hinein. Wenn sich mehrere Messwerte in den Puffern befanden, wird der älteste noch nicht gelesene und nicht überschriebene Wert gelesen (siehe Beschreibung zu BufSize in **GSV4actExt**).

Der Messwert 1 entspricht der positiven Nenn-Aussteuerung des Messverstärkers.

Welchem physikalischen Messwert dieser entspricht, hängt von der Einstellung des Eingangstyps und vom angeschlossenen Sensor ab.

Beispiel:

Sei der Eingangstyp `AinTypeBridge2` (siehe **GSV4setAnalogueType**), d.h. der Eingangstyp ist Brückeneingang mit 2mV/V Eingangsempfindlichkeit. Dann entspricht der rohe Messwert 1 der Brückenauslenkung 2mV/V.

Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Statuscode	-1 bis 1

Beschreibung:

**GSV\_ERROR** = -1 bedeutet, dass ein Fehler aufgetreten ist. In diesem Fall kann mit `GSV4getLastError` ein Fehlercode abgeholt werden, der als Hinweis auf eine mögliche Fehlerursache zu interpretieren ist. Siehe dazu Anhang A.

**GSV\_OK** = 0 bedeutet, dass die Funktion zwar fehlerfrei ausgeführt wurde, sich aber keine Messwerte in den Puffern befinden. In diesem Fall wird nichts nach `out0`, `out1`, `out2` und `out3` hineingeschrieben.

**GSV\_TRUE** = 1 bedeutet, dass kein Fehler auftrat und die Funktion in `out0`, `out1`, `out2` und `out3` jeweils einen Messwert abgelegt hat.

#### 4.14. GSV4readMultiple

Mit dieser Funktion können mehrere Messwerte eines bestimmten oder aller 4 Kanäle aus dem DLL-Puffer gelesen werden. Mit ihr können auch sämtliche in allen 4 DLL-Puffern befindlichen Messwerte abgeholt werden, so dass diese unmittelbar nach Ausführung leer sind. Ihre Verwendung empfiehlt sich bei relativ hohen Datenraten des GSV-4 oder wenn das Betriebssystem durch andere Aufgaben stark ausgelastet ist. Ein Bufferüberschreiben durch den Lesethread der DLL wird durch effiziente Verwendung dieser Funktion (siehe dazu auch Kap. 3) sehr unwahrscheinlich.

**Verwendete GSV-4 Firmware Befehle:** Keine

**DLL-Ordnungszahl (Index):** 7

Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256
<b>Chan</b>	Signed Int32	Kanalwert	0 bis 4
<b>out</b>	Zeiger auf Double-Array	Gelesene Messwerte	-1,05 bis 1,05
<b>count</b>	Signed Int32	Anzahl zu lesender Messwerte	1 bis BufSize
<b>valsread</b>	Zeiger auf Signed Int32	Anzahl tatsächlich gelesener Messwerte	0 bis count

Beschreibung:

**ComNo:** S.o.

**Chan:** Der Kanalwert kann entweder 0 oder 1..4 sein. Ist er nicht 0, ist es die Kanalnummer des Kanals, dessen Messwerte gelesen werden sollen. Wenn Messwerte gelesen wurden, steht der älteste Wert am Anfang des Arrays (Index 0). Wenn der Kanalwert =0 ist, werden alle Messwerte aller Kanäle gelesen. Wenn Messwerte gelesen

wurden, steht der älteste Wert des Kanals 1 am Anfang des Arrays (Index 0), dann folgt Kanal 2 usw, bis Index 3. An Index 4 steht dann der zweitälteste Wert des Kanals 1, usw. Siehe Beispiel unten.

Wenn die Anwendung mehrere Kanäle auswertet, ist es sehr empfehlenswert, GSV4readMultiple mit Chan = 0 zu verwenden, auch dann, wenn nicht alle 4 Kanäle von Interesse sind. In letzterem Fall sollten die irrelevanten Kanäle beim Auslesen des Arrays ignoriert bzw. übersprungen werden. Andernfalls kann die zeitliche Zuordnung der relevanten Kanäle sehr schwierig werden.

**out** wird als Zeiger auf ein vorher alloziertes eindimensionales Array vom Typ Double übergeben. **Die Größe dieses Arrays sollte dem höchsten jemals verwendeten Wert von count entsprechen.** Die Arraygröße kann von der Funktion nicht geprüft werden. Ist das Array zu klein, kann es passieren, dass in einen „verbotenen“ Speicherbereich geschrieben wird, was unbedingt vermieden werden sollte!

Wenn der Rückgabewert GSV\_TRUE = 1 ist, schreibt die Funktion hier die Rohwerte der Messwerte hinein.

Der Messwert 1 entspricht der positiven Nenn-Aussteuerung des Messverstärkers. Welchem physikalischen Messwert dieser entspricht, hängt von der Einstellung des Eingangstyps und vom angeschlossenen Sensor ab, siehe Beschreibung zu **GSV4readAll**.

**count:** Anzahl maximal zu lesender Messwerte. Sind mehr Messwerte im Puffer, als durch count übergeben wurde, werden genau count Werte gelesen, andernfalls werden alle im Puffer befindlichen Messwerte gelesen. Sollen die Messwerte aller Kanäle gelesen werden (Chan = 0), so muss count durch 4 teilbar sein, denn dieser Parameter gibt die Gesamtanzahl der Messwerte aller Kanäle an. Daher muss auch das allozierte Array mindestens die Größe count haben, siehe Warnung oben. Oft wählt man daher die Größe der DLL-Puffer = count oder count = BufSize x 4, wenn Chan = 0, so dass die Funktion alle in dem(n) Puffer(n) befindlichen Messwerte abholt.

**valsread:** Wird als Zeiger auf eine vorher allozierte Variable vom Typ Signed Int32 übergeben. Hierin wird stets (außer im Fehlerfall) die Anzahl tatsächlich nach **<out>** geschriebener Messwerte abgelegt. Wenn der Rückgabewert GSV\_TRUE = 1 ist, so sollte dieser Wert ausgewertet werden, da er die Anzahl der nach **<out>** geschriebener Messwerte angibt, d.h. die Größe des (unteren) Teils von out, der Messwerte enthält.

#### Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Statuscode	-1 bis 1

Beschreibung:

**GSV\_ERROR** = -1 bedeutet, dass ein Fehler aufgetreten ist. In diesem Fall kann mit GSV4getLastError ein Fehlercode abgeholt werden, der als Hinweis auf eine mögliche Fehlerursache zu interpretieren ist. Siehe dazu Anhang A.

**GSV\_OK** = 0 bedeutet, dass die Funktion zwar fehlerfrei ausgeführt wurde, sich aber keine Messwerte in den Puffern befinden. In diesem Fall wird nichts nach out hineingeschrieben und <valsread> ist ebenfalls 0.

**GSV\_TRUE** = 1 bedeutet, dass kein Fehler auftrat und die Funktion Messwerte nach <out> abgelegt hat.

Beispiel 1:

Man hat in <out> ein Double-Array der Größe 77 alloziert.

Man übergibt an GSV4readMultiple Chan = 2 und count = 77. Der Rückgabewert sei 1 und <valsread> = 3.

Dann wurden folgende 3 Messwerte nach <out> geschrieben:

{ ältester Wert Kanal 2, zweitältester Wert Kanal 2, neuester Wert Kanal 2 }

Beispiel 2:

GSV4actExt wurde mit BufSize = 1024 erfolgreich aufgerufen.

Man hat in <out> ein Double-Array der Größe 4096 alloziert.

Man übergibt an GSV4readMultiple Chan = 0 und count = 4096. Der Rückgabewert sei 1 und <valsread> = 8.

Dann wurden folgende 8 Messwerte nach <out> geschrieben:

{ ältester Wert Kanal 1, ältester Wert Kanal2, ältester Wert Kanal3, ältester Wert Kanal4, neuester Wert Kanal 1, neuester Wert Kanal2, neuester Wert Kanal3, neuester Wert Kanal4 }

#### 4.15. GSV4read

Mit dieser Funktion kann ein Messwert eines bestimmten Kanals aus dem DLL-Puffer gelesen werden. Die Verwendung dieser Funktion wird nur dann empfohlen, wenn nur ein Kanal verwendet wird, da die Puffer der nicht ausgelesenen Kanäle durch den Lesethread der DLL überschrieben werden und deren zeitliche Zuordnung somit verloren geht.

**Verwendete GSV-4 Firmware Befehle:** Keine

**DLL-Ordnungszahl** (Index): 6

Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256
<b>Chan</b>	Signed Int32	Kanal	1 bis 4
<b>out</b>	Zeiger auf Double	Gelesener Messwert	-1,05 bis 1,05

Beschreibung:

**ComNo:** S.o.

**Chan:** Nummer des zu lesenden Kanals (1 bis 4).

**out** wird als Zeiger auf eine vorher allozierte Variable vom Typ Double übergeben.

Wenn der Rückgabewert GSV\_TRUE = 1 ist, schreibt die Funktion hier den Rohwert des zu lesenden Messwertes hinein. Wenn sich mehrere Messwerte im Puffer befinden, wird der älteste noch nicht gelesene und nicht überschriebene Wert gelesen (siehe Beschreibung zu BufSize in GSV4actExt).

Der Messwert 1 entspricht der positiven Nenn-Aussteuerung des Messverstärkers.

Welchem physikalischen Messwert dieser entspricht, hängt von der Einstellung des Eingangstyps und vom angeschlossenen Sensor ab, siehe Beschreibung zu

**GSV4readAll.**

Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Statuscode	-1 bis 1

Beschreibung:

**GSV\_ERROR** = -1 bedeutet, dass ein Fehler aufgetreten ist. In diesem Fall kann mit GSV4getLastError ein Fehlercode abgeholt werden, der als Hinweis auf eine mögliche



Fehlerursache zu interpretieren ist. Siehe dazu Anhang A.

**GSV\_OK** = 0 bedeutet, dass die Funktion zwar fehlerfrei ausgeführt wurde, sich aber keine Messwerte im Puffer befinden. In diesem Fall wird nichts nach out hineingeschrieben.

**GSV\_TRUE** = 1 bedeutet, dass kein Fehler auftrat und die Funktion einen Messwert in <out> abgelegt hat.

#### 4.16. GSV4received

Mit dieser Funktion kann ermittelt werden, wie viele Messwerte sich im Messwertpuffer befinden. Die Füllstände der 4 Messwertpuffer können nur dann unterschiedlich sein, wenn GSV4read oder GSV4readMultiple mit Chan >0 verwendet wird, denn der GSV-4 sendet ganze Messwertframes, die stets die 4 Werte aller 4 Kanäle enthalten.

**Verwendete GSV-4 Firmware Befehle:** Keine

**DLL-Ordnungszahl** (Index): 4

Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256
<b>Chan</b>	Signed Int32	Kanalwert	0 bis 5

Beschreibung:

**ComNo:** S.o.

**Chan:** Wenn dieser Kanalwert =0 ist, so wird der niedrigste Füllstand der 4 Messwertpuffer zurückgegeben. Wenn dieser Wert = 5 ist, so wird der höchste Füllstand der 4 Messwertpuffer zurückgegeben. Andernfalls ist es die Nummer des Kanals (1 bis 4), dessen Füllstand zurückgegeben wird.

Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Füllstand oder Fehler	-1 bis 1

Beschreibung:

**GSV\_ERROR** = -1 bedeutet, dass ein Fehler aufgetreten ist. In diesem Fall kann mit GSV4getLastError ein Fehlercode abgeholt werden, der als Hinweis auf eine mögliche Fehlerursache zu interpretieren ist. Siehe dazu Anhang A.

Andernfalls wird die Anzahl der Messwerte im Puffer zurückgeben.

#### 4.17. GSV4getValue

Mit dieser Funktion wird der Messverstärker dazu gebracht, einen Messwertframe zu senden. Die Messwerte aller 4 Kanäle befinden sich anschließend in den DLL-Puffern, wo sie mit einer Lesefunktion (z.B. **GSV4readAll**) abgeholt werden können. GSV4getValue eignet sich daher insbesondere, wenn der Messverstärker nicht selbstständig Messwerte sendet (s. z.B. **GSV4getTXmode**).

**Verwendete GSV-4 Firmware Befehle:** get\_value (0x3B)

**EEPROM-Speicherung:** Nein

**Mindest-Firmwareversionsnummer:** 0x04

**DLL-Ordnungszahl** (Index): 23

#### Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256

Beschreibung:

**ComNo:** S.o.

#### Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Fehlerstatus	-1, 0

Beschreibung:

Wenn die Funktion erfolgreich ausgeführt wurde, wird GSV\_OK =0 zurückgegeben, andernfalls, wenn ein Fehler aufgetreten ist, GSV\_ERROR =-1. Im Fehlerfall kann mit GSV4getLastError ein Fehlercode abgeholt werden, der als Hinweis auf eine mögliche Fehlerursache zu interpretieren ist. Siehe dazu Anhang A.

### 4.18. GSV4clearDLLbuffer

Mit dieser Funktion kann der Inhalt aller 4 DLL-Puffer gelöscht werden. Diese Funktion wird besonders dann verwendet, wenn längere Zeit keine Werte aus den Puffern gelesen wurden und diejenigen Messwerte, die noch darin sind, als nicht relevant erachtet werden. Dies kann z.B. unmittelbar vor Eintritt in eine Messwertleseschleife erfolgen.

**Verwendete GSV-4 Firmware Befehle:** Keine

**DLL-Ordnungszahl** (Index): 3

#### Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256

Beschreibung:

**ComNo:** S.o.

#### Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Fehlerstatus	-1, 0

Beschreibung: S.o.

### 4.19. GSV4setZero

Mit dieser Funktion wird der Messverstärker dazu gebracht, den gerade verarbeiteten Messwert des übergebenen Kanals (oder aller 4 Kanäle) auf 0 zu setzen. Der so ermittelte Offsetwert wird durch den GSV-4 gespeichert und von ihm allen kommenden Messwerten hinzuaddiert. Auf diese Weise kann eine Nulltarierung durchgeführt werden.

**Verwendete GSV-4 Firmware Befehle:** set\_zero (0x0C)

**EEPROM-Speicherung:** Ja

**Mindest-Firmwareversionsnummer:** 0x04

**DLL-Ordnungszahl** (Index): 11

#### Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256
<b>Chan</b>	Signed Int32	Kanalwert	0 bis 4

Beschreibung:

**ComNo:** S.o.

**Chan:** Wenn der Kanalwert =0 ist, werden alle 4 Kanäle auf Null gesetzt, andernfalls nur der durch diesen Wert angegebene.

#### Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Fehlerstatus	-1, 0

Beschreibung: S.o.

### 4.20. GSV4readFreq

Mit dieser Funktion wird die im Messverstärker eingestellte Datenrate gelesen. Diese ist als Einzelabtastrate zu verstehen, d.h. sie gilt für jeden der 4 Kanäle (ist sie z.B. =5, so werden jeweils 5 Werte jedes Kanals pro Sekunde gesendet, d.h. insgesamt 20 Werte/s). Der GSV-4 kann diese Datenrate - insbesondere bei höheren, ca. ab 25 Werte/s – nicht ganz einhalten, d.h. die tatsächliche Datenrate ist u.U. niedriger als die angegebene. Der GSV-4 ermittelt mit dieser Datenrate kontinuierlich Messwerte, unabhängig vom eingestellten TXmode. Aber nur mit aktivierter ständiger Messdatenübertragung werden mit dieser Nenn-Datenrate auch Messwerte gesendet.

**Verwendete GSV-4 Firmware Befehle:** get\_frequency (0x16)

**EEPROM-Speicherung:** Nein

**Mindest-Firmwareversionsnummer:** 0x04

**DLL-Ordnungszahl (Index):** 15

#### Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256

Beschreibung:

**ComNo:** S.o.

#### Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Double	Datenrate oder Fehler	-1 oder 0,625 bis 1000

**GSV\_ERROR** = -1 bedeutet, dass ein Fehler aufgetreten ist. In diesem Fall kann mit GSV4getLastError ein Fehlercode abgeholt werden, der als Hinweis auf eine mögliche Fehlerursache zu interpretieren ist. Siehe dazu Anhang A.

Andernfalls wird die Datenrate zurückgegeben.

### 4.21. GSV4writeFreq

Mit dieser Funktion kann die im Messverstärker eingestellte Datenrate geändert werden, siehe Beschreibung zu **GSV4readFreq**. Es werden nur bestimmte, diskrete Datenraten im

Messverstärker gesetzt. Werden andere als die unten aufgelisteten an die Funktion übergeben, wird der nächstliegende vorhandene Wert gesetzt.

**Verwendete GSV-4 Firmware Befehle:** set\_frequency (0x12)

**EEPROM-Speicherung:** Ja

**Mindest-Firmwareversionsnummer:** 0x04

**DLL-Ordnungszahl (Index):** 14

Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256
<b>Freq</b>	Double	Einzustellende Datenrate	0,625 bis 250 (500) <sup>1</sup>

Beschreibung:

**ComNo:** S.o.

**Freq:** Folgende Werte werden im Gerät gesetzt (alle anderen werden zu einem dieser Werte genähert):

Datenfrequenz in Hz (nominal)	Datenfrequenz in Hz (effektiv)
0,63	0,625
1,25	1,250
2,5	2,500
3,75	3,750
6,25	6,250
7,5	7,500
12,5	12,400
15	14,7
25	24,4
125	114
250	208
500	297

Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Fehlerstatus	-1, 0

Beschreibung:

Wenn die Funktion erfolgreich ausgeführt wurde, wird GSV\_OK =0 zurückgegeben, andernfalls, wenn ein Fehler aufgetreten ist, GSV\_ERROR =-1. Im Fehlerfall kann mit GSV4getLastError ein Fehlercode abgeholt werden, der als Hinweis auf eine mögliche Fehlerursache zu interpretieren ist. Siehe dazu Anhang A.

<sup>1</sup> Die maximal mögliche Datenrate ist 250 Hz bis Firmware-Version 0x13. Die DLL prüft diesen Maximalwert. Wenn kommende Firmware-Versionen höhere Datenraten stabil verarbeiten können, wird dies kommenden aktualisierten DLL-Versionen bekannt sein und ein höherer Wert kann gesetzt werden.

## 4.22 GSV4readSerialNo

Mit dieser Funktion kann die individuell vergebene Seriennummer des Geräteexemplars gelesen werden.

**Verwendete GSV-4 Firmware Befehle:** get\_serial\_number (0x1F)

**EEPROM-Speicherung:** Nein

**Mindest-Firmwareversionsnummer:** 0x04

**DLL-Ordnungszahl (Index):** 17

Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256
<b>SerNo</b>	Zeiger auf Byte-String	Seriennummer als ASCII-String	„07000000“ bis „99999999“

Beschreibung:

**ComNo:** S.o.

**SerNo:** Zeiger auf ein vorher zu allozierendes Array mit dem Datentyp Unsigned Int8 (Char/Byte), dessen Größe 9 Bytes sein muss. Bei erfolgreicher Ausführung schreibt die Funktion hier einen ASCII-kodierten und mit 0x00 terminierten String hinein, der die Seriennummer angibt.

Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Fehlerstatus	-1, 0

Beschreibung: S.o.

## 4.23 GSV4DispGetNorm

Mit dieser Funktion kann eine vorher geschriebene Anzeigenormierung gelesen werden, und zwar für jeden Kanal individuell. Man verwendet diesen Wert meistens dazu, um aus den mit den o.g. Lesefunktionen abgeholten Rohwerten physikalisch korrekt skalierte Messwerte zu berechnen, indem die Rohwerte mit der Anzeigenormierung multipliziert werden. Wenn die Messwertanzeige mit der vorhandener Auswertsoftware (z.B. GSVmultichannel) konsistent sein soll, wird die Verwendung dieses Parameters empfohlen.

**Verwendete GSV-4 Firmware Befehle:** get\_user\_scale (0xBB)

**EEPROM-Speicherung:** Nein

**Mindest-Firmwareversionsnummer:** 0x05

**DLL-Ordnungszahl (Index):** 31

Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256
<b>Chan</b>	Signed Int32	Kanal	1 bis 4

Beschreibung:

**ComNo:** S.o.

**Chan:** Kanal, für den dort eine Anzeigenormierung gespeichert wurde.

#### Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Double	Anzeigenormierung oder <b>0 = Fehler</b>	1,175 E-38 bis 3.402 E38

Beschreibung:

Der **Wert 0** bedeutet (hier ausnahmsweise), dass ein Fehler aufgetreten ist. In diesem Fall kann mit GSV4getLastError ein Fehlercode abgeholt werden, der als Hinweis auf eine mögliche Fehlerursache zu interpretieren ist. Siehe dazu Anhang A.

Andernfalls ist der Rückgabewert die gespeicherte Anzeigenormierung.

## 4.24 GSV4DispSetNorm

Mit dieser Funktion kann eine Anzeigenormierung geschrieben werden, und zwar für jeden Kanal individuell. Der GSV-4 verwendet diesen Wert nicht selbst, sondern speichert ihn lediglich nichtflüchtig. Für eine mit vorhandener Auswertsoftware (z.B. GSVmultichannel) übereinstimmende Messwertanzeige wird die Verwendung dieses Parameters empfohlen.

**Verwendete GSV-4 Firmware Befehle:** set\_user\_scale (0xBA)

**EEPROM-Speicherung:** Ja

**Mindest-Firmwareversionsnummer:** 0x05

**DLL-Ordnungszahl (Index):** 30

#### Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256
<b>Chan</b>	Signed Int32	Kanal	1 bis 4
<b>Norm</b>	Double	Anzeigenormierung	-3,402E38 bis 3,402E38 <b>außer 0</b>

Beschreibung:

**ComNo:** S.o.

**Chan:** Kanal, für den die Anzeigenormierung gespeichert werden soll.

**Norm:** Übergebene Anzeigenormierung. Der Wert 0 ist nicht erlaubt, sehr wohl aber negative Werte. Mit solchen kann die Richtung der Sensorauslenkung quasi umgepolt werden, ohne dass der Sensoranschluss in Hardware geändert werden muss.

#### Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Fehlerstatus	-1, 0

Beschreibung:

Wenn die Funktion erfolgreich ausgeführt wurde, wird GSV\_OK =0 zurückgegeben, andernfalls, wenn ein Fehler aufgetreten ist, GSV\_ERROR =-1. Im Fehlerfall kann mit GSV4getLastError ein Fehlercode abgeholt werden, der als Hinweis auf eine mögliche Fehlerursache zu interpretieren ist. Siehe dazu Anhang A.

## 4.25 GSV4getGain

Mit dieser Funktion kann die Einstellung des analogen Eingangstyps des GSV-4 ausgelesen werden, und zwar für den übergebenen Kanal. Dieser Eingangstyp sollte mit dem Ausgangssignaltyp des angeschlossenen Sensors übereinstimmen.

**Verwendete GSV-4 Firmware Befehle:** get\_gain (0xB3)

**EEPROM-Speicherung:** Nein

**Mindest-Firmwareversionsnummer:** 0x08 (0x09)

**DLL-Ordnungszahl (Index):** 29

Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256
<b>Chan</b>	Signed Int32	Kanalwert	0 bis 4

Beschreibung:

**ComNo:** S.o.

**Chan:** Wenn Kanal = 0, werden die Eingangstypen aller 4 Kanäle gelesen. Andernfalls (1 bis 4) ist es die Nummer des Kanals, dessen Eingangstyp gelesen werden soll.

Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Eingangstyp oder Fehler	-1 bis 0x07070707

Beschreibung:

Wenn ein Fehler aufgetreten ist, gibt die Funktion GSV\_ERROR = -1 zurück. Dann kann mit GSV4getLastError ein Fehlercode abgeholt werden, der als Hinweis auf eine mögliche Fehlerursache zu interpretieren ist. Siehe dazu Anhang A.

Andernfalls ist der Eingangstyp folgendermaßen kodiert:

Rückgabewert	Eingangstyp = Sensortyp	Anzeigenormierung für Sensor	Einheit <sup>1</sup>
1	Brückensensor 2mV/V	2	mV/V
2	Brückensensor 10mV/V	10	mV/V
3	Spannungseingang 0..5V	5	V
4	Temperaturfühler PT1000	1000	°C
6	Kabel-Temperatursensor Typ K	1000	°C
7	Spannungseingang 0..10V	10	V

Wenn der Übergabeparameter Chan = 0 ist, stehen die o.g. Werte für die 4 Kanäle innerhalb des Rückgabewertes wie folgt:

Bits	Kanal
31 bis 24	1
23 bis 16	2
17 bis 8	3

<sup>1</sup> Die beiden letzten Spalten dieser Tabelle sind in Bezug auf diese Funktion nicht direkt relevant. Sie sind nur angegeben, um dem Benutzer eine Grundparametrierung zu erleichtern, denn der GSV-4 ändert Einheit und Anzeigenormierung niemals selbst.



7 bis 0	4
---------	---

Andernfalls (Chan = 1..4) ist der Wert in Bits 0 bis 7.

## 4.26 GSV4setGain

Mit dieser Funktion kann die Einstellung des analogen Eingangstyps des GSV-4 gesetzt werden, und zwar für den übergebenen Kanal. Dieser Eingangstyp sollte mit dem Ausgangssignaltyp des angeschlossenen Sensors übereinstimmen.

**Verwendete GSV-4 Firmware Befehle:** set\_gain (0xB2)

**EEPROM-Speicherung:** Ja

**Mindest-Firmwareversionsnummer:** 0x08 (0x09)

**DLL-Ordnungszahl (Index):** 28

Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256
<b>Chan</b>	Signed Int32	Kanalwert	0 bis 4
<b>gain</b>	Signed Int32	Eingangstyp	1 bis 7, außer 5

Beschreibung:

**ComNo:** S.o.

**Chan:** Wenn Kanal = 0, werden die Eingangstypen aller 4 Kanäle auf den selben Wert gesetzt. Andernfalls (1 bis 4) ist es die Nummer des Kanals, für den der Eingangstyp gesetzt werden soll.

**gain:** Der Eingangstyp ist genauso kodiert wie der Rückgabetyt der Funktion

**GSV4getGain** (s.o.) beim Lesen eines einzelnen Kanals. Die Typen 6 und 7 sind erst ab Firmwareversion 0x09 verfügbar.

Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Fehlerstatus	-1, 0

Beschreibung:

Wenn die Funktion erfolgreich ausgeführt wurde, wird GSV\_OK =0 zurückgegeben, andernfalls, wenn ein Fehler aufgetreten ist, GSV\_ERROR =-1. Im Fehlerfall kann mit GSV4getLastError ein Fehlercode abgeholt werden, der als Hinweis auf eine mögliche Fehlerursache zu interpretieren ist. Siehe dazu Anhang A.

## 4.27 GSV4getAnalyseType

Mit dieser Funktion kann die Einstellung des analogen Eingangstyps des GSV-4 gelesen werden, ebenso wie mit GSV4getGain. Allerdings wird für jeden Kanal ein Zeiger übergeben, auf dem vorher eine Variable alloziert werden muss, in die die Eingangstypen hineingeschrieben werden. Für die Verwendung mit C/C++ ist hierbei der Aufzählungsdentyp (enum) *AinType* vorgesehen, siehe Header MEGSV4wXX.h.

**Verwendete GSV-4 Firmware Befehle:** get\_gain (0xB3)

**EEPROM-Speicherung:** Nein

**Mindest-Firmwareversionsnummer:** 0x08 (0x09)

**DLL-Ordnungszahl (Index):** 49

#### Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256
<b>typCh1</b>	Zeiger auf Signed Int32	Eingangstyp Kanal 1	1 bis 7, außer 5
<b>typCh2</b>	Zeiger auf Signed Int32	Eingangstyp Kanal 2	1 bis 7, außer 5
<b>typCh3</b>	Zeiger auf Signed Int32	Eingangstyp Kanal 3	1 bis 7, außer 5
<b>typCh4</b>	Zeiger auf Signed Int32	Eingangstyp Kanal 4	1 bis 7, außer 5

Beschreibung:

**ComNo:** S.o.

**typCh1** bis **typCh4:** Zeiger auf Variablen, die bei erfolgreicher Ausführung die Eingangstypen der 4 Kanäle enthalten. Kodierung siehe **GSV4getGain**.

#### Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Fehlerstatus	-1, 0

Beschreibung: S.o.

## 4.28 GSV4setAnalogueType

Diese Funktion ist fast mit GSV4setGain identisch, außer dass für den Übergabeparameter *type* der C/C++ Aufzählungstyp *AinType* verwendet wird (s. MEGSV4wXX.h).

**Verwendete GSV-4 Firmware Befehle:** set\_gain (0xB2)

**EEPROM-Speicherung:** Ja

**Mindest-Firmwareversionsnummer:** 0x08 (0x09)

**DLL-Ordnungszahl** (Index): 48

#### Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256
<b>Chan</b>	Signed Int32	Kanalwert	0 bis 4
<b>type</b>	Signed Int32	Eingangstyp	1 bis 7, außer 5

Alles weitere: siehe GSV4setGain.

## 4.29 GSV4saveAll

Mit dieser Funktion kann ein kompletter Parametrierungsdatensatz alternativ abgespeichert werden. Der GSV-4 speichert alle Einstellungen, sobald diese geändert werden und stellt diese Einstellungen beim Einschalten wieder her. Wird eine abweichende Parametrierung gewünscht, bzw. verschiedene Parameter-Datensätze, so können diese mithilfe dieser Funktion im EEPROM des GSV-4 gespeichert werden. Diese können dann mit GSV4getAll wieder hergestellt werden.

**Verwendete GSV-4 Firmware Befehle:** save\_konfiguration (0x09).

**EEPROM-Speicherung: Ja**

**Mindest-Firmwareversionsnummer: 0x04**

**DLL-Ordnungszahl (Index): 12**

Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256
<b>DataNo</b>	Signed Int32	Datensatz-No.	2 bis 4

Beschreibung:

**ComNo:** S.o.

**DataNo:** Hier wird die Nummer des Datensatzes angegeben, unter der die Betriebsparameter gespeichert werden sollen und unter der sie mit GSV4getAll wieder hergestellt werden können.

Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Fehlerstatus	-1, 0

Beschreibung: S.o.

#### 4.30 GSV4getAll

Mit dieser Funktion kann ein vorher mit GSV4saveAll abgespeicherter Betriebsparameter-Datensatz wieder hergestellt werden. Zusätzlich können mit DataNo =1 Herstellereinstellungen geladen werden. Die geladenen Einstellungen werden auch (wie stets alle nichtflüchtigen Einstellungen) nach dem nächsten Einschalten wiederhergestellt.

**Verwendete GSV-4 Firmware Befehle:** restore\_konfiguration (0x0A).

**EEPROM-Speicherung: Ja**

**Mindest-Firmwareversionsnummer: 0x04**

**DLL-Ordnungszahl (Index): 13**

Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256
<b>DataNo</b>	Signed Int32	Datensatz-No.	1 bis 4

Beschreibung:

**ComNo:** S.o.

**DataNo:** Hier wird die Nummer des Datensatzes angegeben, der geladen werden soll, wobei 1 der Hersteller-Datensatz ist, der Defaultwerte enthält, die nur werksseitig konfiguriert werden. Unter DataNo 2 bis 4 können vorher mit GSV4saveAll abgelegte Datensätze geladen werden.

Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Fehlerstatus	-1, 0

Beschreibung: S.o.

#### 4.31 GSV4readThreshold

Der GSV-4 hat 8 konfigurierbare Schwellwertschalter. Diese werden eingeschaltet, wenn der Messwert des dem Schwellwertschalter assoziierten Eingangskanals die Einschaltsschwelle überschreitet; oder ausgeschaltet, wenn die Ausschaltsschwelle unterschritten wird. Der Schwellwertschalter sollte auf einen der digitalen I/O-Leitungen gemappt sein; dies kann mit der Funktion **GSV4setDigitalIotype** getan werden. Mit **GSV4readThreshold** werden die Ein- und Ausschaltsschwellen des angegebenen Schwellwertschalters ausgelesen.

**Verwendete GSV-4 Firmware Befehle:** get\_threshold (0x21).

**EEPROM-Speicherung:** Nein

**Mindest-Firmwareversionsnummer:** 0x04

**DLL-Ordnungszahl** (Index): 19

Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256
<b>ThresNo</b>	Signed Int32	Schwellwert-Nr.	1 bis 8
<b>ThresOn</b>	Zeiger auf Double	Einschaltsschwelle	-1,05 bis 1,05
<b>ThresOff</b>	Zeiger auf Double	Ausschaltsschwelle	-1,05 bis 1,05

Beschreibung:

**ComNo:** S.o.

**ThresNo:** Die Nummer des Schwellwertes ist den Kanälen wie folgt zugeordnet, wobei jeweils zwei Schwellwertschalter auf denselben Kanal reagieren:

Schwellwert-No.	Kanal
1 und 2	1
3 und 4	2
5 und 6	3
7 und 8	4

**ThresOn:** Zeiger auf eine vorher allozierte Variable vom Typ Double, die nach erfolgreicher Ausführung die Ein-Schaltsschwelle enthält. Das ist derjenige Schwellwert, bei dessen Überschreitung durch den Messwert des entsprechenden Kanals der Schwellwertschalter eingeschaltet wird.

**ThresOff:** Zeiger auf eine vorher allozierte Variable vom Typ Double, die nach erfolgreicher Ausführung die Aus-Schaltsschwelle enthält. Das ist derjenige Schwellwert, bei dessen Unterschreitung durch den Messwert des entsprechenden Kanals der Schwellwertschalter ausgeschaltet wird.

Beide Schaltsschwellen sind entnormiert, d.h. die Roh-Messwerte (siehe **GSV4readAll**) werden direkt mit ihnen verglichen. Ggf. müssen die von der Funktion gelesenen Schwellen-Rohwerte mit der Anzeigenormierung multipliziert werden.

Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Fehlerstatus	-1, 0

Beschreibung: S.o.

#### 4.32 GSV4writeThreshold

Mit dieser Funktion können die Ein- und Ausschaltschwellen des angegebenen Schwellwertschalters gesetzt werden. Für die Schwellwertschalter und die Schwellwerte gilt das in der die Beschreibung von **GSV4readThreshold** gesagte.

**Verwendete GSV-4 Firmware Befehle:** set\_threshold (0x20).

**EEPROM-Speicherung:** Ja

**Mindest-Firmwareversionsnummer:** 0x04

**DLL-Ordnungszahl (Index):** 18

Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256
<b>ThresNo</b>	Signed Int32	Schwellwert-No.	1 bis 8
<b>ThresOn</b>	Double	Einschaltschwelle	-1,05 bis 1,05
<b>ThresOff</b>	Double	Ausschaltschwelle	-1,05 bis 1,05

Beschreibung:

**ComNo:** S.o.

**ThresNo:** Die Nummer des Schwellwertes ist den Kanälen wie folgt zugeordnet, d.h. jeweils zwei Schwellwertschalter reagieren auf denselben Kanal:

Schwellwert-No.	Kanal
1 und 2	1
3 und 4	2
5 und 6	3
7 und 8	4

**ThresOn:** Wert der Ein-Schaltschwelle. Das ist derjenige Schwellwert, bei dessen Überschreitung durch den Messwert des entsprechenden Kanals der Schwellwertschalter eingeschaltet wird.

**ThresOff:** Wert der Aus-Schaltschwelle. Das ist derjenige Schwellwert, bei dessen Unterschreitung durch den Messwert des entsprechenden Kanals der Schwellwertschalter ausgeschaltet wird.

Beide Schaltschwellen sind entnormiert, d.h. die Roh-Messwerte (siehe GSV4readAll) werden direkt mit ihnen verglichen. Ggf. muss der gewünschte in physikalischen Einheiten skalierte Schwellwert vor der Übergabe durch die Anzeigenormierung geteilt werden.

Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Fehlerstatus	-1, 0

Beschreibung: S.o.

#### 4.33 GSV4getDigitalIOType

Der GSV-4 besitzt 12 konfigurierbare digitale I/O-Leitungen. Mit dieser Funktion kann der

Typ der per Parameter übergebenen I/O-Leitungsnummer gelesen werden. Bei Firmwareversionen 0x08 und 0x09 sind es nur 8 I/O-Leitungen.

**Verwendete GSV-4 Firmware Befehle:** get\_digital (0xB7).

**EEPROM-Speicherung:** Nein

**Mindest-Firmwareversionsnummer:** 0x10 (0x08)

**DLL-Ordnungszahl (Index):** 45

Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256
<b>DigIO_No</b>	Signed Int32	Nummer der digitalen I/O-Leitung	1 bis 12 (8)

Beschreibung:

**ComNo:** S.o.

**DigIO\_No:** Nummer der digitalen I/O-Leitung, deren Typ gelesen werden soll.

Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Digital-I/O-Typ oder Fehler	-1 bis 17

Beschreibung:

Wenn ein Fehler aufgetreten ist, gibt die Funktion GSV\_ERROR =-1 zurück. Dann kann mit GSV4getLastError ein Fehlercode abgeholt werden, der als Hinweis auf eine mögliche Fehlerursache zu interpretieren ist. Siehe dazu Anhang A.

Andernfalls wird der Typ der angegebenen digitalen I/O-Leitung wie folgt angegeben:

Typ-No	Funktion	Input/ Output	GSV-4 Regist erwert
<b>0</b>	Allgemeiner Eingang	Input	0x00
<b>1</b>	Allgemeiner Ausgang, manuell schaltbar	Output	0x01
<b>2</b>	Get Value. Bei Hi-Flanke wird ein Messwertframe übertragen	Input	0x02
<b>3</b>	Zero All. Bei Hi-Flanke werden alle 4 analogen Messkanäle auf Null gesetzt.	Input	0x0A
<b>4</b>	Zero Chan 1. Bei Hi-Flanke wird der Messwert des analogen Kanals 1 auf Null gesetzt.	Input	0x0B
<b>5</b>	Zero Chan 2. Bei Hi-Flanke wird der Messwert des analogen Kanals 2 auf Null gesetzt.	Input	0x0C
<b>6</b>	Zero Chan 3. Bei Hi-Flanke wird der Messwert des analogen Kanals 3 auf Null gesetzt.	Input	0x0D
<b>7</b>	Zero Chan 4. Bei Hi-Flanke wird der Messwert des analogen Kanals 4 auf Null gesetzt.	Input	0x0E
<b>8</b>	Schwellwertausgang 1. Die Leitung ist dem Schwellwertschalter 1 zugeordnet.	Output	0x11
<b>9</b>	Schwellwertausgang 2. Die Leitung ist dem Schwellwertschalter 2 zugeordnet.	Output	0x12
<b>10</b>	Schwellwertausgang 3. Die Leitung ist dem Schwellwertschalter 3 zugeordnet.	Output	0x13

Typ-No	Funktion	Input/ Output	GSV-4 Regist erwert
11	Schwellwertausgang 4. Die Leitung ist dem Schwellwertschalter 4 zugeordnet.	Output	0x14
12	Schwellwertausgang 5. Die Leitung ist dem Schwellwertschalter 5 zugeordnet.	Output	0x15
13	Schwellwertausgang 6. Die Leitung ist dem Schwellwertschalter 6 zugeordnet.	Output	0x16
14	Schwellwertausgang 7. Die Leitung ist dem Schwellwertschalter 7 zugeordnet.	Output	0x17
15	Schwellwertausgang 8. Die Leitung ist dem Schwellwertschalter 8 zugeordnet.	Output	0x18
16	TrigSyncSlave. Auf dieser Leitung kann durch ein als Master konfiguriertes weiteres GSV-4 Gerät das synchrone Senden eines Messwertframes getriggert werden.	Input	0x0F
17	TrigSyncMaster. Bei Übertragung eines Messwertes steuert dieser als Master konfigurierte GSV-4 weitere als Slave konfigurierte Geräte zur synchronen Übertragung der Messwertframes.	Output	0x10

#### 4.34 GSV4setDigitalIOtype

Der GSV-4 besitzt 12 konfigurierbare digitale I/O-Leitungen. Mit dieser Funktion kann der Typ der per Parameter übergebenen I/O-Leitung gesetzt werden. Bei Firmwareversionen 0x08 und 0x09 sind es nur 8 I/O-Leitungen.

**Verwendete GSV-4 Firmware Befehle:** set\_digital (0xB6)

**EEPROM-Speicherung:** Ja

**Mindest-Firmwareversionsnummer:** 0x10 (0x08)

**DLL-Ordnungszahl (Index):** 44

Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
ComNo	Signed Int32	Nummer des geöffneten COMports	1 bis 256
DigIO_No	Signed Int32	Nummer der digitalen I/O-Leitung	0 bis 12 (8)
type	Signed Int32 / enum	Zu setzender Typ der digitalen I/O-Leitung	0 bis 17

Beschreibung:

**ComNo:** S.o.

**DigIO\_No:** Ist dieser Parameter =0, so werden alle digitalen I/O-Leitungen auf denselben übergebenen Typ-Wert gesetzt. Andernfalls ist es die Nummer der digitalen I/O-Leitung, deren Typ gesetzt werden soll.

**type:** Diese Typ-No definiert den Typ der zu verändernden digitalen I/O-Leitung. Sie ist wie der Rückgabewert der Funktion GSV4getDigitalIOtype kodiert, siehe Tabelle oben.



#### Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Fehlerstatus	-1, 0

#### Beschreibung:

Wenn die Funktion erfolgreich ausgeführt wurde, wird GSV\_OK =0 zurückgegeben, andernfalls, wenn ein Fehler aufgetreten ist, GSV\_ERROR =-1. Im Fehlerfall kann mit GSV4getLastError ein Fehlercode abgeholt werden, der als Hinweis auf eine mögliche Fehlerursache zu interpretieren ist. Siehe dazu Anhang A.

### 4.35 GSV4getDigitalIOvalue

Mit dieser Funktion kann der Zustand der digitalen I/O-Leitungen gelesen werden.

**Verwendete GSV-4 Firmware Befehle:** get\_digital\_port (0xB9, 0xBF)

**EEPROM-Speicherung:** Nein

**Mindest-Firmwareversionsnummer:** 0x10 (0x08)

**DLL-Ordnungszahl (Index):** 47

#### Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
ComNo	Signed Int32	Nummer des geöffneten COMports	1 bis 256
DigIO_No	Signed Int32	Nummer der digitalen I/O-Leitung	0 bis 12 (8)

#### Beschreibung:

**ComNo:** S.o.

**DigIO\_No:** Ist dieser Parameter =0, so werden alle (verfügbaren) digitalen I/O-Leitungen gelesen. Andernfalls ist es die Nummer der digitalen I/O-Leitung, die gelesen werden soll. Die I/O-Leitungen 8 bis 12 sind erst ab Firmware-Version 0x10 (d16) verfügbar.

#### Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	I/O-Zustand oder Fehler	-1 bis 0x00000FFF

#### Beschreibung:

Wenn ein Fehler aufgetreten ist, gibt die Funktion GSV\_ERROR =-1 zurück. Dann kann mit GSV4getLastError ein Fehlercode abgeholt werden, der als Hinweis auf eine mögliche Fehlerursache zu interpretieren ist. Siehe dazu Anhang A.

Andernfalls ist die Rückgabe =1, wenn eine einzelne Leitung gelesen werden soll und diese High ist und =0, wenn eine einzelne Leitung gelesen werden soll und diese Low ist. Sollen alle Leitungen gelesen werden, so sind die Zustände in den Bits <11:0> der Rückgabe wie folgt zugeordnet:

Bit	Dig.I/O No.
11	12
10	11
9	10
8	9
7	8
6	7

Bit	Dig.I/O No.
5	6
4	5
3	4
2	3
1	2
0	1

#### 4.36 GSV4setDigitalIOvalue

Mit dieser Funktion kann der Zustand der digitalen I/O-Leitungen gesetzt werden, sofern diese als Ausgang konfiguriert sind (siehe **GSV4getDigitalIOtype**).

**Verwendete GSV-4 Firmware Befehle:** set\_digital\_on\_off (0xB8)

**EEPROM-Speicherung:** Ja

**Mindest-Firmwareversionsnummer:** 0x10 (0x08)

**DLL-Ordnungszahl (Index):** 46

Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256
<b>DigIO_No</b>	Signed Int32	Nummer der digitalen I/O-Leitung	0 bis 12 (8)
<b>value</b>	Signed Int32	Zu setzender Wert	0 bis 1

Beschreibung:

**ComNo:** S.o.

**DigIO\_No:** Ist dieser Parameter =0, so werden alle (verfügbaren) digitalen I/O-Leitungen auf denselben Wert gesetzt. Andernfalls ist es die Nummer der digitalen I/O-Leitung, die gesetzt werden soll. Die I/O-Leitungen 8 bis 12 sind erst ab Firmware-Version 0x10 (d16) verfügbar.

**value:** Wert =0, um die Leitung(en) auf Low zu setzen und =1, um sie auf High zu setzen.

Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Fehlerstatus	-1, 0

Beschreibung:

Wenn die Funktion erfolgreich ausgeführt wurde, wird GSV\_OK =0 zurückgegeben, andernfalls, wenn ein Fehler aufgetreten ist, GSV\_ERROR =-1. Im Fehlerfall kann mit GSV4getLastError ein Fehlercode abgeholt werden, der als Hinweis auf eine mögliche Fehlerursache zu interpretieren ist. Siehe dazu Anhang A.

#### 4.37 GSV4getUnit

Mit dieser Funktion kann für jeden Kanal eine vorher gespeicherte Nummer gelesen werden, die eine Einheit repräsentiert. Diese Einheit wird von bestehenden Anwendungsprogrammen, z.B. von GSVmultichannel, angezeigt.

**Verwendete GSV-4 Firmware Befehle:** get\_unit (0xB5).

**EEPROM-Speicherung:** Nein

**Mindest-Firmwareversionsnummer:** 0x0A

## DLL-Ordnungszahl (Index): 41

### Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256
<b>Chan</b>	Signed Int32	Kanal	1 bis 4

Beschreibung:

**ComNo:** s.o.

**Chan:** Nummer des Kanals, dessen Einheit gelesen werden soll.

### Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Nummer der Einheit oder Fehler	-1 bis 42

Beschreibung:

Wenn ein Fehler aufgetreten ist, gibt die Funktion GSV\_ERROR = -1 zurück. Dann kann mit GSV4getLastError ein Fehlercode abgeholt werden, der als Hinweis auf eine mögliche Fehlerursache zu interpretieren ist. Siehe dazu Anhang A.

Andernfalls wurde die wie folgt kodierte Einheit gelesen:

Wert	Einheit
0	mV/V
1	kg
2	g
3	N
4	cN
5	V
6	µm/m
7	<keine>
8	t
9	kN
10	lb
11	oz
12	kp
13	lbf
14	pdl
15	mm
16	m
17	cNm
18	Nm
19	°C

Wert	Einheit
20	°F
21	K
22	oztr
23	dwt
24	kNm
25	%
26	0/00
27	W
28	kW
29	rpm
30	bar
31	Pa
32	hPa
33	MPa
34	N/mm <sup>2</sup>
35	°
36	Hz
37	m/s
38	km/h
39	m <sup>3</sup> /h
40	mA
41	A
42	m/s <sup>2</sup>

#### 4.38 GSV4setUnit

Mit dieser Funktion kann für jeden Kanal eine vorher gespeicherte Nummer abgelegt werden, die die Einheit repräsentiert. Diese Einheit wird von bestehenden Anwendungsprogrammen, z.B. von GSVmultichannel, angezeigt. Der GSV-4 verwendet diesen Wert nicht selbst, sondern speichert ihn nur nichtflüchtig.

**Verwendete GSV-4 Firmware Befehle:** set\_unit (0xB4).

**EEPROM-Speicherung:** Ja

**Mindest-Firmwareversionsnummer:** 0x0A

**DLL-Ordnungszahl (Index):** 40

Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
ComNo	Signed Int32	Nummer des geöffneten COMports	1 bis 256
Chan	Signed Int32	Kanal	1 bis 4

<b>UnitNo</b>	Signed Int32	Nummer der Einheit	0 bis 42
---------------	--------------	--------------------	----------

Beschreibung:

**ComNo:** s.o.

**Chan:** Nummer des Kanals, dessen Einheit gesetzt werden soll.

**UnitNo:** Nummer der zu setzenden Einheit. Für deren Kodierung, so dass sie mit bestehenden Anwendungsprogrammen identisch ist, siehe Rückgabeparameter von **GSV4getUnit**.

Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Fehlerstatus	-1, 0

Beschreibung:

Wenn die Funktion erfolgreich ausgeführt wurde, wird GSV\_OK =0 zurückgegeben, andernfalls, wenn ein Fehler aufgetreten ist, GSV\_ERROR =-1. Im Fehlerfall kann mit GSV4getLastError ein Fehlercode abgeholt werden, der als Hinweis auf eine mögliche Fehlerursache zu interpretieren ist. Siehe dazu Anhang A.

#### 4.39 GSV4getUnitText

Mit dieser Funktion kann für jeden Kanal eine vorher gespeicherte Einheit gelesen werden. Diese kanalspezifische Einheit wird von bestehenden Anwendungsprogrammen, z.B. von GSVmultichannel, angezeigt. Es kann auch eine vorher vom Benutzer beliebig als Text abgespeicherte Einheit (oder ein beliebiger Text) gelesen werden.

**Verwendete GSV-4 Firmware Befehle:** get\_unit (0xB5) oder get\_user\_string (0xBD)

**EEPROM-Speicherung:** Nein

**Mindest-Firmwareversionsnummer:** 0x0B

**DLL-Ordnungszahl (Index):** 43

Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256
<b>Chan</b>	Signed Int32	Kanalwert	0 bis 4
<b>UnitTxt</b>	Zeiger auf Byte-String	Zeiger auf den zu lesenden Text	Max. Stringlänge: <b>5 Bytes</b>

Beschreibung:

**ComNo:** s.o.

**Chan:** Ist dieser Wert =0, so wird ein beliebiger vorher gespeicherter Text gelesen. Andernfalls ist es die Nummer des Kanals, dessen Einheit gelesen werden soll.

**UnitTxt:** Zeiger auf den String, der gelesen werden soll. Dieser muss ein vorher alloziertes Byte-Array sein, dessen Länge mindestens **6 Bytes** sein muss. Nach erfolgreicher Ausführung wird hier ein 0x00-terminierter Byte-String hineingeschrieben.

Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Fehlerstatus	-1, 0

Beschreibung: s.o.

#### 4.40 GSV4setUnitText

Mit dieser Funktion kann für jeden Kanal eine Einheit gesetzt werden. Diese kanalspezifische Einheit wird von bestehenden Anwendungsprogrammen, z.B. von GSVmultichannel, angezeigt und muss einer der in **GSV4getUnit** tabellierten Einheiten entsprechen. Alternativ dazu kann ein beliebiger Text, der aus Int8-Werten besteht und maximal 5 Zeichen lang sein darf, abgespeichert werden.

**Verwendete GSV-4 Firmware Befehle:** set\_unit (0xB4) oder set\_user\_string (0xBC)

**EEPROM-Speicherung:** Ja

**Mindest-Firmwareversionsnummer:** 0x0B

**DLL-Ordnungszahl (Index):** 42

Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256
<b>Chan</b>	Signed Int32	Kanalwert	0 bis 4
<b>UnitTxt</b>	Zeiger auf Byte-String	Zeiger auf den zu schreibenden Text	Max. Stringlänge: <b>5 Bytes</b>

Beschreibung:

**ComNo:** s.o.

**Chan:** Ist dieser Wert =0, so wird ein beliebiger Text gespeichert. Andernfalls ist es die Nummer des Kanals, dessen Einheit gespeichert werden soll.

**UnitTxt:** Zeiger auf den String, der gespeichert werden soll. Dessen Länge ist maximal **5** Bytes groß und idealerweise ASCII kodiert. Die non-ASCII-Zeichen sollten dem UTF-8 Code entsprechen.

Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Fehlerstatus	-1, 0

Beschreibung: s.o.

#### 4.41 GSV4getCanID

Mit dieser Funktion kann bei GSV-4 CAN-Geräten die CAN-ID für die übergebene Nummer eines der vier verschiedenen Dienste gelesen werden.

**Verwendete GSV-4 Firmware Befehle:** get\_can\_id (0xC6)

**EEPROM-Speicherung:** Nein

**Mindest-Firmwareversionsnummer:** 0x04

**DLL-Ordnungszahl (Index):** 35

Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
<b>ComNo</b>	Signed Int32	Nummer des geöffneten COMports	1 bis 256
<b>IDno</b>	Signed Int32	Nummer des CAN-Dienstes	1, 2, 5, 6

Beschreibung:

**ComNo:** s.o.

**IDno:** Nummer des Dienstes, für den die CAN-ID gelesen werden soll. Es gibt die folgenden 4 verschiedenen Dienste:

CAN-ID:	Beschreibung
TX_1	1 Messwert Senden (610)
TX_2	2 Befehls-Antwort Senden (611)
-	-
-	-
RX_1	5 Get_Value (CANSYNC 110)
RX_2	6 Befehle Empfangen (111)

#### Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	CAN-ID oder Fehler	-1 bis 2047

Beschreibung:

Wenn ein Fehler aufgetreten ist, gibt die Funktion GSV\_ERROR = -1 zurück. Dann kann mit GSV4getLastError ein Fehlercode abgeholt werden, der als Hinweis auf eine mögliche Fehlerursache zu interpretieren ist. Siehe dazu Anhang A.

Andernfalls wurde die CAN-ID gelesen.

#### 4.42 GSV4setCanID

Mit dieser Funktion kann bei GSV-4 CAN-Geräten die CAN-ID für die übergebene Nummer eines der vier verschiedenen Dienste gesetzt werden.

**Verwendete GSV-4 Firmware Befehle:** set\_can\_id (0xC5)

**EEPROM-Speicherung:** Ja

**Mindest-Firmwareversionsnummer:** 0x04

**DLL-Ordnungszahl (Index):** 34

#### Übergabeparameter:

Name	Datentyp	Kurzbeschreibung	Wertebereich
ComNo	Signed Int32	Nummer des geöffneten COMports	1 bis 256
IDno	Signed Int32	Nummer des CAN-Dienstes	1, 2, 5, 6
CanID	Signed Int32	Zu setzende CAN-ID	0 bis 2047

Beschreibung:

**ComNo:** s.o.

**IDno:** Nummer des Dienstes, für den die CAN-ID gesetzt werden soll. Es gibt die folgenden 4 verschiedenen Dienste:

CAN-ID:	Beschreibung
TX_1	1 Messwert Senden (610)
TX_2	2 Befehls-Antwort Senden (611)
-	-
-	-
RX_1	5 Get_Value (CANSYNC 110)
RX_2	6 Befehle Empfangen (111)

#### Rückgabeparameter

Datentyp	Kurzbeschreibung	Wertebereich
Signed Int32	Fehlerstatus	-1, 0

Beschreibung: s.o.





## Anhang A Fehlercodes

Die Fehlercodes, die größer als 0x30.00.00.00 (d805.306.368) sind, sind durch die DLL selbst definiert und werden durch deren Funktionen ausgelöst.

Diejenigen, die im Bereich von 0x20.00.00.00 (d536.870.912) bis 0x2F.FF.FF.FF (d805306367) liegen, sind Windows System Error Codes, die mit 0x20.00.00.00 verodert sind (bitweises ODER).

Die darunter liegenden sind Windows System Error Codes. Diese können z.B. online in der MSDN Dokumentation nachgesehen werden. Sie wurden durch fehlgeschlagene Betriebssystemfunktionen ausgelöst.

Die ersten 6 tabellierten Codes werden u.a. auch von den Aktivierungsfunktionen GSV4actExt und GSV4activateExtended generiert, insbesondere wenn die Verbindung mit dem Messverstärker fehlschlug und treten daher häufiger auf als die restlichen.

No. Hex	No. Dez.	Name	Mögliche Ursache
0x00000002	2	ERROR_FILE_NOT_FOUND	Falsche COMportnummer oder Fehler beim Kommunikationsgerät
0x00000005	5	ERROR_ACCESS_DENIED	1) Der COMport ist bereits durch eine andere Anwendung geöffnet oder immer noch offen durch vorher fehlgeschlagenes Schließen des COMports. Schließen Sie alle Anwendungen, die möglicherweise auf diesen Port zugreifen. Falls dies nicht hilft oder (scheinbar) keine solche Anwendung offen ist, starten Sie das System neu. Oder 2) Der angemeldete Benutzer hat nicht die erforderlichen Rechte zum Zugriff auf diesen Port. Bitte melden Sie sich als Benutzer mit solchen Rechten an, ändern die Rechte und falls Sie nicht wissen wie, kontaktieren Sie ggf. den Systemadministrator.
0x0000001F	31	ERROR_GEN_FAILURE	Hardware Problem mit diesem Port oder mit dem Kommunikationsgerät (z.B. fehlender oder defekter Bluetooth dongle) oder Problem mit dem Treiber des Kommunikationsgerätes oder falsche Konfiguration desselben. Auch möglich: (besonders mit drahtlosen Geräten wie GSV-3BT und GSV-4BT): Batterie leer oder nicht vorhanden. Prüfen Sie die Porteinstellungen in der Systemsteuerung oder – wenn vorhanden – in der GUI des Kommunikationsgerätetreibers.
0x3000001F	805306399	ERR_COM_GEN_FAILURE	Wenn das ACTEX_FLAG_HANDSHAKE Flag in GSV4activateExtended gesetzt war: Problem mit Hardware-Handshaking oder irgendeiner der für Fehler No 31 genannten Gründe (siehe oben).
0x30000058	805306456	ERR_NO_GSV_ANSWER	Bei Aktivierungsfunktionen: Der COMport konnte zwar fehlerfrei geöffnet werden, aber an diesem wurde kein GSV4-Gerät gefunden. Bei Lesefunktionen: Der GSV-4 antwortet nicht innerhalb des Timeouts von ca. 6 Sek. Prüfen Sie die Stromversorgung Ihres GSVs und auch die Kommunikationsleitungen. Möglicherweise machen gewisse Geräteeinstellungen die Kommunikation mit dem GSV schwierig (z.B. Beim GSV-4 mit einer Datenrate von 0,625 /s oder Geräte mit hohen Datenraten, z.B. ab 250 /s). Versuchen Sie in diesem Fall, das Gerät mit einer anderen

No. Hex	No. Dez.	Name	Mögliche Ursache
			Software zu öffnen, (GSVmultichannel oder gsv4test.exe) und verringern Sie hiermit die Datenrate, z.B. auf 12,5 Werte/s.
0x30000101	805306625	ERR_WRONG_COMNO	Die COMportnummer ist kleiner als 1 oder größer als 255. Bitte richtige COMportnummer wählen und nochmal versuchen.
0x300000F6	805306614	ERR_COM_ALREADY_OPEN	Der COMport ist bereits geöffnet. Er kann durch keine zweite Instanz noch einmal geöffnet werden.
0x30000100	805306624	ERR_WRONG_PARAMETER	Falscher Funktions-Übergabeparameter, z.B. Wertebereichsüberschreitung.
0x30000062	805306466	ERR_NOT_SUPPORTED	Ein aufzurufender Firmwarebefehl wird von dem angeschlossenen GSV-4 nicht unterstützt. In der Regel ist dann dessen Firmware zu alt.
0x300000F3	805306611	ERR_MEM_ALLOC	Der Versuch, PC-Speicherbereich zu allozieren, schlug fehl. Eventuell ist der zugesicherte Arbeitsspeicher zu klein.
0x300000F0	805306608	ERR_MUTEXFAILED	Der Versuch, vom OS einen geschützten bzw. gemanagten Zugriff (Mutex) anzumelden, wurde abgewiesen.
0x300000F1	805306609	ERR_EVENTFAILED	Der Versuch, vom OS einen Ereignis-handle (Event) anzumelden, wurde abgewiesen.
0x300000F5	805306613	ERR_BYTES_WRITTEN	Es konnten nicht genug Bytes auf die Schnittstelle geschrieben werden.
0x30000059	805306457	ERR_WRONG_ANSWER_NUM	Der Antwortframe des GSV-4 scheint nicht die definierte Anzahl an Bytes zu enthalten. Grund kann auch eine Fehlsynchronisation sein, oder ein Hardwarefehler.
0x30000060	805306464	ERR_WRONG_ANSWER	Die als Nutzdaten ausgewerteten Bytes im Antwortframe des GSV-4 entsprechen nicht Ihrer Definition. Grund kann auch eine Fehlsynchronisation sein, oder ein Hardwarefehler.
0x30000061	805306465	ERR_WRONG_FRAME_SUFFIX	Der ausgewertete Kommunikationsframe vom GSV-4 hat ein falsches Suffix. Grund kann auch eine Fehlsynchronisation sein, oder ein Hardwarefehler.