

## Fronius Datamanager Modbus TCP & RTU

EN-US | DE

- Benutzerinformation
- Anlagenüberwachung
- User information
- System monitoring





# Inhaltsverzeichnis

Das Modbus Protokoll .....	5
Allgemeines .....	5
Aufbau von Modbus Nachrichten .....	5
Modbus TCP – MBAP Header .....	6
Unterstützte Funktionscodes .....	6
03 (0x03) Read Holding Registers .....	7
06 (0x06) Write Single Register .....	7
16 (0x10) Write Multiple Registers .....	7
Exception Codes .....	8
CRC Berechnung für Modbus RTU .....	9
CRC Prüfsumme berechnen .....	10
CRC Prüfsumme zur Nachricht hinzufügen .....	11
Allgemeines .....	12
Verwendete Abkürzungen .....	12
Kommunikation mit dem Modbus Master .....	12
Register Maps .....	13
Antwortzeiten .....	13
Modbus Geräte-ID für Wechselrichter .....	14
Modbus Geräte-ID für Fronius String Controls .....	14
Modbus Geräte-ID für Energiezähler .....	15
Event Flags .....	15
Registeradressen .....	15
Nicht vorhandene Datensätze .....	17
Zeitverhalten der unterstützten Betriebsarten .....	18
Vorzeichenkonvention für den Power Factor .....	19
Auf der Karte gespeicherte Werte .....	19
Skalierungsfaktoren .....	20
Nicht beschreibbare Register .....	20
Schreiben ungültiger Werte .....	20
Einstellungen - Modbus .....	22
Allgemeines .....	22
Einstellungen - Modbus öffnen .....	22
Datenausgabe über Modbus .....	23
Steuerung einschränken .....	25
Änderungen speichern oder verwerfen .....	26
Fronius Register .....	27
Fronius Register .....	27
Status-Code des Wechselrichters .....	27
Löschen der Event Flags und des Status-Codes .....	27
Daten speichern und löschen .....	28
Datentyp ändern .....	28
Anlagensummen .....	28
Common & Inverter Model .....	29
Common Block Register .....	29
Inverter Model Register .....	29
SunSpec State Codes .....	34
Fronius State Codes .....	35
Nameplate Model (IC120) .....	36
Allgemeines .....	36
Nameplate Register .....	36
Basic Settings Model (IC121) .....	39
Basic Settings Register .....	39
Referenzspannung .....	41
Abweichung zur Referenzspannung .....	41
Extended Measurements & Status Model (IC122) .....	42
Allgemeines .....	42
Extended Measurements & Status Register .....	42
Immediate Controls Model (IC123) .....	45
Allgemeines .....	45
Immediate Controls Register .....	45

Standby .....	47
Leistungsreduktion .....	47
Beispiel: Leistungsreduktion einstellen .....	48
Beispiel: Ändern der Rückkehrzeit bei aktiver Leistungsreduktion .....	48
Auswirkungen der Blindleistungs-Vorgaben auf die Wirkleistung .....	48
Konstanter Power Factor .....	50
Beispiel: Konstanten Power Factor vorgeben .....	50
Konstante relative Blindleistung .....	51
Beispiel: Konstante Blindleistung vorgeben .....	51
Basic Storage Control Model (IC124) .....	52
Allgemeines .....	52
Bereitgestellte Informationen .....	52
Leistungsfenster-Vorgaben .....	52
Vorgabe des minimalen Ladestandes .....	54
Laden des Energiespeichers vom Netz .....	54
Basic Storage Controls Register .....	54
Multiple MPPT Inverter Extension Model (I160) .....	57
Allgemeines .....	57
Multiple MPPT Inverter Extension Register .....	57
String Combiner Model (SC403) .....	60
String Combiner Register .....	60
Meter Model .....	62
Meter Model Register .....	62
End Block .....	69
Allgemeines .....	69
End Block .....	69
String Combiner Event Flags .....	70
String Combiner Event Flags .....	70

# Das Modbus Protokoll

## Allgemeines

Die Beschreibung des Protokolls entstammt zum größten Teil den Modbus Spezifikationen, die öffentlich auf [www.modbus.org/specs.php](http://www.modbus.org/specs.php) erhältlich sind.

Modbus ist ein einfaches, offenes Kommunikationsprotokoll, mit dem eine Master-Slave- oder Client-Server-Kommunikation zwischen den am Netzwerk angeschlossenen Geräten realisiert werden kann. Das Grundprinzip von Modbus ist: Ein Master sendet eine Anfrage und ein Slave antwortet darauf. Bei Modbus TCP wird der Master als Client, ein Slave als Server bezeichnet. Die Funktion ist dieselbe. In weiterer Folge werden für die Beschreibungen der Funktionen des Protokolls unabhängig von den Varianten RTU und TCP nur die gebräuchlicheren Namen Master und Slave verwendet. In Fällen, wo Unterschiede bei zwischen RTU und TCP auftreten, wird speziell darauf hingewiesen.

Am Fronius Datamanager kann Modbus auf 2 Arten benutzt werden:

- Modbus TCP  
Mittels TCP/IP über Ethernet (kabelgebunden oder über WLAN)
- Modbus RTU  
Mittels asynchroner serieller Übertragung über RS-485 (EIA/TIA-485-A), nur bei Fronius Datamanager 2.0

Im Fall von Modbus RTU kann es immer nur einen Master im System geben. Grundsätzlich gilt, dass nur ein Master Anforderungen (Requests) initiieren darf. Ein Slave darf nur antworten (Response), wenn dieser vom Master angesprochen wurde; untereinander dürfen die Slaves nicht kommunizieren. Wird ein Broadcast Request (Anforderung an alle vorhandenen Slaves per Slave ID oder Unit ID 0) ausgesendet, darf keiner der Slaves antworten. Daher können Broadcasts nur für Schreibbefehle verwendet.

Wenn ein Master eine Anforderung an einen Slave sendet, dann erwartet dieser eine Antwort. Bei einer Anforderung eines Masters gibt es eine von fünf Möglichkeiten:

- Erhält der Slave die Anforderung ohne Kommunikationsfehler und kann dieser die Anforderung fehlerfrei bearbeiten, dann wird eine normale Antwort mit den gewünschten Daten zurückgesendet.
- Erhält der Slave die Anforderung wegen eines Kommunikationsfehlers nicht, dann wird keine Antwort gesendet. Das führt zu einem Timeout am Master.
- Erhält der Slave die Anforderung, entdeckt aber einen Kommunikationsfehler (Parity, CRC, ...), wird keine Antwort gesendet. Das führt zu einem Timeout am Master.
- Erhält der Slave die Anforderung ohne Kommunikationsfehler, kann aber diese nicht fehlerfrei bearbeiten (z. B. wenn ein nicht vorhandenes Register ausgelesen werden soll), wird eine Fehlermeldung (Exception Response) mit dem Grund für den Fehler zurückgesendet.
- Erhält der Slave eine Broadcast Anforderung, die auch an alle anderen Geräte geht, so wird weder im Fehlerfall noch wenn die Anforderung erfolgreich bearbeitet wurde, eine Antwort gesendet. Daher sind Broadcast Anforderungen nur für Schreibbefehle geeignet.

Modbus Geräte stellen Daten in 16 Bit großen Datenblöcken (Register) zur Verfügung. In bestimmten Fällen können einzelne Datenpunkte auch mehrere Datenblöcke umfassen (z. B. 2 Register = 32 Bit Wert).

## Aufbau von Modbus Nachrichten

Eine Modbus Nachricht besteht grundsätzlich aus der Protokolldateneinheit (protocol data unit, PDU). Diese ist von darunter liegenden Kommunikationsschichten unabhängig. Abhängig von dem verwendeten Bus oder Netzwerk können noch weitere Felder hinzukommen. Diese Struktur wird dann Anwendungsdateneinheit (application data unit, ADU) genannt.

ADU			
Adressfeld	Funktionscode	Daten	CRC
PDU			

Struktur einer Modbus Nachricht bei Modbus RTU

ADU			
MBAP Header	Funktionscode	Daten	PDU
PDU			

Struktur einer Modbus Nachricht bei Modbus TCP

Für Modbus TCP wird ein eigener Header verwendet, um die Anwendungsdateneinheit zu identifizieren. Dieser Header heißt MBAP Header (MODBUS Application Protocol Header).

Die Größe der Protokolldateneinheit (PDU) ist limitiert, bedingt durch die ersten Modbus Implementierungen in einem seriellen Netzwerk (max. RS485 ADU = 256 Bytes). Dadurch ergibt sich für die Größe der Protokolldateneinheit PDU: PDU = 256 – Slave ID (1 Byte) – CRC (2 Bytes) = 253 Bytes

Damit ergibt sich:

- Modbus RTU ADU = 253 + Slave ID (1 Byte) + CRC (2 Bytes) = 256 Bytes
- Modbus TCP ADU = 253 Bytes + MBAP (7 Bytes) = 260 Bytes

## Modbus TCP – MBAP Header

Der MBAP Header umfasst 7 Bytes:

- **Transaction ID** (2 Bytes): Wird benutzt, um Anfrage und Antwort zu synchronisieren. Der Slave übernimmt die Transaction ID von der Anfrage in die Antwort.
- **Protocol ID** (2 Bytes): Ist immer 0 (Modbus Protokoll).
- **Länge** (2 Bytes): Das Längenfeld enthält die Anzahl der Bytes der nachkommenden Felder, einschließlich Unit ID und Datenfelder.
- **Unit ID** (1 Byte): Dieses Feld wird zur Adressierung der an den Fronius Datamanager angeschlossenen Geräte verwendet (Gateway-Funktion des Fronius Datamanagers). Die Unit ID entspricht der Slave ID bei Modbus RTU. Der Wert wird vom Master vorgegeben und wird vom Slave unverändert mit der Antwort zurückgegeben.

Für Details über die Adressierung der Geräte siehe:

- Modbus Geräte-ID für Wechselrichter auf Seite 14
- Modbus Geräte-ID für Fronius String Controls auf Seite 14
- Modbus Geräte-ID für Energiezähler auf Seite 15

**WICHTIG!** Die richtige Unit ID muss immer angegeben werden, auch wenn der Fronius Datamanager nur mit einem einzelnen Wechselrichter verbunden ist.

## Unterstützte Funktionscodes

Der Funktionscode bestimmt die am Slave auszuführende Aktion. Der Fronius Datamanager unterstützt drei Funktionscodes für Lese- und Schreiboperationen:

- 03 (0x03)<sup>1)</sup> Read Holding Registers
- 06 (0x06)<sup>1)</sup> Write Single Register
- 16 (0x10)<sup>1)</sup> Write Multiple Registers

Tritt am Slave bei der Bearbeitung einer Anforderung ein Fehler auf, so wird eine Fehler-nachricht als Antwort (Exception Response) gesendet. Bei einer solchen Antwort wird beim Funktionscode das höchstwertige Bit auf 1 gesetzt (entspricht einer Addition des Funktionscodes mit 0x80)<sup>1)</sup> und ein Exception Code hinzugefügt, der den Grund des Fehlers angibt.

<sup>1)</sup> Das Prefix "0x" steht für hexadezimale Zahlen

### 03 (0x03) Read Holding Registers

Dieser Funktionscode wird dazu verwendet, den Inhalt eines oder mehrerer aufeinanderfolgenden Register eines Gerätes auszulesen. Die Anforderung enthält die Adresse des ersten auszulesenden Registers und die Anzahl der zu lesenden Register. In der Anforderung werden Register beginnend bei 0 adressiert. Das bedeutet, dass die Register 1 bis 16 über die Adressen 0 bis 15 angesprochen werden.

#### Anforderung

Funktionscode	1 Byte	0x03
Startadresse	2 Bytes	0x0000 bis 0xFFFF (0 bis 65535)
Anzahl der Register	2 Bytes	1 bis 125

#### Antwort

Funktionscode	1 Byte	0x03
Anzahl der Bytes	1 Byte	2 x N*
Registerwerte	N* x 2 Bytes	

\*N = Anzahl der Register

#### Fehler

Fehlercode	1 Byte	0x83
Exception Code	1 Byte	01 oder 02 oder 03 oder 04 oder 11

### 06 (0x06) Write Single Register

Dieser Funktionscode wird dazu verwendet, ein einzelnes Register zu beschreiben. Die Anforderung enthält nur die Adresse des zu beschreibenden Registers. Register werden beginnend bei 0 adressiert. Das bedeutet, dass das Register 1 über die Adresse 0 angesprochen. Die normale Antwort ist eine Kopie der Anforderung, die nach dem erfolgreichen Beschreiben des Registers gesendet wird.

#### Anforderung

Funktionscode	1 Byte	0x06
Registeradresse	2 Bytes	0x0000 bis 0xFFFF (0 bis 65535)
Registerwert	2 Bytes	

#### Antwort

Funktionscode	1 Byte	0x06
Registeradresse	2 Bytes	0x0000 bis 0xFFFF (0 bis 65535)
Registerwert	2 Bytes	

#### Fehler

Fehlercode	1 Byte	0x86
Exception Code	1 Byte	01 oder 02 oder 03 oder 04 oder 11

### 16 (0x10) Write Multiple Registers

Dieser Funktionscode wird dazu verwendet, einen Block von aufeinanderfolgenden Registern zu beschreiben. Die Anforderung enthält die Adresse des ersten zu beschreibenden Registers, die Anzahl der zu beschreibenden Register, die Anzahl der zu schreibenden Bytes und die zu schreibenden Werte (2 Bytes pro Register). Die normale Antwort enthält den Funktionscode, die Startadresse und die Anzahl der beschriebenen Register.

### Anforderung

Funktionscode	1 Byte	0x10
Startadresse	2 Bytes	0x0000 bis 0xFFFF (0 bis 65535)
Anzahl der Register	2 Bytes	1 bis 123
Anzahl der Bytes	1 Byte	2 x N*
Registerwerte	N* x 2 Bytes	

\*N = Anzahl der Register

### Antwort

Funktionscode	1 Byte	0x10
Startadresse	2 Bytes	0x0000 bis 0xFFFF (0 bis 65535)
Anzahl der Register	2 Bytes	1 bis 123

### Fehler

Fehlercode	1 Byte	0x90
Exception Code	1 Byte	01 oder 02 oder 03 oder 04 oder 11

**Exception Codes** Eine Fehlernachricht (Exception Response) besitzt zwei Felder, die sie von einer normalen Antwort unterscheidet:

- **Feld Funktionscode**

In einer normalen Antwort wird der Funktionscode der Anforderung in das Funktionscode Feld der Antwort übernommen. Bei allen Funktionscodes ist das höchstwertige Bit (MSB) 0 (die Werte der Funktionscodes sind alle kleiner als 0x80). In einer Fehlernachricht wird das MSB auf 1 gesetzt. Das bedeutet eine Addition des Wertes für den Funktionscode mit 0x80. Aufgrund des gesetzten MSB kann der Master die Antwort als Fehlernachricht identifizieren.

- **Datenfeld**

Eine normale Antwort enthält Daten oder Statistikwerte im Datenfeld. Bei einer Fehlernachricht wird ein Exception Code im Datenfeld zurückgeliefert. Dieser Exception Code zeigt den Grund für die Fehlernachricht an.

Modbus Exception Codes		
Code	Name	Bedeutung
01	ILLEGAL FUNCTION	Der Funktionscode in der Anforderung wird vom Slave nicht unterstützt.
02	ILLEGAL DATA ADDRESS	Es werden ungültige Registeradressen abgefragt.
03	ILLEGAL DATA VALUE	Ein Wert in der Anforderung ist außerhalb des gültigen Bereichs. Dies gilt sowohl für die Felder einer Anforderung (z. B. ungültige Anzahl an Registern) als auch für ungültige Einstellungswerte der SunSpec Inverter Control Models.
04	SLAVE DEVICE FAILURE	Während des Versuchs, ein oder mehrere Register zu beschreiben, ist ein Fehler aufgetreten.
11	GATEWAY TARGET DEVICE FAILED TO RESPOND	Nur bei Modbus TCP. Das angesprochene Gerät kann nicht gefunden werden: a) das Gerät befindet sich nicht im SolarNet Ring oder b) das Gerät ist ausgeschaltet oder c) der SolarNet Ring ist offen

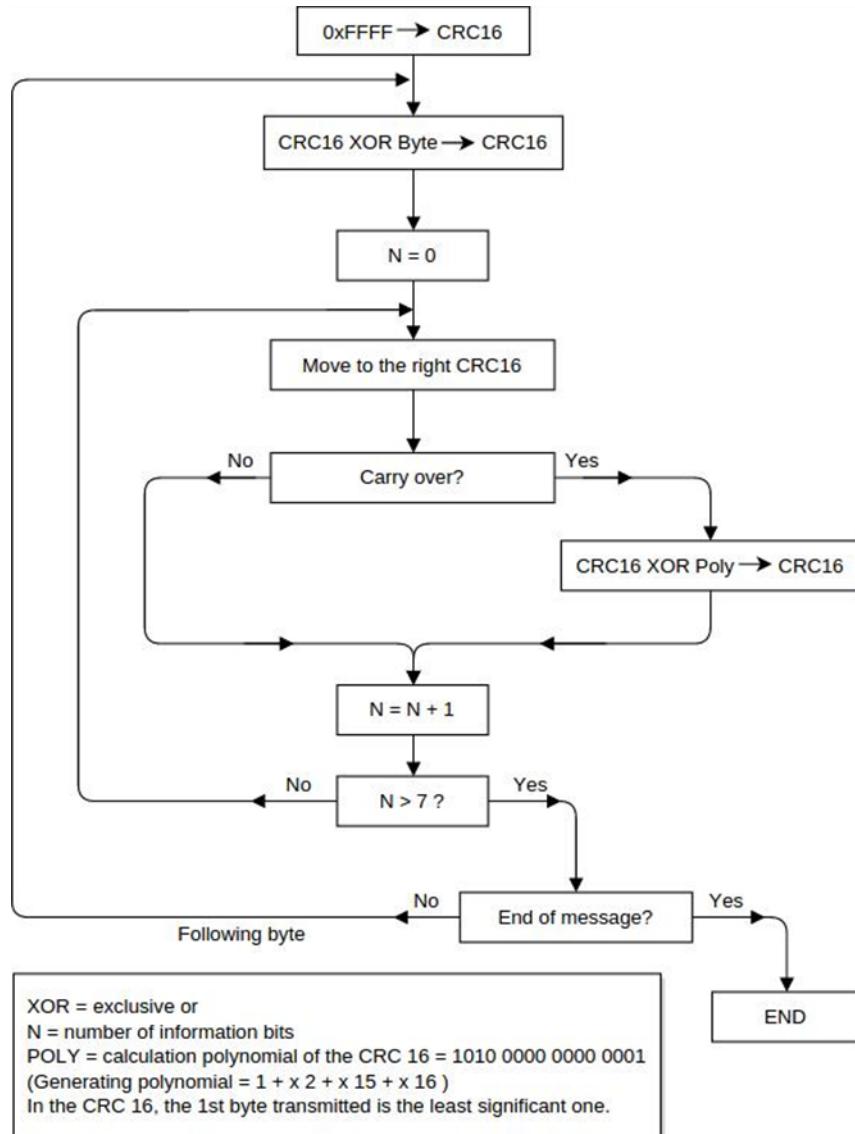
## CRC Berechnung für Modbus RTU

Jede Modbus RTU Nachricht wird mit einer Prüfsumme (CRC, Cyclic Redundancy Check) versehen, um Übertragungsfehler erkennen zu können. Die Prüfsumme ist 2 Bytes groß. Sie wird vom sendenden Gerät berechnet und an die zu sendende Nachricht angehängt. Der Empfänger berechnet seinerseits über alle Bytes der erhaltenen Nachricht (ohne CRC) die Prüfsumme und vergleicht diese mit der empfangenen Prüfsumme. Wenn diese beiden Prüfsummen unterschiedlich sind, ist ein Fehler aufgetreten.

Die Berechnung der Prüfsumme beginnt mit dem Setzen aller Bits eines 16 Bit Registers (CRC Register) auf 1 (0xFFFF). Danach werden alle Bytes der Nachricht einzeln mit dem CRC Register verarbeitet. Nur die Datenbytes einer Nachricht werden zur Berechnung herangezogen. Start-, Stop- und Paritätsbits werden nicht berücksichtigt.

Während der Berechnung der CRC wird jedes Byte mit dem CRC Register XOR-verknüpft. Danach wird das Ergebnis in Richtung des niederwertigsten Bits (LSB) verschoben und das höchstwertige Bit (MSB) auf 0 gesetzt. Das LSB wird betrachtet. Wenn das LSB vorhin 1 war, wird das CRC Register mit einem fix vorgegebenen Wert XOR-verknüpft. War das LSB 0, dann ist nichts zu tun.

Dieser Prozess wird so oft wiederholt, bis das CRC Register 8 Mal verschoben wurde. Nach dem letzten (achten) Schiebevorgang, wird das nächste Byte genommen und mit dem aktuellen CRC Register XOR-verknüpft. Danach beginnt der Schiebeprozess von vorne; wieder wird 8 Mal verschoben. Nach Abhandlung aller Bytes der Nachricht ist der Wert des CRC Registers die Prüfsumme.



Berechnungsalgorithmus der CRC16

### CRC Prüfsumme berechnen

- 1 Initialisierung eines 16 Bit Registers (2 Bytes) mit 0xFFFF. Dieses Register wird als CRC16 Register bezeichnet.
- 2 XOR-Verknüpfung des ersten Bytes der Nachricht mit dem niederwertigen Byte des CRC16 Registers. Das Ergebnis wird im CRC16 Register gespeichert.
- 3 Verschieben des CRC16 Registers um 1 Bit nach rechts (in Richtung LSB), MSB mit 0 auffüllen. LSB betrachten.
- 4 LSB Wert überprüfen
  - War das LSB 0: Gehe zu Schritt 3 (neuerlich verschieben).
  - War das LSB 1: XOR Verknüpfung des CRC16 Registers mit dem CRC Polynom 0xA001 (1010 0000 0000 0001).
- 5 Wiederholung der Schritte 3 und 4 bis 8 Schiebeoperationen durchgeführt worden sind. Wenn diese durchgeführt wurden, wurde ein komplettes Byte der Nachricht bearbeitet.
- 6 Wiederholung der Schritte 3 bis 5 für das nächste Byte der Nachricht. Das ganze wiederholen bis alle Bytes der Nachricht abgearbeitet wurden.
- 7 Nach dem letzten Byte enthält das CRC16 Register die Prüfsumme.

- 8** Wenn die Prüfsumme an die zu sendende Nachricht angehängt wird, dann müssen die beiden Bytes wie unten beschreiben vertauscht werden.

### CRC Prüfsumme zur Nachricht hinzufügen

Wenn die 16 Bit (2 Bytes) CRC Prüfsumme mit einer Nachricht versendet wird, dann wird das niedrigerwertige vor dem höherwertigen Byte übertragen.

Zum Beispiel, wenn die CRC Prüfsumme 0x1241 (0001 0010 0100 0001) ist:

Addr	Func	Data Count	Data	Data	Data	Data	CRC Lo	CRC Hi
							0x41	0x12

# Allgemeines

## Verwendete Abkürzungen

AC	Wechselstrom
DC	Gleichstrom
FW	Firmware
MBC	Fronius Modbus Card
PF	Power Factor ( $\cos \varphi$ )
PV	Photovoltaik
RTC	Echtzeit-Uhr
SF	Skalierungsfaktor
SW	Software
V	Spannung (Volt)
VA	Scheinleistung
VAr	Blindleistung
VMax	Maximale Spannung
VMin	Minimale Spannung
VRef	Referenzspannung
W	Leistung (Watt)
WR	Wechselrichter

## Kommunikation mit dem Modbus Master

Die Kommunikation des Fronius Datamanager mit dem Modbus-Master erfolgt über Registeradressen entsprechend der Spezifikationen der SunSpec Alliance.  
(<http://www.sunspec.org/>)



**HINWEIS!** Der Fronius Datamanager unterstützt auch die Anbindung von Fronius String Controls über Fronius Solar Net.

Fronius String Controls werden durch einen eigenen Common Block und das darauffolgende String Combiner Model dargestellt.

Zusätzlich bietet der Fronius Datamanager die Möglichkeit, die Daten eines über Modbus RTU angeschlossenen Energiezählers via Modbus TCP zur Verfügung zu stellen. Der Zähler wird durch einen eigenen Common Block und das darauf folgende Meter Model dargestellt.

Die Zuordnung der Registeradressen zur entsprechenden Funktion ist folgenden Tabellen zu entnehmen:

- Für alle Geräte:
  - Common Block (C001)
- Für Wechselrichter:
  - Fronius Register
  - Inverter Model (I101, I102, I103, I111, I112 oder I113)
  - Inverter Controls:
    - Nameplate (IC120)
    - Basic Settings (IC121)
    - Extended Measurements & Status (IC122)
    - Immediate Controls (IC123)
    - Multiple MPPT Inverter Extension (I160)
- Für String Controls:
  - String Combiner Model (SC403)
- Für Energiezähler:
  - Meter Model (M201, M202, M203, M211, M212 oder M213)



**HINWEIS!** gilt nur für Modbus RTU und nur wenn kein Energiezähler angeschlossen ist:

Wenn kein Datenaustausch am RS-485 Bus stattfindet, können Rauschen und Störungen die Leitungen beeinflussen. Damit ein Empfänger in einem definierten Zustand bleibt wenn keine Datensignale anliegen, sollten Vorspannungswiderstände verwendet werden, um einen definierten Ruhezustand auf den Datenleitungen zu erhalten.

Der Fronius Datamanager verfügt über keine Vorspannungswiderstände. Detaillierte Informationen über die Verwendung solcher Widerstände finden sich im Dokument „MODBUS over serial line specification and implementation guide V1.02“ ([http://modbus.org/docs/Modbus\\_over\\_serial\\_line\\_V1\\_02.pdf](http://modbus.org/docs/Modbus_over_serial_line_V1_02.pdf)).

## Register Maps

Wechselrichter	String Control	Energiezähler
<b>SID</b> Identifizierung als SunSpec Gerät	<b>SID</b> Identifizierung als SunSpec Gerät	<b>SID</b> Identifizierung als SunSpec Gerät
<b>Common Block</b> Geräteinformationen	<b>Common Block</b> Geräteinformationen	<b>Common Block</b> Geräteinformationen
<b>Inverter Model</b> Wechselrichter-Daten	<b>String Combiner Model</b> String Control Daten	<b>Meter Model</b> Energiezähler-Daten
<b>Nameplate Model</b>	<b>End Block</b>	<b>End Block</b>
<b>Basic Settings Model</b>		
<b>Ext. Measurement Model</b>		
<b>Immediate Controls Model</b>		
<b>Multi. MPPT Inv. Ext. Model</b>		
<b>End Block</b>		

Die Registerlisten können im xlsx-Dateiformat von der Fronius Homepage heruntergeladen werden:

[www.fronius.com - Solar Electronics - Info & Support - Third-party Downloads](http://www.fronius.com - Solar Electronics - Info & Support - Third-party Downloads)

## Antwortzeiten

Die Antwortzeiten hängen unter anderem von der Anzahl der Geräte im Fronius Solar Net Ring ab. Je mehr Geräte verwendet werden, desto größer muss das Timeout für Antworten sein.



**HINWEIS!** Bei mehreren Geräten im Fronius Solar Net Ring sollte für Abfragen von Wechselrichterdaten ein Timeout von mindestens 1 Sekunde verwendet werden.

### Empfehlung für Timeout-Werte

Da bei Fronius String Controls eine einzige Modbus-Abfrage zwei Abfragen über Fronius Solar Net bewirken kann, sind etwas längere Antwortzeiten als bei Wechselrichteranfragen möglich. Wenn Fronius String Controls vorhanden sind, sollte daher ein größerer Timeout-Wert für Antworten verwendet werden.

Bei der ersten Abfrage der Common Block Daten nach einem Neustart des Fronius Data-managers müssen die Informationen über die Fronius String Control einmalig über Fronius Solar Net abgefragt werden. Daher benötigt diese erste Abfrage ein wenig mehr Zeit als die darauffolgenden.

Bei einer größeren Anzahl von Geräten in einem Fronius Solar Net Ring, wird empfohlen diese auf mehrere Fronius Solar Net Ringe mit jeweils einem eigenen Fronius Datamanager aufzuteilen, um noch vertretbare Antwortzeiten zu erhalten.

---

**Modbus Geräte-ID für Wechselrichter**

Die Modbus Geräte-ID des Wechselrichters entspricht seiner Wechselrichter-Nummer, welche über das Bedienpanel des Wechselrichters eingestellt werden kann.  
(siehe Bedienungsanleitung des Wechselrichters)



**HINWEIS!** Hierbei gibt es nur eine einzige Ausnahme:

Die Wechselrichter-Nummer 00 wird auf Geräte-ID 100 umgelegt, da bei Modbus die Geräte-ID 0 für Broadcast Nachrichten reserviert ist.

Beispiel:

Wechselrichter-Nummer	Modbus Geräte-ID
00	100
01	001
02	002
03	003
99	099

---

**Modbus Geräte-ID für Fronius String Controls**

Die Modbus Geräte-ID einer Fronius String Control ergibt sich aus

- ihrer Adresse im Fronius Solar Net
- einem String Control Offset-Wert

Der Standardwert für den String Control Offset ist 101 da für die Wechselrichter der Bereich bis Modbus Geräte-ID 100 reserviert ist.

Der Offset-Wert kann jedoch über die Webseite des Fronius Datamanager verändert werden.

=> siehe Abschnitt "Datenausgabe über Modbus"

**Beispiel 1:** String Control Offset = 101 (Standardwert)

Fronius String Control Adresse	Modbus Geräte-ID
0	101
1	102
2	103
99	200

Ein Fronius Solar Net Ring erlaubt bis zu 100 Wechselrichter und bis zu 200 Fronius String Controls. Die verfügbaren Modbus Geräte-IDs sind ab 240 für andere Funktionen reserviert (z. B. für Energiezähler).

Mit dem Standard String Control Offset von 101 wären also Fronius String Control Adressen ab 139 (entspricht Modbus ID 240) nicht möglich.

Daher kann der String Control Offset über die Webseite des Fronius Datamanager verändert werden, wenn weniger als 100 Wechselrichter zum Einsatz kommen.

**Beispiel 2:** 30 Wechselrichter, 200 Fronius String Controls, String Control Offset = 40

Fronius String Control Adresse	Modbus Geräte-ID
0	40
1	41
2	42
199	239

---

<b>Modbus Geräte-ID für Energiezähler</b>	Ist ein Energiezähler (z. B. Fronius Smart Meter 63A) per Modbus RTU an den Fronius Datamanager angeschlossen, kann dieser per Modbus TCP über die fixe Modbus Geräte-ID 240 ausgelesen werden.
-------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

---

<b>Event Flags</b>	Zustandsänderungen und Fehler der Wechselrichter und Fronius String Controls werden als Event Flags dargestellt.
--------------------	------------------------------------------------------------------------------------------------------------------

Detaillierte Informationen und Listen in verschiedenen Dateiformaten (xlsx, csv, json) können von der Fronius Homepage heruntergeladen werden:

[www.fronius.com](http://www.fronius.com) - Solar Electronics - Info & Support - Third-party Downloads

 **HINWEIS!** Es können auch mehrere State Codes zu einem Ereignis zusammengefasst sein.

#### Für Wechselrichter gilt:

Eine genaue Beschreibung der State Codes ist in der Bedienungsanleitung des betreffenden Wechselrichters zu finden.

Wenn der Wechselrichter einen State Code erzeugt, wird im Fronius Datamanager das entsprechende Event Flag gesetzt.

 **HINWEIS!** Zusätzlich wird der entsprechende State Code auch in Register F\_Active\_State\_Code (214) angezeigt.

Event Flag und State Code bleiben so lange aktiv, wie auch der State Code am Wechselrichter anliegt. Tritt ein weiterer State Code auf, wird dieser ebenfalls in den Event Flags dargestellt. In diesem Fall kann es passieren, dass das vorherige Event Flag nicht gelöscht wird.

Daher ist es möglich, die Event Flags und den State Code manuell zu löschen: durch Schreiben von 0xFFFF in Register F\_Reset\_All\_Event\_Flags (215)

---

<b>Registeradressen</b>	<b>WICHTIG!</b>
	<ul style="list-style-type: none"> <li>- Registeradressen bleiben nicht konstant.</li> <li>- Die tatsächlichen Registeradressen sind abhängig von der Zusammensetzung der dynamischen Sunspec Registerliste.</li> </ul>

Richtige Vorgehensweise:

- das Model per Abfrage suchen (Startadresse ermitteln)
- dann mit Offsets arbeiten

Um ein Register auszulesen muss in der Modbus-Anfrage die Startadresse des Registers angegeben werden.

Fronius Basis Register: 212

SunSpec Basis Register: 40001

Register beginnen bei 1 und stellen keinen Funktionscode dar.

Register nicht mit dem Modicon Adress-Schema verwechseln:

Beim Modicon Adress-Schema wird 40001 als 4x40001 dargestellt.

Um Register 40001 auszulesen, die Adresse 40000 (0x9C40) verwenden.

 **HINWEIS!** Die ausgesendete Registeradresse ist also immer um 1 geringer als die eigentliche Registernummer.

**WICHTIG!** Aufgrund der verwendeten Datentypen können sich die Längen von einzelnen Models verändern. Daher werden bei einigen Registertabellen für SunSpec Models Startadressen angegeben.

Diese Startadresse zusammen mit dem Offset aus der Tabelle ergibt dann den Wert der tatsächlichen Registernummer.

**Beispiel:** Tabelle Nameplate Model (IC120) auf Seite 36:

Das Register *WRtg* des Nameplate Model hat einen Offset von 4. Die Startadresse ist bei der Einstellung „float“ mit 40131 angegeben.

Somit ist die korrekte Registernummer:  $40131 + 4 = 40135$ .

#### Beispiele für Modbus RTU:

##### 1. Abfrage von 4 Registern ab Register 40005 (Mn, Manufacturer)

Senden (Bytes in Hexadezimal)

01	03	9C 44	00 04	2A 4C
Geräte-ID	Function Code	Adresse 40004 (entspricht Register 40005)	Anzahl der auszulesenden Register	Checksumme Low Byte      High Byte

Empfangen (Bytes in Hexadezimal)

01	03	08	46 72	6F 6E	69 75	73 00	8A 2A
Geräte-ID	Function Code	Anzahl der Bytes	Adresse 40005 “F“ und “r“	Adresse 40006 “o“ und “n“	Adresse 40007 “i“ und “u“	Adresse 40008 “s“ und 0	Checksumme Low Byte      High Byte

##### 2. Schreiben von 1 Register ab Register 40242 (WmaxLimPct)

01	10	9D 32	00 01	02	13 88	E3 DD
Geräte-ID	Function Code	Adresse 40242	Anzahl der zu schreibenden Register	Anzahl Datenbytes, die noch folgen	zu schreibender Registerwert $0x1388 = 5000$	Checksumme Low Byte      High Byte

01	10	9D 32	00 01	8F AA
Geräte-ID	Function Code	Adresse 40242	Anzahl der geschriebenen Register	Checksumme Low Byte      High Byte

#### Beispiele für Modbus TCP:

## 1. Abfrage von 4 Registern ab Register 40005 (Mn, Manufacturer)

Senden (Bytes in Hexadezimal)

MBAP Header	01	03	9C 44	00 04
Details siehe Beschreibung MPAB Header	Geräte-ID	Function Code	Adresse 40004 (entspricht Register 40005)	Anzahl der auszulesenden Register

Empfangen (Bytes in Hexadezimal)

MBAP Header	01	03	08	46 72	6F 6E	69 75	73 00
Details siehe Beschreibung MPAB Header	Geräte-ID	Function Code	Anzahl der Bytes	Adresse 40005 "F" und "r"	Adresse 40006 "o" und "n"	Adresse 40007 "i" und "u"	Adresse 40008 "s" und 0

## 2. Schreiben von 1 Register ab Register 40242 (WmaxLimPct)

MBAP Header	01	10	9D 32	00 01	02	13 88
Details siehe Beschreibung MPAB Header	Geräte-ID	Function Code	Adresse 40242	Anzahl der zu schreibenden Register	Anzahl Datenbytes, die noch folgen	zu schreibender Registerwert 0x1388 = 5000

MBAP Header	01	10	9D 32	00 01
Details siehe Beschreibung MPAB Header	Geräte-ID	Function Code	Adresse 40242	Anzahl der geschriebenen Register

### Nicht vorhandene Datensätze

Fronius Wechselrichter können nicht immer alle Daten, die in den SunSpec-Datenmodellen spezifiziert sind, zur Verfügung stellen. Diese Daten werden je nach Datentyp laut SunSpec Spezifikation durch folgende Werte dargestellt:

- int16 (-32767 bis 32767): 0x8000<sup>1)</sup>
- uint16 (0 bis 65534): 0xFFFF
- acc16 (0 bis 65535): 0
- enum16(0 bis 65534): 0xFFFF
- bitfield16 (0 bis 0x7FFF): 0xFFFF
- pad (0x8000): immer 0x8000
- int32 (-2147483647 bis 2147483647) : 0x80000000
- uint32 (0 bis 4294967294): 0xFFFFFFFF
- acc32 (0 bis 4294967295): 0
- enum32(0 bis 4294967294): 0xFFFFFFFF
- bitfield32 (0 bis 0xFFFFFFFF): 0xFFFFFFFF
- int64 (-9223372036854775807 bis 9223372036854775807): 0x8000000000000000
- acc64 (0 bis 18446744073709551615): 0
- stringX: alle X Register mit 0x0000 gefüllt
- float32 (Bereich siehe IEEE 754): 0x7FC00000 (NaN)
- sunssf (Skalierungsfaktoren; -10 bis 10): 0x8000

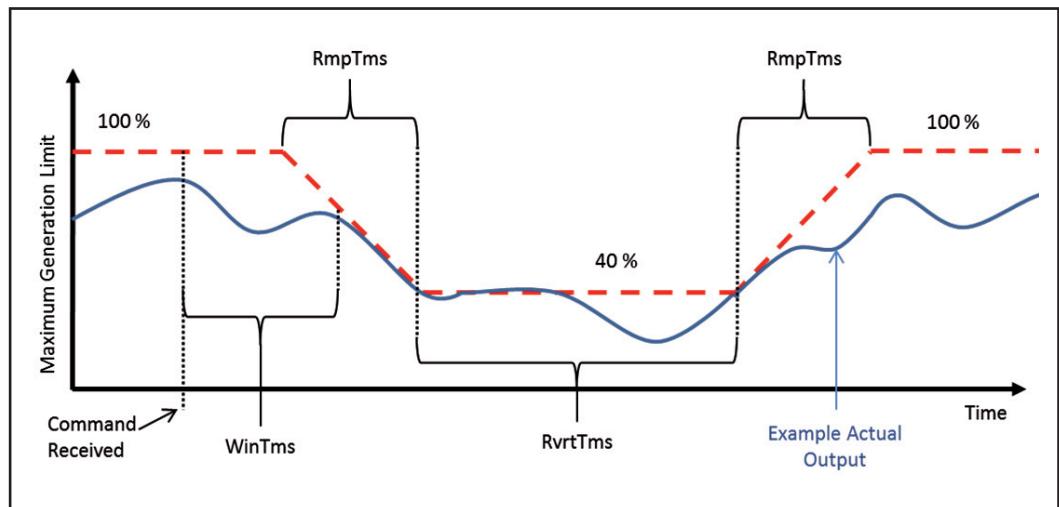
<sup>1)</sup> Das Prefix "0x" steht für hexadezimale Zahlen



**HINWEIS!** Vom Datamanager nicht unterstützte Datenpunkte sind in den Registertabellen in der Spalte „Range of values“ mit „Not supported“ gekennzeichnet. In diesem Fall erhält man beim Auslesen je nach Datentyp den entsprechenden Wert aus der obigen Liste.

In bestimmten Fällen kann es vorkommen, dass grundsätzlich als unterstützt angeführte Register ebenfalls einen solchen Wert zurückliefern. Der Grund dafür ist, dass einige Werte vom Gerätetyp abhängig sind, z.B. die Ströme AphB und AphC bei einem einphasigen Wechselrichter.

## Zeitverhalten der unterstützten Betriebsarten



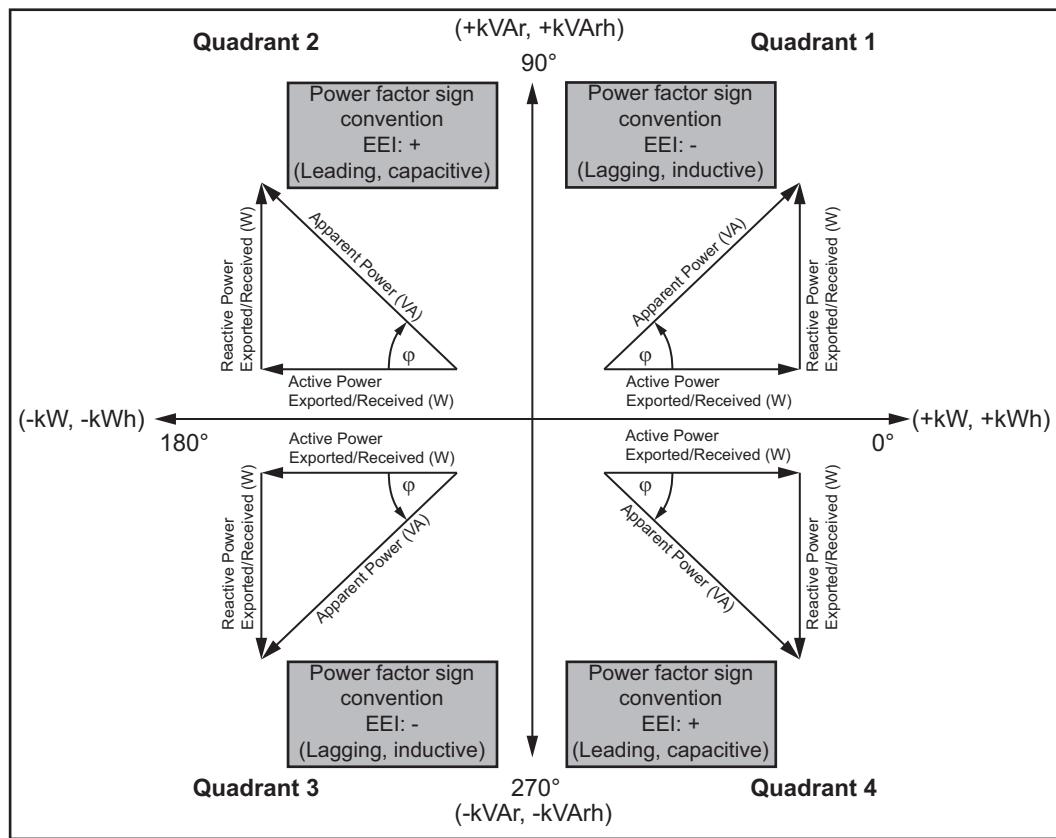
Zeitverhalten am Beispiel einer Leistungsreduktion

Das Zeitverhalten des Wechselrichters in einer Betriebsart kann durch mehrere Zeitwerte festgelegt werden.

In der Abbildung „Zeitverhalten am Beispiel einer Leistungsreduktion“ sind die drei möglichen Zeitwerte dargestellt:

- **WinTms 0 - 300 [Sekunden]**  
gibt ein Zeitfenster an, in dem die Betriebsart zufällig gestartet wird. Das Zeitfenster beginnt mit dem Startbefehl der Betriebsart (z.B. `OutPFSel_Ena = 1`).  
Mit `WinTms` kann verhindert werden, dass alle Wechselrichter in der Anlage die Änderungen gleichzeitig übernehmen. Bei 0 (Standardwert) startet die Betriebsart sofort.
- **RvrtTms 0 - 28800 [Sekunden]**  
bestimmt, wie lange die Betriebsart aktiv sein soll. Nach Ablauf der Zeit wird die Betriebsart automatisch beendet. Ist `RvrtTms` = 0 (Standardwert) bleibt die Betriebsart so lange aktiv, bis er manuell über das entsprechende Register wieder deaktiviert wird.
- **RmpTms** (derzeit nicht vom Datamanager unterstützt)  
gibt vor, wie schnell die Änderungen durchgeführt werden sollen. Der entsprechende Wert wird in der angegebenen Zeit schrittweise vom alten zum neuen Wert hin verändert.  
Ist `RmpTms` = 0 (Standardwert) oder wird dieser Wert gar nicht unterstützt, wird sofort der neue Wert aktuell.

## Vorzeichenkonvention für den Power Factor



Die EEI-Vorzeichenkonvention<sup>1)</sup> für den Power Factor entspricht der SunSpec Spezifikation, und basiert auf den Angaben aus dem "Handbook for Electricity Metering" und der IEC 61557-12 (2007).

Der Power Factor ist:

- negativ bei positiver Blindleistung (übererregt, Quadrant 1)
- positiv bei negativer Blindleistung (untererregt, Quadrant 4)

<sup>1)</sup> EEI = Edison Electrical Institute

## Auf der Karte gespeicherte Werte

Nameplate Model (IC120):

- **WRtg**  
AC Nennleistung des Wechselrichters
- **VArRtg**  
AC Nennscheinleistung des Wechselrichters  
Standardwert = WRtg
- **VArRtgQ1**  
Maximale AC Blindleistung im 1. Quadranten (übererregt).  
Standardwert wird anhand von verfügbarem cos Phi (0.85) und der Nennscheinleistung berechnet. Skalierungsfaktor VArRtg\_SF beachten
- **VArRtgQ4**  
Maximale AC Blindleistung im 4. Quadranten (untererregt).  
Standardwert wird anhand von verfügbarem cos Phi (0.85) und der Nennscheinleistung berechnet. Skalierungsfaktor VArRtg\_SF beachten
- **ARtg**  
AC Nennstrom des Wechselrichters

Basic Settings Model (IC121):

- **WMax**  
Maximale AC Leistung  
Standardwert = WRtg
- **VRef**  
Referenzspannung am Einspeisepunkt
- **VRefOfs**  
Abweichung zur Referenzspannung
- **VMax**  
Maximale AC Spannung
- **VMin**  
Minimale AC Spannung
- **VAMax**  
Maximale AC Scheinleistung  
Standardwert = VARtg

#### **Werte speichern**

Bei nicht vorhandenen oder falsch angezeigten Daten können die oben angeführten Werte angepasst und am Datamanager gespeichert werden.

Änderungen haben derzeit keinen Einfluss auf die Funktionsweise des Datamanagers oder der Wechselrichter und dienen ausschließlich zur Anzeige von gerätespezifischen Informationen.

Um die Werte zu speichern, muss das Register *F\_Store\_Data* (213) eines beliebigen Wechselrichters mit 0xFFFF beschrieben werden. Anschließend sind die Werte für alle Wechselrichter permanent gespeichert und auch nach einem AC Reset des Datamanager verfügbare.

#### **Werte löschen**

Es können nur die Werte für einen einzelnen Wechselrichter gelöscht werden. Dazu ist Register *F\_Delete\_Data* (212) des Wechselrichters mit 0xFFFF zu beschreiben.

---

#### **Skalierungsfaktoren**

**WICHTIG!** Skalierungsfaktoren (auch bei Auswahl von "Float" möglich!) sind nicht statisch, auch wenn diese als Fixwert in dieser BA angeben werden.

Skalierungsfaktoren können sich bei jeder Firmware-Änderung verändern (z.B.: Skalierungsfaktor für Leistungsvorgabe).

Skalierungsfaktoren mit unveränderlichen Werten sind in den Tabellen in der Spalte "Range of values" angeführt.

Aktuelldaten (Daten von Wechselrichtern, String Controls und Energiezählern) können veränderliche Skalierungsfaktoren haben. Diese müssen aus den entsprechenden Registern ausgelesen werden.

---

#### **Nicht beschreibbare Register**

Folgende Register können nicht beschrieben werden:

- Read-Only (R) Register
- aktuell nicht unterstützte Register



**HINWEIS!** Wird versucht solche Register zu beschreiben, gibt der Fronius Datamanager keinen Exception Code zurück!

Die in diese Register geschriebenen Werte werden ohne Fehlermeldung vom Fronius Datamanager ignoriert.

---

#### **Schreiben ungültiger Werte**

Einige Register lassen nur bestimmte Werte zu. Die gültigen Werte sind der jeweiligen Register-Tabelle zu entnehmen.

Wird ein ungültiger Wert in ein Register geschrieben, so gibt der Fronius Datamanager den Exception Code 3 (Illegal Data Value) zurück. Der ungültige Wert wird ignoriert.

Werden mehrere Register auf einmal beschrieben, werden alle gültigen Werte bis zu dem Register mit dem ungültigen Wert geschrieben. Anschließend wird der Schreibvorgang abgebrochen.

# Einstellungen - Modbus

## Allgemeines

Über die Web-Schnittstelle des Fronius Datamanager können via Internet-Browser Einstellungen für die Modbus Anbindung vorgenommen werden, welche über das Modbus-Protokoll nicht ansprechbar sind.



**HINWEIS!** Bei Datenübertragung über Modbus RTU ist die Verwendung der Web-Schnittstelle normalerweise nicht erforderlich, da Modbus RTU werkseitig aktiviert ist.

## Einstellungen - Modbus öffnen

- 1** Fronius Datamanager installieren  
=> siehe Bedienungsanleitung Fronius Datamanager
- 2** Internet-Browser öffnen
- 3** Im Adressfeld des Internet-Browsers eingeben:
  - die IP Adresse des Fronius Datamanager (unter Systeminformationen abrufbar)
  - oder Hostnamen und Domainnamen des Fronius DatamanagerDie Startseite der Web-Schnittstelle wird angezeigt
- 4** Den Bereich "Einstellungen" (1) auswählen
- 5** Den Menüpunkt "Modbus" (2) öffnen

The screenshot shows the Fronius Datamanager web interface. On the left, there is a sidebar with various menu items: PASSWÖRTER, ALLGEMEIN, WECHSELRICHTER, FRONIUS SENSOR CARDS, FRONIUS SOLAR.WEB, SERVICE-MELDUNGEN, NETZWERK, ENERGY MANAGER, MODBUS (2), and EVU EDITOR. The 'MODBUS' item is highlighted. The main content area is titled 'Modbus'. It contains the following settings:

- Datenausgabe über Modbus: A radio button group with 'aus' (selected), 'tcp', and 'rtu' options. There is also a checked checkbox and an 'X' button.
- Steuerungs-Prioritäten: A section with three radio buttons labeled 1, 2, and 3. Below them are three groups of radio buttons: 'Rundsteuersignal Empfänger' (radio 1 selected), 'Dynamische Leistungsreduzierung' (radio 2 selected), and 'Steuerung über Modbus' (radio 3 selected).

On the right side of the interface, there is a sidebar with links like 'Aktuelle Daten', 'Aktuelle Gesamtansicht', 'Services', 'Systeminformationen', 'Netzwerkdagnostik', 'Firmware-Update', 'Assistenten aufrufen', and 'Einstellungen (1)'.



**HINWEIS!** Beim Fronius Datamanager 2.0 ist die 'Datenausgabe über Modbus' werkseitig auf rtu eingestellt.

Die Auswahlmöglichkeit rtu ist beim Fronius Datamanager nicht vorhanden.

## Datenausgabe über Modbus

### Modbus

(1) (2) (3)  
Datenausgabe über Modbus  aus  tcp  rtu

<input checked="" type="checkbox"/>	<input type="checkbox"/>
(5)	(6)

### Steuerungs-Prioritäten

	1	2	3
IO-Steuerung	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Dynamische Leistungsreduzierung	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
Steuerung über Modbus	<input type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>

Hinweis: eine Veränderung der Steuerungsprioritäten ist nur im EVU Editor Menü mit dem Service Passwort möglich.

Legende:  
1 ... höchste Priorität  
2 ... mittlere Priorität  
3 ... niedrigste Priorität

### Datenausgabe über Modbus

Aktivierung des Modbus Dienstes und Auswahl des Übertragungs-Protokolles.  
Wird der Modbus Dienst aktiviert, stehen weitere Eingabefelder zur Verfügung.

Das Übertragungs-Protokoll Modbus rtu ist nur beim Fronius Datamanager 2.0 verfügbar.



**HINWEIS!** Befindet sich ein unter Einstellungen / Zähler konfigurierter Modbus Energiezähler (z.B. Fronius Smart Meter) im System, kann die Einstellung „rtu“ nicht verwendet werden.

Bei Auswahl von „rtu“ wird in diesem Fall die Datenausgabe per Modbus automatisch deaktiviert. Diese Änderung ist erst nach einem erneuten Laden der Data-manager Web-Seite sichtbar.

Ein über RS485 an den Datamanager angeschlossener Energiezähler kann auch per Modbus TCP über die entsprechenden SunSpec Modelle ausgelesen werden. Die Modbus ID für den Zähler ist 240.

- (1) **aus**  
keine Datenausgabe über Modbus

Ist die Datenausgabe über Modbus deaktiviert, werden über Modbus an die Wechselrichter übertragene Steuerungsbefehle zurückgesetzt, z.B. keine Leistungsreduktion oder keine Blindleistungs-Vorgabe.

- (2) **tcp**  
Datenausgabe über Modbus tcp

Datenausgabe über Modbus	<input checked="" type="radio"/> aus <input type="radio"/> tcp <input type="radio"/> rtu
Modbus Port	<input type="text" value="502"/> (2a)
String Control Adress-Offset	<input type="text" value="101"/> (2b)
Sunspec Model Type	(2c) <input checked="" type="radio"/> float <input type="radio"/> int + SF (2d)
Demo Modus	<input type="checkbox"/> (2e)
Wechselrichter-Steuerung über Modbus	<input checked="" type="checkbox"/> (2f)

(2a) **Modbus Port**

Nummer des TCP Ports, der für die Modbus-Kommunikation zu verwenden ist.

Voreinstellung: 502

Port 80 kann hierfür nicht verwendet werden.

(2b) **String Control Adress-Offset**

Offset-Wert für die Adressierung von Fronius String Controls per Modbus.

Für weitere Details siehe Abschnitt "Modbus Geräte-ID für Fronius String Controls".

#### Sunspec Model Type

zum Auswählen des Datentyps von Datenmodellen für Wechselrichter und von Datenmodellen für Energiezähler

(2c) **float**

Darstellung als Gleitkommazahlen

SunSpec Inverter Model I111, I112 oder I113

SunSpec Meter Model M211, M212 oder M213

(2d) **int+SF**

Darstellung als ganze Zahlen mit Skalierungsfaktoren

SunSpec Inverter Model I101, I102 oder I103

SunSpec Meter Model M201, M202 oder M203

**WICHTIG!** Da die verschiedenen Modelle über unterschiedliche Anzahlen an Registern verfügen, ändern sich durch den Wechsel des Datentyps auch die Registeradressen aller nachfolgenden Modelle.

(2e) **Demo Modus**

Der Demo Modus dient zur Implementierung oder Validierung eines Modbus Masters. Er ermöglicht es, Wechselrichter-, Energiezähler- und String Control Daten auszulesen, ohne dass ein Gerät wirklich angeschlossen oder aktiv ist. Es werden für alle Register immer dieselben Daten zurückgeliefert.

(2f) **Wechselrichter-Steuerung über Modbus**

Wenn diese Option aktiviert ist, können die Wechselrichter über Modbus gesteuert werden.

Das Auswahlfeld Steuerung einschränken wird angezeigt.

Zur Wechselrichter-Steuerung gehören folgende Funktionen:

- Ein / Aus
- Leistungsreduktion
- Vorgabe eines konstanten Leistungs-Faktors cos Phi
- Vorgabe einer konstanten Blindleistung

(3) **rtu**

Datenausgabe über Modbus rtu

Datenausgabe über Modbus

(3)

aus  tcp  rtu

Baudrate

(3a)

9600

Parität

(3b)

keine

String Control Adress-Offset

(3c)

101

Sunspec Model Type

(3d)

float  int + SF

Demo Modus

(3e)

Wechselrichter-Steuerung über Modbus

(3f)

Hinweis: bei Anbindung eines Fronius Smart Meters wird Modbus RTU automatisch deaktiviert.

- (3a) **Baudrate**  
zum Eingeben der Baudrate,
- (3b) **Parität**  
Auswahlfeld zum Eingeben der Parität
- (3c) **String Control Adress-Offset**  
Offset-Wert für die Adressierung von Fronius String Controls per Modbus.  
Für weitere Details siehe Abschnitt "Modbus Geräte-ID für Fronius String Controls".

#### **Sunspec Model Type**

zum Auswählen des Datentyps von Datenmodellen für Wechselrichter

- (3d) **float**  
Darstellung als Gleitkommazahlen  
SunSpec Inverter Model I111, I112 oder I113
- (3e) **int+SF**  
Darstellung als ganze Zahlen mit Skalierungsfaktoren  
SunSpec Inverter Model I101, I102 oder I103

**WICHTIG!** Da die verschiedenen Modelle über unterschiedliche Anzahlen an Registern verfügen, ändern sich durch den Wechsel des Datentyps auch die Registeradressen aller nachfolgenden Modelle.

- (3f) **Demo Modus**  
Der Demo Modus dient zur Implementierung und Validierung eines Modbus Masters. Er ermöglicht es, Wechselrichter-, Energiezähler- und String Control Daten auszulesen, ohne dass ein Gerät wirklich angeschlossen oder aktiv ist. Es werden für alle Register immer dieselben Daten zurückgeliefert.
- (3g) **Wechselrichter-Steuerung über Modbus**  
Wenn diese Option aktiviert ist, erfolgt die Wechselrichter-Steuerung über Modbus.  
Zur Wechselrichter-Steuerung gehören folgende Funktionen:
  - Ein / Aus
  - Leistungsreduktion
  - Vorgabe eines konstanten Power Factors (cos Phi)
  - Vorgabe einer konstanten Blindleistung

- (4) **Steuerungs-Prioritäten**  
Die Steuerungs-Prioritäten legen fest, welcher Dienst bei der Wechselrichtersteuerung priorisiert wird.

1 = höchste Priorität, 3 = niedrigste Priorität

Die Steuerungs-Prioritäten können nur im Menüpunkt **EVU EDITOR** geändert werden.

- (5) **Schaltfläche Übernehmen / Speichern**
- (6) **Schaltfläche Abbrechen / Eingaben verwerfen**

#### **Steuerung einschränken**

Die Option "Steuerung einschränken" ist nur beim Übertragungsprotokollen tcp verfügbar. Sie dient dazu Wechselrichter-Steuerungsbefehle durch Unbefugte zu verhindern, indem die Steuerung nur für bestimmte Geräte erlaubt wird.

Wechselrichter-Steuerung über Modbus	<input checked="" type="checkbox"/>	
Steuerung einschränken	<input checked="" type="checkbox"/>	(1)
IP-Adresse	<input type="text" value="XX.X.XX.X"/>	(2)

(1) **Steuerung einschränken**  
Wenn diese Option aktiviert ist, dürfen nur bestimmte Geräte Steuerungsbefehle schicken.

(2) **IP-Adresse**  
Um die Wechselrichter-Steuerung auf ein oder mehrere Geräte zu beschränken, werden in diesem Feld die IP-Adressen jener Geräte eingetragen die Befehle an den Fronius Datamanager senden dürfen. Mehrere Einträge werden durch Beistriche getrennt.

Beispiele:

- eine IP-Adresse: **98.7.65.4**
  - Steuerung nur durch IP Adresse 98.7.65.4 zulässig
- mehrere IP-Adressen: **98.7.65.4,222.44.33.1**
  - Steuerung nur durch IP Adressen 98.7.65.4 und 222.44.33.1 zulässig
- IP-Adressbereich z.B. von 98.7.65.1 bis 98.7.65.254 (CIDR Notation): **98.7.65.0/24**
  - Steuerung nur durch IP Adressen 98.7.65.1 bis 98.7.65.254 zulässig

**Änderungen speichern oder verworfen**



Speichert die Einstellungen und zeigt eine Meldung an, dass die Speicherung erfolgreich war.

Wird der Menüpunkt "Modbus" verlassen ohne zu speichern, so werden alle vorgenommenen Änderungen verworfen.



Stellt eine Sicherheitsabfrage ob die vorgenommenen Änderungen tatsächlich verworfen werden sollen, und stellt dann die zuletzt gespeicherten Werte wieder her.

# Fronius Register

## Fronius Register



**HINWEIS!** Diese Register gelten nur für Wechselrichter. Für Fronius String Controls und Energiezähler sind diese Register nicht relevant.

Start	End	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
212	212	1	RW	0x03 0x06 0x10	F_Delete_Data	uint16			Delete stored data (e.g. ratings, curve data) of the current inverter by writing 0xFFFF.	0xFFFF
213	213	1	RW	0x03 0x06 0x10	F_Store_Data	uint16			Rating data of all inverters connected to the Fronius Datamanager are persistently stored by writing 0xFFFF.	0xFFFF
214	214	1	R	0x03	F_Active_Status_Code	uint16			Current active state code of inverter - Description can be found in inverter manual	
215	215	1	RW	0x03 0x06 0x10	F_Reset_All_Event_Flags	uint16			Write 0xFFFF to reset all event flags and active state code.	0xFFFF
216	216	1	RW	0x03 0x06 0x10	F_ModelType	uint16				1: Float 2: Int & SF
500	501	2	R	0x03	F_Site_Power	uint32	W			
502	505	4	R	0x03	F_Site_Energy_Day	uint64	Wh			
506	509	4	R	0x03	F_Site_Energy_Year	uint64	Wh			
510	513	4	R	0x03	F_Site_Energy_Total	uint64	Wh			

### Status-Code des Wechselrichters

Das Register **F\_Active\_Status\_Code** (214) zeigt den Status-Code des Wechselrichter an der gerade aufgetreten ist. Dieser wird eventuell auch am Display des Wechselrichter angezeigt. Dieser Code wird auch als Event Flag im Inverter Modell dargestellt. Der angezeigte Code bleibt so lange aktiv bis der entsprechende Status nicht mehr am Wechselrichter anliegt. Alternativ kann der Status auch per Register **F\_Reset\_All\_Event\_Flags** gelöscht werden.

### Löschen der Event Flags und des Status-Codes

Die Event Flags in den Inverter Models (101, 102, 103 und 111, 112, 113) bleiben so lange aktiv bis der entsprechende Status nicht mehr am Wechselrichter anliegt. Es gibt einige wenige Ausnahmen, wo die Event Flags nicht mehr gelöscht werden. Daher können die Event Flags und der angezeigte Status-Code per Modbus-Befehl zurückgesetzt werden.

## **[1] 0xFFFF in das Register *F\_Reset\_All\_Event\_Flags* (215) schreiben**

Der Inhalt folgender Register wird gelöscht:

- *F\_Active\_State\_Code* (214)
  - *Evt1*
  - *Evt2*
  - *EvtVnd1* bis *EvtVnd4*
- 

## **Daten speichern und löschen**

Schreibt man in das Register ***F\_Store\_Data*(213)** den Wert 0xFFFF werden alle Nennwerte (Ratings) für alle Wechselrichter am Fronius Datamanager gespeichert. Diese Werte können in den entsprechenden Registern des Nameplate Models und des Basic Settings Models verändert werden. Dies kann nützlich sein, wenn z. B. für ein Gerät keine Nennwerte automatisch ermittelt werden konnten und man die Werte manuell eintragen will.

Will man die gespeicherten Werte für einen bestimmten Wechselrichter löschen, muss man in das Register ***F\_Delete\_Data*(212)** den Wert 0xFFFF schreiben. Dann werden die Werte nur für diesen Wechselrichter gelöscht. Das Löschen kann immer nur auf den Wechselrichter angewendet werden, mit dem gerade kommuniziert wird.

---

## **Datentyp ändern**

Über das Register ***F\_ModelType*(216)** kann der Datentyp für die Datenmodelle für Wechselrichter und Energiezähler ausgewählt werden. Entweder kann die Darstellung als Gleitkommazahlen (float, Standard) oder als ganze Zahlen mit Skalierungsfaktoren (int+SF) ausgewählt werden.



**HINWEIS!** Diese Einstellung betrifft nur das Inverter Model (Wechselrichter) und das Meter Model (Energiezähler). Alle anderen Models verwenden weiterhin ganze Zahlen und Skalierungsfaktoren.

Diese Einstellung funktioniert gleich wie die über das Webinterface Modbus Einstellungen - SunSpec Model Type.

Einstellmöglichkeiten:

- Float = 1 (Standard): Inverter Model I111, I112 oder I113; Meter Model M211, M212 oder M213
- int+SF = 2: Inverter Model I101, I102 oder I103; Meter Model M201, M202 oder M203



**HINWEIS!** Da die verschiedenen Models über eine unterschiedliche Anzahl an Registern verfügen, ändern sich durch den Wechsel des Datentyps auch die Registeradressen aller nachfolgenden Models.

---

## **Anlagensummen**

Über die folgenden Register können Leistungs- und Energiedaten von allen per Solar Net mit diesem Fronius Datamanager verbundenen Wechselrichtern, abgefragt werden. Die Werte werden in Watt (W) bzw. Wattstunden (Wh) abgebildet und benötigen keine Skalierungsfaktoren.

- ***F\_Site\_Power*(500-501)**: Leistung
- ***F\_Site\_Energy\_Day*(502-505)**: Tagesenergie
- ***F\_Site\_Energy\_Year*(506-509)**: Jahresenergie
- ***F\_Site\_Energy\_Total*(510-513)**: Gesamtenergie der gesamten Anlage

# Common & Inverter Model

## Common Block Register

Die Beschreibung des Common Block inklusive der SID Register (Register 40001-40002) zur Identifizierung als SunSpec Gerät gilt für jeden Gerätetyp (Wechselrichter, String Control, Energiezähler). Jedes Gerät besitzt einen eigenen Common Block, in dem Informationen über das Gerät (Modell, Seriennummer, SW Version, etc.) aufgeführt sind.

Start	End	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
40001	40002	2	R	0x03	SID	uint32			Well-known value. Uniquely identifies this as a SunSpec Modbus Map	0x53756e53 (SunS)
40003	40003	1	R	0x03	ID	uint16			Well-known value. Uniquely identifies this as a SunSpec Common Model block	1
40004	40004	1	R	0x03	L	uint16	Registers		Length of Common Model block	65
40005	40020	16	R	0x03	Mn	String32			Manufacturer	z.B. Fronius
40021	40036	16	R	0x03	Md	String32			Device model	z.B. IG+150V
40037	40044	8	R	0x03	Opt	String16			SW version of datamanager	z.B. 3.3.6-13
40045	40052	8	R	0x03	Vr	String16			SW version of inverter	
40053	40068	16	R	0x03	SN	String32			Serialnumber of inverter, string control or energy meter	
40069	40069	1	R	0x03	DA	uint16			Modbus Device Address	1 - 247

## Inverter Model Register

Für die Wechselrichter-Daten werden zwei verschiedene SunSpec Models unterstützt:

- das standardmäßig eingestellte Inverter Model mit Gleitkomma-Darstellung (Einstellung „float“; I111, I112 oder I113)
- das Inverter Model mit ganzen Zahlen und Skalierungsfaktoren (Einstellung „int+SF“; I101, I102 oder I103)



**HINWEIS!** Die Registeranzahl der beiden Model-Typen ist unterschiedlich!

### Floating Point Darstellung – „float“

Start	End	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
40070	40070	1	R	0x03	ID	uint16			Uniquely identifies this as a SunSpec Inverter Modbus Map 111: single phase 112: split phase 113: three phase	111, 112, 113
40071	40071	1	R	0x03	L	uint16	Registers		Length of inverter model block	60
40072	40073	2	R	0x03	A	float32	A		AC Total Current value	
40074	40075	2	R	0x03	AphA	float32	A		AC Phase-A Current value	
40076	40077	2	R	0x03	AphB	float32	A		AC Phase-B Current value	
40078	40079	2	R	0x03	AphC	float32	A		AC Phase-C Current value	
40080	40081	2	R	0x03	PPVphAB	float32	V		AC Voltage Phase-AB value	
40082	40083	2	R	0x03	PPVphBC	float32	V		AC Voltage Phase-BC value	
40084	40085	2	R	0x03	PPVphCA	float32	V		AC Voltage Phase-CA value	
40086	40087	2	R	0x03	PhVphA	float32	V		AC Voltage Phase-A-to-neutral value	
40088	40089	2	R	0x03	PhVphB	float32	V		AC Voltage Phase-B-to-neutral value	
40090	40091	2	R	0x03	PhVphC	float32	V		AC Voltage Phase-C-to-neutral value	
40092	40093	2	R	0x03	W	float32	W		AC Power value	
40094	40095	2	R	0x03	Hz	float32	Hz		AC Frequency value	
40096	40097	2	R	0x03	VA	float32	VA		Apparent Power	
40098	40099	2	R	0x03	VAr	float32	VAr		Reactive Power	
40100	40101	2	R	0x03	PF	float32	%		Power Factor	
40102	40103	2	R	0x03	WH	float32	Wh		AC Lifetime Energy production	
40104	40105	2	R	0x03	DCA	float32	A		DC Current value	DC current only if one MPPT available; with multiple MPPT "not implemented"

### Floating Point Darstellung – „float“

Start	End	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
40106	40107	2	R	0x03	DCV	float32	V		DC Voltage value	DC voltage only if one MPPT available; with multiple MPPT "not implemented"
40108	40109	2	R	0x03	DCW	float32	W		DC Power value	Total DC power of all available MPPT
40110	40111	2	R	0x03	TmpCab	float32	° C		Cabinet Temperature	
40112	40113	2	R	0x03	TmpSnk	float32	° C		Coolant or Heat Sink Temperature	
40114	40115	2	R	0x03	TmpTrns	float32	° C		Transformer Temperature	
40116	40117	2	R	0x03	TmpOt	float32	° C		Other Temperature	
40118	40118	1	R	0x03	St	enum16	Enumerated		Operating State	1)
40119	40119	1	R	0x03	StVnd	enum16	Enumerated		Vendor Defined Operating State	2)
40120	40121	2	R	0x03	Evt1	unit32	Bitfield		Event Flags (bits 0-31)	3)
40122	40123	2	R	0x03	Evt2	unit32	Bitfield		Event Flags (bits 32-63)	
40124	40125	2	R	0x03	EvtVnd1	unit32	Bitfield		Vendor Defined Event Flags (bits 0-31)	
40126	40127	2	R	0x03	EvtVnd2	unit32	Bitfield		Vendor Defined Event Flags (bits 32-63)	3)
40128	40129	2	R	0x03	EvtVnd3	unit32	Bitfield		Vendor Defined Event Flags (bits 64-95)	
40130	40131	2	R	0x03	EvtVnd4	unit32	Bitfield		Vendor Defined Event Flags (bits 96-127)	

1) siehe SunSpec State Codes

2) siehe Fronius State Codes

3) Detaillierte Informationen und Listen in verschiedenen Dateiformaten (xlsx, csv, json) können von der Fronius Homepage heruntergeladen werden:

[www.fronius.com - Solar Electronics - Info & Support - Third-party Downloads](http://www.fronius.com - Solar Electronics - Info & Support - Third-party Downloads)

### Ganzzahldarstellung – „int+SF“

Start	End	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
40070	40070	1	R	0x03	ID	uint16			Uniquely identifies this as a SunSpec Inverter Modbus Map 101: single phase 102: split phase 103: three phase	101, 102, 103
40071	40071	1	R	0x03	L	uint16	Registers		Length of inverter model block	50
40072	40072	2	R	0x03	A	uint16	A	A_SF	AC Total Current value	
40073	40073	2	R	0x03	AphA	uint16	A	A_SF	AC Phase-A Current value	
40074	40074	2	R	0x03	AphB	uint16	A	A_SF	AC Phase-B Current value	
40075	40075	2	R	0x03	AphC	uint16	A	A_SF	AC Phase-C Current value	
40076	40076	2	R	0x03	A_SF	sunssf			AC Current Scale factor	
40077		1	R	0x03	PPVphAB	uint16	V	V_SF	AC Voltage Phase-AB value	
40078	40078	1	R	0x03	PPVphBC	uint16	V	V_SF	AC Voltage Phase-BC value	
40079	40079	1	R	0x03	PPVphCA	uint16	V	V_SF	AC Voltage Phase-CA value	
40080	40080	1	R	0x03	PhVphA	uint16	V	V_SF	AC Voltage Phase-A-to-neutral value	
40081	40081	1	R	0x03	PhVphB	uint16	V	V_SF	AC Voltage Phase-B-to-neutral value	
40082	40082	1	R	0x03	PhVphC	uint16	V	V_SF	AC Voltage Phase-C-to-neutral value	
40083	40083	1	R	0x03	V_SF	sunssf			AC Voltage Scale factor	
40084	40084	1	R	0x03	W	int16	W	W_SF	AC Power value	
40085	40085	1	R	0x03	W_SF	sunssf			AC Power Scale factor	
40086	40086	1	R	0x03	Hz	uint16	Hz	Hz_SF	AC Frequency value	
40087	40087	1	R	0x03	Hz_SF	sunssf			Scale factor	
40088	40088	1	R	0x03	VA	int16	VA	VA_SF	Apparent Power	
40089	40089	1	R	0x03	VA_SF	sunssf			Scale factor	
40090	40090	1	R	0x03	VAr	int16	VAr	VAr_SF	Reactive Power	
40091	40091	1	R	0x03	VAr_SF	sunssf			Scale factor	
40092	40092	1	R	0x03	PF	int16	%	PF_SF	Power Factor	
40093	40093	1	R	0x03	PF_SF	sunssf			Scale factor	
40094	40095	2	R	0x03	WH	uint32	Wh	WH_SF	AC Lifetime Energy production	
40096	40096	1	R	0x03	WH_SF	sunssf			AC Lifetime Energy production scale factor	

**Ganzzahldarstellung – „int+SF“**

<b>Start</b>	<b>End</b>	<b>Size</b>	<b>R/W</b>	<b>Function codes</b>	<b>Name</b>	<b>Type</b>	<b>Units</b>	<b>Scale factor</b>	<b>Description</b>	<b>Range of values</b>
40097	40097	1	R	0x03	DCA	uint16	A	DCA_SF	DC Current value	DC current only if one MPPT available; with multiple MPPT "not implemented"
40098	40098	1	R	0x03	DCA_SF	sunssf			Scale factor	Scale factor for DC current only if one MPPT available; with multiple MPPT "not implemented"
40099	40099	1	R	0x03	DCV	uint16	V	DCV_SF	DC Voltage value	DC voltage only if one MPPT available; with multiple MPPT "not implemented"
40100	40100	1	R	0x03	DCV_SF	sunssf			Scale factor	Scale factor for DC voltage only if one MPPT available; with multiple MPPT "not implemented"
40101	40101	1	R	0x03	DCW	int16	W	DCW_SF	DC Power value	Total DC power of all available MPPT
40102	40102	1	R	0x03	DCW_SF	sunssf			Scale factor	
40103	40103	1	R	0x03	TmpCab	int16	° C	Tmp_SF	Cabinet Temperature	Not supported
40104	40104	1	R	0x03	TmpSnk	int16	° C	Tmp_SF	Coolant or Heat Sink Temperature	Not supported
40105	40105	1	R	0x03	TmpTrns	int16	° C	Tmp_SF	Transformer Temperature	Not supported
40106	40106	1	R	0x03	TmpOt	int16	° C	Tmp_SF	Other Temperature	Not supported
40107	40107	1	R	0x03	Tmp_SF	sunssf			Scale factor	Not supported

### Ganzzahldarstellung – „int+SF“

Start	End	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
40108	40108	1	R	0x03	St	enum16	Enum-erated		Operating State	1)
40109	40109	1	R	0x03	StVnd	enum16	Enum-erated		Vendor Defined Operating State	2)
40110	40111	2	R	0x03	Evt1	uint32	Bit-field		Event Flags (bits 0-31)	3)
40112	40113	2	R	0x03	Evt2	uint32	Bit-field		Event Flags (bits 32-63)	
40114	40115	2	R	0x03	EvtVnd1	uint32	Bit-field		Vendor Defined Event Flags (bits 0-31)	
40116	40117	2	R	0x03	EvtVnd2	uint32	Bit-field		Vendor Defined Event Flags (bits 32-63)	3)
40118	40119	2	R	0x03	EvtVnd3	uint32	Bit-field		Vendor Defined Event Flags (bits 64-95)	
40120	40121	2	R	0x03	EvtVnd4	uint32	Bit-field		Vendor Defined Event Flags (bits 96-127)	

1) siehe SunSpec State Codes

2) siehe Fronius State Codes

3) Detaillierte Informationen und Listen in verschiedenen Dateiformaten (xlsx, csv, json) können von der Fronius Homepage heruntergeladen werden:

[www.fronius.com - Solar Electronics - Info & Support - Third-party Downloads](http://www.fronius.com - Solar Electronics - Info & Support - Third-party Downloads)

#### SunSpec State Codes

Name	Wert	Beschreibung
I_STATUS_OFF	1	Wechselrichter ist aus
I_STATUS_SLEEPING	2	Auto-Shutdown
I_STATUS_STARTING	3	Wechselrichter startet
I_STATUS_MPPT	4	Wechselrichter arbeitet normal
I_STATUS_THROTTLED	5	Leistungsreduktion aktiv
I_STATUS_SHUTTING_DOWN	6	Wechselrichter schaltet ab
I_STATUS_FAULT	7	Ein oder mehr Fehler existieren, siehe St *oder Evt * Register
I_STATUS_STANDBY	8	Standby

\* Inverter Model Register

**Fronius State Codes**

Name	Wert	Beschreibung
I_STATUS_OFF	1	Wechselrichter ist aus
I_STATUS_SLEEPING	2	Auto-Shutdown
I_STATUS_STARTING	3	Wechselrichter startet
I_STATUS_MPPT	4	Wechselrichter arbeitet normal
I_STATUS_THROTTLED	5	Leistungsreduktion aktiv
I_STATUS_SHUTTING_DOWN	6	Wechselrichter schaltet ab
I_STATUS_FAULT	7	Ein oder mehr Fehler existieren, siehe St * oder Evt * Register
I_STATUS_STANDBY	8	Standby
I_STATUS_NO_BUSINIT	9	Keine SolarNet Kommunikation
I_STATUS_NO_COMM_INV	10	Keine Kommunikation mit Wechselrichter möglich
I_STATUS_SN_OVERCURRENT	11	Überstrom an SolarNet Stecker erkannt
I_STATUS_BOOTLOAD	12	Wechselrichter wird gerade upgedatet
I_STATUS_AFCI	13	AFCI Event (Arc-Erkennung)

\* Inverter Model Register

# Nameplate Model (IC120)

---

<b>Allgemeines</b>	Dieses Modell entspricht einem Leistungsschild. Folgende Daten können ausgelesen werden:
-	<b>DERType (3)</b> Art des Geräts. Das Register liefert den Wert 4 zurück (PV-Gerät)
-	<b>WRtg (4)</b> Nennleistung des Wechselrichters
-	<b>VARtg (6)</b> Nenn-Scheinleistung des Wechselrichters
-	<b>VArRtgQ1 (8) - VArRtgQ4 (11)</b> Nenn-Blindleistungswerte für die vier Quadranten
-	<b>ARtg (13)</b> Nennstrom des Wechselrichters
-	<b>PFRtgQ1 (15) – PFRtgQ4 (18)</b> Minimale Werte für den Power Factor für die vier Quadranten

---

<b>Nameplate Register</b>	Startadresse:
-	bei Einstellung „float“: <b>40131</b>
-	bei Einstellung „int+SF“: <b>40121</b>

Start Offset	End Offset	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
1	1	1	R	0x03	ID	uint16			A well-known value 120. Uniquely identifies this as a SunSpec Nameplate Model	120
2	2	1	R	0x03	L	uint16	Registers		Length of Nameplate Model	26
3	3	1	R	0x03	DERTyp	enum16			Type of DER device. Default value is 4 to indicate PV device.	4
4	4	1	R	0x03	WRtg	uint16	W	WRtg_SF	Continuous power output capability of the inverter.	
5	5	1	R	0x03	WRtg_SF	sunssf			Scale factor	1
6	6	1	R	0x03	VARtg	uint16	VA	VARtg_SF	Continuous Volt-Amperie capability of the inverter.	
7	7	1	R	0x03	VArRtgQ1	int16	var	VArRtg_SF	Continuous VAR capability of the inverter in quadrant 1.	
8	8	1	R	0x03	VArRtgQ2	int16	var	VArRtg_SF	Continuous VAR capability of the inverter in quadrant 2.	Not supported

Start Offset	End Offset	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
10	10	1	R	0x03	VArRtgQ3	int16	var	VArRtg_SF	Continuous VAR capability of the inverter in quadrant 3.	Not supported
11	11	1	R	0x03	VArRtgQ4	int16	var	VArRtg_SF	Continuous VAR capability of the inverter in quadrant 4.	
12	12	1	R	0x03	VArRtg_SF	sunssf			Scale factor	1
13	13	1	R	0x03	ARtg	uint16	A	ARtg_SF	Maximum RMS AC current level capability of the inverter.	
14	14	1	R	0x03	ARtg_SF	sunssf			Scale factor	-2
15	15	1	R	0x03	PFRtgQ1	int16	cos()	PFRtg_SF	Minimum power factor capability of the inverter in quadrant 1.	
16	16	1	R	0x03	PFRtgQ2	int16	cos()	PFRtg_SF	Minimum power factor capability of the inverter in quadrant 2.	Not supported
17	17	1	R	0x03	PFRtgQ3	int16	cos()	PFRtg_SF	Minimum power factor capability of the inverter in quadrant 3.	Not supported
18	18	1	R	0x03	PFRtgQ4	int16	cos()	PFRtg_SF	Minimum power factor capability of the inverter in quadrant 4.	
19	19	1	R	0x03	PFRtg_SF	sunssf			Scale factor	-3
20	20	1	R	0x03	WHRtg	uint16	Wh	WHRtg_SF	Nominal energy rating of storage device.	*
21	21	1	R	0x03	WHRtg_SF	sunssf			Scale factor	0*
22	22	1	R	0x03	AhrRtg	uint16	AH	AhrRtg_SF	The useable capacity of the battery. Maximum charge minus minimum charge from a technology capability perspective (Amp-hour capacity rating).	Not supported
23	23	1	R	0x03	AhrRtg_SF	sunssf			Scale factor for amp-hour rating.	Not supported
24	24	1	R	0x03	MaxChaRte	uint16	W	MaxCharTe_SF	Maximum rate of energy transfer into the storage device.	*
25	25	1	R	0x03	MaxChaRte_SF	sunssf			Scale factor	0*
26	26	1	R	0x03	MaxDisChaRte	uint16	W	MaxDisCharTe_SF	Maximum rate of energy transfer out of the storage device.	*
27	27	1	R	0x03	MaxDisChaRte_SF	sunssf			Scale factor	0*
28	28	1	R	0x03	Pad				Pad register	

<b>Start Offset</b>	<b>End Offset</b>	<b>Size</b>	<b>R/W</b>	<b>Function codes</b>	<b>Name</b>	<b>Type</b>	<b>Units</b>	<b>Scale factor</b>	<b>Description</b>	<b>Range of values</b>
---------------------	-------------------	-------------	------------	-----------------------	-------------	-------------	--------------	---------------------	--------------------	------------------------

\* wird nur von Fronius Hybrid Wechselrichtern unterstützt

# Basic Settings Model (IC121)

## Basic Settings Register

Startadresse:

- bei Einstellung „float“: **40159**
- bei Einstellung „int+SF“: **40149**

Start Offset	End Offset	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
1	1	1	R	0x03	ID	uint16			A well-known value 121. Uniquely identifies this as a SunSpec Basic Settings Model	121
2	2	1	R	0x03	L	uint16	Registers		Length of Basic Settings Model	30
3	3	1	R	0x03	WMax	uint16	W	WMax_SF	Setting for maximum power output. Default to WRtg.	
4	4	1	RW	0x03 0x06 0x10	VRef	uint16	V	VRef_SF	Voltage at the PCC.	
5	5	1	RW	0x03 0x06 0x10	VRefOfs	int16	V	VRefOfs_SF	Offset from PCC to inverter.	
6	6	1	R	0x03	VMax	uint16	V	VMin-Max_SF	Currently not supported	
7	7	1	R	0x03	VMin	uint16	V	VMin-Max_SF	Currently not supported	
8	8	1	R	0x03	VAMax	unit16	VA	VA-Max_SF	Setpoint for maximum apparent power. Default to VARtg.	
9	9	1	R	0x03	VARMaxQ1	int16	var	VAR-Max_SF	Setting for maximum reactive power in quadrant 1. Default to VArRtgQ1.	
10	10	1	R	0x03	VARMaxQ2	int16	var	VAR-Max_SF	Setting for maximum reactive power in quadrant 2. Default to VArRtgQ2.	Not supported
11	11	1	R	0x03	VARMaxQ3	int16	var	VAR-Max_SF	Setting for maximum reactive power in quadrant 3. Default to VArRtgQ3.	Not supported
12	12	1	R	0x03	VARMaxQ4	int16	var	VAR-Max_SF	Setting for maximum reactive power in quadrant 4. Default to VArRtgQ4.	
13	13	1	R	0x03	WGra	uint16	% WMax/min	WGra_SF	Default ramp rate of change of active power due to command or internal action.	Not supported

<b>Start Offset</b>	<b>End Offset</b>	<b>Size</b>	<b>R/W</b>	<b>Function codes</b>	<b>Name</b>	<b>Type</b>	<b>Units</b>	<b>Scale factor</b>	<b>Description</b>	<b>Range of values</b>
14	14	1	R	0x03	PFMinQ1	int16	cos()	PF-Min_SF	Setpoint for minimum power factor value in quadrant 1. Default to PFRtgQ1.	
15	15	1	R	0x03	PFMinQ2	int16	cos()	PF-Min_SF	Setpoint for minimum power factor value in quadrant 2. Default to PFRtgQ2.	Not supported
16	16	1	R	0x03	PFMinQ3	int16	cos()	PF-Min_SF	Setpoint for minimum power factor value in quadrant 3. Default to PFRtgQ3.	Not supported
17	17	1	R	0x03	PFMinQ4	int16	cos()	PF-Min_SF	Setpoint for minimum power factor value in quadrant 4. Default to PFRtgQ4.	
18	18	1	R	0x03	VArAct	enum16			VAR action on change between charging and discharging: 1=switch 2=maintain VAR characterization.	Not supported
19	19	1	R	0x03	ClcTotVA	enum16			Calculation method for total apparent power. 1=vector 2=arithmetic.	Not supported
20	20	1	R	0x03	MaxRmpRte	uint16	%W Gra	MaxRm pr-te_SF	Setpoint for maximum ramp rate as percentage of nominal maximum ramp rate. This setting will limit the rate that watts delivery to the grid can increase or decrease in response to intermittent PV generation.	Not supported
21	21	1	R	0x03	ECPNomHz	uint16	Hz	ECP-NomH-z_SF	Setpoint for nominal frequency at the ECP.	Not supported
22	22	1	R	0x03	ConnPh	enum16			Identity of connected phase for single phase inverters. A=1 B=2 C=3.	Not supported
23	23	1	R	0x03	WMax_SF	sunssf			Scale factor for maximum power output.	1
24	24	1	R	0x03	VRef_SF	sunssf			Scale factor for voltage at the PCC.	0
25	25	1	R	0x03	VRefOfs_SF	sunssf			Scale factor for offset voltage.	0
26	26	1	R	0x03	VMinMax_SF	sunssf			Scale factor for min/max voltages.	0
27	27	1	R	0x03	VAMax_SF	sunssf			Scale factor for voltage at the PCC.	1

Start Offset	End Offset	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
28	28	1	R	0x03	VARMax_SF	sunssf			Scale factor for reactive power.	1
29	29	1	R	0x03	WGra_SF	sunssf			Scale factor for default ramp rate.	Not supported
30	30	1	R	0x03	PFMin_SF	sunssf			Scale factor for minimum power factor.	-3
31	31	1	R	0x03	MaxRmpRate_SF	sunssf			Scale factor for maximum ramp percentage.	Not supported
32	32	1	R	0x03	ECPNomHz_SF	sunssf			Scale factor for nominal frequency.	Not supported

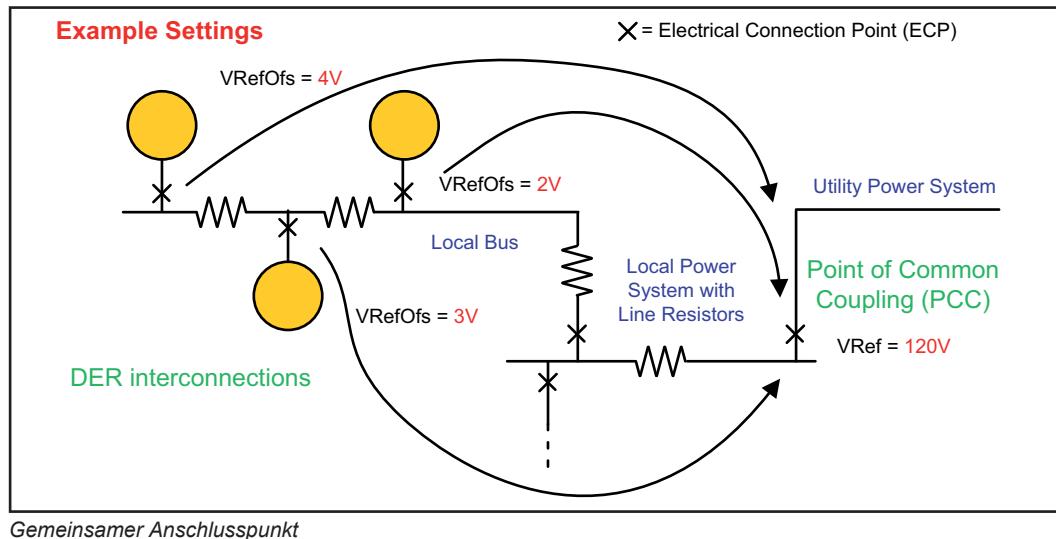
## Referenzspannung

### VRef (4)

Die Referenzspannung ist die Spannung an jenem gemeinsamen Anschlusspunkt, an welchem das lokale Netz mit dem öffentlichen Stromnetz verknüpft ist, und entspricht der Nennspannung des Wechselrichters.

=> siehe Abbildung "Gemeinsamer Anschlusspunkt"

Die Angabe erfolgt in Volt im Bereich von 0 (0x0000) bis 400 (0x0190).



## Abweichung zur Referenzspannung

### VRefOfs (5)

Je nach Verschaltung des lokalen Netzes kann es am Anschlusspunkt jedes einzelnen Wechselrichters an das lokale Netz zu einer Abweichung zur Referenzspannung kommen (siehe Abbildung "Gemeinsamer Anschlusspunkt").

Die Angabe erfolgt in Volt im Bereich -20 (0xFFEC) bis 20 (0x0014).

# Extended Measurements & Status Model (IC122)

---

<b>Allgemeines</b>	Dieses Modell liefert einige zusätzliche Mess- und Statuswerte, die das normale Inverter Model nicht abdeckt:
-	<b>PVConn (3)</b> Dieses Bitfeld zeigt den Status des Wechselrichter an <ul style="list-style-type: none"> <li>- Bit 0: Verbunden</li> <li>- Bit 1: Ansprechbar</li> <li>- Bit 2: Arbeitet (Wechselrichter speist ein)</li> </ul>
-	<b>ECPConn (5)</b> Dieses Register zeigt den Verbindungsstatus zum Netz an <ul style="list-style-type: none"> <li>- <i>ECPConn</i> = 1: Wechselrichter speist gerade ein</li> <li>- <i>ECPConn</i> = 0: Wechselrichter speist nicht ein</li> </ul>
-	<b>ActWH (6 - 9)</b> Wirkenergiezähler
-	<b>StActCtl (36 - 37)</b> Bitfeld für zurzeit aktive Wechselrichter-Modi <ul style="list-style-type: none"> <li>- Bit 0: Leistungsreduktion (FixedW; entspricht WMaxLimPct Vorgabe)</li> <li>- Bit 1: konstante Blindleistungs-Vorgabe (FixedVAR; entspricht VArMaxPct)</li> <li>- Bit 2: Vorgabe eines konstanten Power Factors (FixedPF; entspricht OutPFSet)</li> </ul>
-	<b>TmSrc (38 - 41)</b> Quelle für die Zeitsynchronisation. Das Register liefert den String „RTC“ zurück.
-	<b>Tms (42 - 43)</b> Aktuelle Uhrzeit und Datum der RTC Angegeben werden die Sekunden vom 1. Jänner 2000 00:00 (UTC) bis zur aktuellen Zeit

---

<b>Extended Measurements &amp; Status Register</b>	Startadresse:
	- bei Einstellung „float“: <b>40191</b>
	- bei Einstellung „int+SF“: <b>40181</b>

Start Offset	End Offset	Size	RW	Function codes	Name	Type	Units	Scale factor	Description	Range of values
1	1	1	R	0x03	ID	uint16			A well-known value 122. Uniquely identifies this as a SunSpec Measurements_Status Model	122
2	2	1	R	0x03	L	uint16	Registers		Length of Measurements_Status Model	44

Start Offset	End Offset	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
3	3	1	R	0x03	PVConn	bitfield16			PV inverter present/available status. Enumerated value.  Connected Available Operating Test	Bit 0 = 1 Bit 1 = 1 Bit 2 = 1 Bit 3 = 1
4	4	1	R	0x03	StorConn	bitfield16			Storage inverter present/available status. Enumerated value.  Connected Available Operating	Bit 0 = 1* Bit 1 = 1* Bit 2 = 1*
5	5	1	R	0x03	ECPConn	bitfield16			ECP connection status  Connected	Bit 0 = 1
6	9	4	R	0x03	ActWh	acc64	Wh		AC lifetime active (real) energy output.	
10	13	4	R	0x03	ActVAh	acc64	VAh		AC lifetime apparent energy output.	Not supported
14	17	4	R	0x03	ActVArhQ1	acc64	varh		AC lifetime reactive energy output in quadrant 1.	Not supported
18	21	4	R	0x03	ActVArhQ2	acc64	varh		AC lifetime reactive energy output in quadrant 2.	Not supported
22	25	4	R	0x03	ActVArhQ3	acc64	varh		AC lifetime negative energy output in quadrant 3.	Not supported
26	29	4	R	0x03	ActVArhQ4	acc64	varh		AC lifetime reactive energy output in quadrant 4.	Not supported
30	30	1	R	0x03	VArAval	int16	var	VArA-val_SF	Amount of VARs available without impacting watts output.	Not supported
31	31	1	R	0x03	VArAval_SF	sunssf			Scale factor for available VARs.	Not supported
32	32	1	R	0x03	WAval	uint16	W	WA-val_SF	Amount of Watts available.	Not supported
33	33	1	R	0x03	WAval_SF	sunssf			Scale factor for available Watts.	Not supported
34	35	2	R	0x03	StSetLimMsk	bitfield32			Bit Mask indicating set-point limit(s) reached. Bits are persistent and must be cleared by the controller.	Not supported

Start Offset	End Offset	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
36	37	2	R	0x03	StActCtl	bitfield32			Bit Mask indicating which inverter controls are currently active. FixedW FixedVAR FixedPF	Bit 0 = 1 Bit 1 = 1 Bit 2 = 1
38	41	4	R	0x03	TmSrc	string			Source of time synchronization.	RTC
42	43	2	R	0x03	Tms	uint32	Secs		Seconds since 01-01-2000 00:00 UTC	
44	44	1	R	0x03	RtSt	bitfield16			Bit Mask indicating which voltage ride through modes are currently active.	0
45	45	1	R	0x03	Riso	uint16	Ohm	Riso_SF	Isolation resistance	Not supported
46	46	1	R	0x03	Riso_SF	int16			Scale factor for Isolation resistance	Not supported

\* wird nur von Fronius Hybrid Wechselrichtern unterstützt

# Immediate Controls Model (IC123)

Allgemeines	Mit den Immediate Controls können folgende Einstellungen am Wechselrichter vorgenommen werden: <ul style="list-style-type: none"> <li>- Unterbrechung des Einspeisebetriebs des Wechselrichters (Standby)</li> <li>- Konstante Reduktion der Ausgangsleistung</li> <li>- Vorgabe eines konstanten Power Factors</li> <li>- Vorgabe einer konstanten relativen Blindleistung</li> </ul>
-------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Immediate Controls Register	Startadresse: - bei Einstellung „float“: <b>40237</b> - bei Einstellung „int+SF“: <b>40227</b>
-----------------------------	------------------------------------------------------------------------------------------------------

Start Offset	End Offset	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
1	1	1	R	0x03	ID	uint16			A well-known value 123. Uniquely identifies this as a SunSpec Immediate Controls Model	123
2	2	1	R	0x03	L	uint16	Registers		Length of Immediate Controls Model	24
3	3	1	RW	0x03 0x06 0x10	Conn_WinTms	uint16	Secs		Time window for connect/disconnect.	0 - 300
4	4	1	RW	0x03 0x06 0x10	Conn_RvrTms	uint16	Secs		Timeout period for connect/disconnect.	0 - 28800
5	5	1	RW	0x03 0x06 0x10	Conn	enum16			Enumerated valued. Connection control.  Disconnected Connected	0 1
6	6	1	RW	0x03 0x06 0x10	WMaxLimPct	uint16	% WMax	WMaxLimPct_SF	Set power output to specified level.	
7	7	1	RW	0x03 0x06 0x10	WMaxLimPct_WinTms	uint16	Secs		Time window for power limit change.	0 - 300
8	8	1	RW	0x03 0x06 0x10	WMaxLimPct_RvrTms	uint16	Secs		Timeout period for power limit.	0 - 28800
9	9	1	R	0x03	WMaxLimPct_RmpTms	uint16	Secs		Ramp time for moving from current setpoint to new setpoint.	Not supported

Start Offset	End Offset	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
10	10	1	RW	0x03 0x06 0x10	WMaxLim_Ena	enum16			Enumerated valued. Throttle enable/disable control.  Disabled Enabled	0 1
11	11	1	RW	0x03 0x06 0x10	OutPFSet	int16	cos()	OutPF- Set_SF	Set power factor to specific value - cosine of angle.	
12	12	1	RW	0x03 0x06 0x10	OutPF- Set_WinTms	uint16	Secs		Time window for power factor change.	0 – 300
13	13	1	RW	0x03 0x06 0x10	OutPFSet_Rv- rtTms	uint16	Secs		Timeout period for power factor.	0 – 28800
14	14	1	R	0x03	OutPF- Set_RmpTms	uint16	Secs		Ramp time for moving from current setpoint to new setpoint.	Not supported
15	15	1	RW	0x03 0x06 0x10	OutPFSet_Ena	enum16			Enumerated valued. Fixed power factor enable/disable control.  Disabled Enabled	0 1
16	16	1	R	0x03	VArWMaxPct	int16	% WMax	VArW- MaxPc- t_SF	Reactive power in percent of WMax.	Not supported
17	17	1	RW	0x03 0x06 0x10	VArMaxPct	int16	% VAr- Max	VArPc- t_SF	Reactive power in percent of VArMax.	
18	18	1	R	0x03	VArAvalPct	int16	% VA- rAval	VArPc- t_SF	Reactive power in percent of VArAval.	Not supported
19	19	1	RW	0x03 0x06 0x10	VArPct_WinT- ms	uint16	Secs		Time window for VAR limit change.	0 – 300
20	20	1	R	0x03	VArPct_RvrT- ms	uint16	Secs		Timeout period for VAR limit.	0 – 28800
21	21	1	RW	0x03 0x06 0x10	VArPct_RmpT- ms	uint16	Secs		Ramp time for moving from current setpoint to new setpoint.	Not supported
22	22	1	R	0x03	VArPct_Mod	enum16			Enumerated value. VAR limit mode.	2: VAR limit as a % of VArMax
23	23	1	RW	0x03 0x06 0x10	VArPct_Ena	enum16			Enumerated valued. Fixed VAR enable/disable control.  Disabled Enabled	0 1
24	24	1	R	0x03	WMaxLimPc- t_SF	sunssf			Scale factor for power output percent.	-2

Start Offset	End Offset	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
25	25	1	R	0x03	OutPFSet_SF	sunssf			Scale factor for power factor.	-3
26	26	1	R	0x03	VArPct_SF	sunssf			Scale factor for reactive power.	0

**Standby****Conn\_WinTms (3) bis Conn (5)**

Diese Register dienen zur Steuerung des Standby Modus (kein Einspeisebetrieb) des Wechselrichters.

**Conn\_WinTms (3) und Conn\_RvrTms (4)**

Mit diesen Registern kann das Verhalten des Wechselrichters zeitlich gesteuert werden.  
=> siehe Abschnitt "Zeitverhalten der unterstützten Betriebsarten".

Als Standard ist für alle Register 0 vorgegeben.

**Conn (5)**

Register Conn zeigt an, ob der Wechselrichter aktuell einspeist (0 = Standby, 1 = Einspeisebetrieb).

- Um den Wechselrichter in den Standby zu schalten schreibt man in dieses Register den Wert 0
- Um den Wechselrichter wieder zu aktivieren schreibt man in dieses Register den Wert 1



**HINWEIS!** Ob der Wechselrichter einspeist oder nicht kann auch über das Register ECPConn aus dem Extended Measurements and Status Model ausgelesen werden.

**Leistungsreduktion****WMaxLimPct (6) bis WMaxLim\_Ena (10)**

Über diese Register kann beim Wechselrichter eine Reduktion der Ausgangsleistung eingestellt werden.

**WMaxLimPct (6)**

In Register WMaxLimPct können Werte zwischen 0% und 100% eingetragen werden. Abhängig von der Software-Version des Wechselrichters können Werte kleiner als 10 zu einem erzwungenen Standby des Wechselrichters führen (kein Einspeisebetrieb).

Die Werte beschränken die maximal mögliche Ausgangsleistung des Gerätes, und haben daher nicht unbedingt eine Auswirkung auf die aktuelle Leistung.

**WICHTIG!** Den Skalierungsfaktor für dieses Register beachten!

Weitere Informationen unter:

<http://sunspec.org/wp-content/uploads/2015/06/SunSpec-Information-Models-12041.pdf>

**WMaxLimPct\_WinTms (7), WMaxLimPct\_RvrTms (8)**

Mit diesen Registern kann das Verhalten des Wechselrichters für diese Betriebsart zeitlich gesteuert werden. => siehe Abschnitt "Zeitverhalten der unterstützten Betriebsarten".

Als Standard ist für alle Register 0 vorgegeben.

### ***WMaxLim\_Ena (10)***

Zum Starten und Beenden diese Betriebsart

- Wert 1 in das Register *WMaxLim\_Ena* schreiben = Betriebsart starten
- Wert 0 in das Register *WMaxLim\_Ena* schreiben = Betriebsart beenden



**HINWEIS!** Um bei einer aktiven Betriebsart Werte zu verändern (z. B. ein anderes Leistungslimit oder eine andere Rückkehrzeit einstellen), folgendermaßen vorgehen:

- neuen Wert in das entsprechende Register schreiben
- die Betriebsart über Register *WMaxLim\_Ena* erneut starten

---

**Beispiel:  
Leistungsreduktion einstellen**

- [1]** Wert für die Reduktion der Ausgangsleistung in Register *WMaxLimPct* schreiben (z. B. 30 für 30%)
- [2]** Optional die Start- und Rückkehrzeit über Register *WMaxLimPct\_WinTms* und *WMaxLimPct\_RvrtTms* einstellen
- [3]** Durch Schreiben von 1 in Register *WMaxLim\_Ena* die Betriebsart starten

**WICHTIG!** Den Skalierungsfaktor für dieses Register beachten!

Weitere Informationen unter:

<http://sunspec.org/wp-content/uploads/2015/06/SunSpec-Information-Models-12041.pdf>

---

**Beispiel:  
Ändern der Rückkehrzeit bei aktiver  
Leistungsreduktion**

Leistungsreduktion ursprünglich mit *WMaxLimPct\_RvrtTms* = 0 gestartet, das heißt die Betriebsart muss manuell beendet werden.

- [1]** *WMaxLimPct\_RvrtTms* auf z.B. 30 setzen
- [2]** Durch Schreiben von 1 in Register *WMaxLim\_Ena* Änderung übernehmen
  - Betriebsart wird nach 30 Sekunden selbständig beendet.

---

**Auswirkungen  
der Blindleis-  
tungs-Vorgaben  
auf die Wirkleis-  
tung**

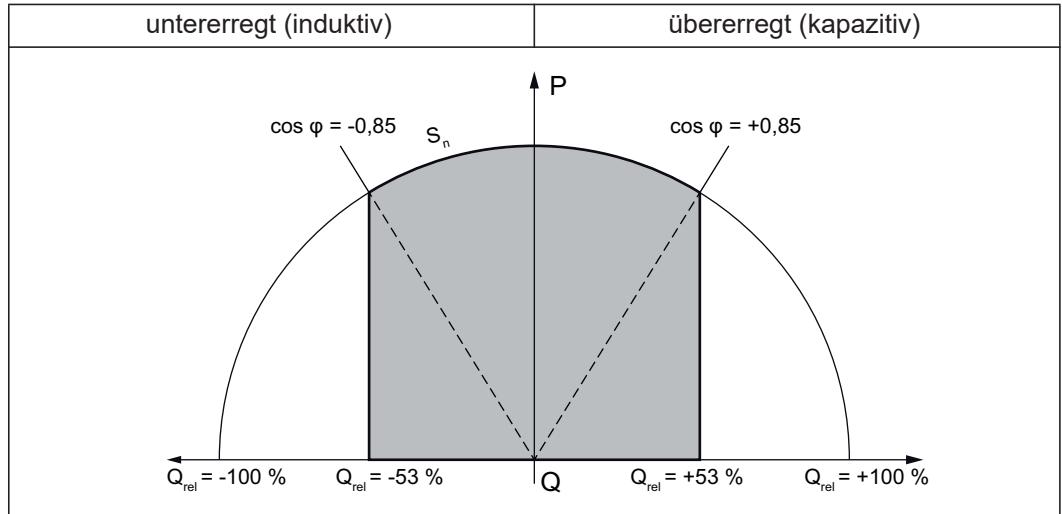
Der Blindleistungs-Betrieb wird grundsätzlich durch den maximalen Ausgangsstrom (die maximale Scheinleistung) sowie durch die operative Blindleistungs-Grenze des Wechselrichters begrenzt:

- Fronius IG Plus, CL cos phi = 0,85, VArrel = 100 %
- Fronius Galvo cos phi = 0,85, VArrel = 53 %
- Fronius Symo cos phi = 0,7, VArrel = 71 %



**HINWEIS!** Aufgrund der aktuellen technischen Gegebenheiten kann per Modbus nur ein cos phi bis zu maximal  $\pm 0,85$  vorgegeben werden. VAr<sub>rel</sub> Vorgaben können unter Umständen aber einen niedrigeren Wert erzwingen.

Die folgende Abbildung zeigt den möglichen Arbeitsbereich des Wechselrichters. Alle durch Wirkleistung P und Blindleistung Q definierten gültigen Arbeitspunkte sind innerhalb des grauen Bereiches.



*Blindleistung und Power Factor*

**Legende:**

$W$	Leistung	$VAr_{max}$	Nenn-Blindleistung
$W_{max}$	Nennleistung	$VAr_{rel}$	relative Blindleistung ( $VAr/VAr_{max}$ )
$VAr$	Blindleistung		

---

<b>Konstanter Power Factor</b>	<b><i>OutPFSet (11)</i> bis <i>OutPFSet_Ena (15)</i></b> Über diese Register kann beim Wechselrichter ein konstanter Power Factor vorgegeben werden.
--------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------

***OutPFSet (11)***

- In Register *OutPFSet* können positive und negative Werte für den Power Factor eingegeben werden
- Die Werte sind mit Faktor 1000 zu skalieren  
z.B. Power Factor 0,95 = Register-Wert 950
- Die minimal möglichen Werte hängen vom Wechselrichter-Typ ab und können dem Nameplate Model entnommen werden



**HINWEIS!** Der Wert für den Power Factor muss mit dem korrekten Vorzeichen eingegeben werden => siehe Abschnitt "Vorzeichenkonvention für den Power Factor"

- positiv für untererregt
- negativ für übererregt

***OutPFSet\_WinTms (12), OutPFSet\_RvrtTms (13)***

Mit diesen Registern kann das Verhalten des Wechselrichters für diese Betriebsart zeitlich gesteuert werden. => siehe Abschnitt "Zeitverhalten der unterstützten Betriebsarten". Als Standard ist für alle Register 0 vorgegeben.

***OutPFSet\_Ena (15)***

Zum Starten und Beenden dieser Betriebsart

- Wert 1 in das Register *OutPFSet\_Ena* schreiben = Betriebsart starten
- Wert 0 in das Register *OutPFSet\_Ena* schreiben = Betriebsart beenden



**HINWEIS!** Um bei einer aktiven Betriebsart Werte zu verändern (z. B. einen anderen Power Factor oder eine andere Rückkehrzeit einstellen), folgendermaßen vorgehen:

- neuen Wert in das entsprechende Register schreiben
- die Betriebsart über Register *OutPFSet\_Ena* erneut starten

---

**Beispiel:  
Konstanten Power Factor vorgeben**

- [1]** Wert für den Power Factor in Register *OutPFSet* schreiben  
(z. B. 950 für 0,95)
- [2]** Optional die Start- und Rückkehrzeit über Register *OutPFSet\_WinTms* und *OutPFSet\_RvrtTms* einstellen
- [3]** Durch Schreiben von 1 in Register *OutPFSet\_Ena* die Betriebsart starten

## Konstante relative Blindleistung

### **VArMaxPct (17) bis VArPct\_Ena (23)**

Über diese Register kann am Wechselrichter ein konstanter Wert für die Blindleistung eingestellt werden, die der Wechselrichter liefern soll.

#### **VArMaxPct (17)**

- zum Einstellen eines Wertes für die konstante Blindleistung
- Die minimal und maximal möglichen Werte hängen vom Wechselrichter-Typ ab



**HINWEIS!** Im praktischen Betrieb wird die tatsächlich verfügbare Blindleistung durch die Betriebsgrenzen des Wechselrichters vorgegeben. Deshalb kann die Blindleistungs-Vorgabe nur dann erreicht werden, wenn ausreichend Wirkleistung eingespeist wird.

Wird zu wenig Wirkleistung eingespeist, arbeitet der Wechselrichter an der Betriebsgrenze.

#### **VArPct\_WinTms (19), VArPct\_RvrtTms (20)**

Mit diesen Registern kann das Verhalten des Wechselrichters für diese Betriebsart zeitlich gesteuert werden. => siehe Abschnitt "Zeitverhalten der unterstützten Betriebsarten".

Als Standard ist für alle Register 0 vorgegeben.

#### **VArPct\_Mod (22)**

- dieses Register kann nicht verändert werden
- liefert die (derzeit) unterstützte Betriebsart zurück  
Blindleistung in Prozent der maximal möglichen Blindleistung

#### **VArPct\_Ena (23)**

Zum Starten und Beenden dieser Betriebsart

- Wert 1 in das Register VArPct\_Ena schreiben = Betriebsart starten
- Wert 0 in das Register VArPct\_Ena schreiben = Betriebsart beenden



**HINWEIS!** Um bei einer aktiven Betriebsart Werte zu verändern (z. B. eine andere Blindleistung oder eine andere Rückkehrzeit einstellen), folgendermaßen vorgehen:

- neuen Wert in das entsprechende Register schreiben
- die Betriebsart über Register VArPct\_Ena erneut starten

---

## Beispiel: Konstante Blindleistung vorgeben

- 1** Wert für die relative Blindleistung in Register VArMaxPct schreiben  
(z. B. 80 für 80%)
- 2** Optional die Start- und Rückkehrzeit über Register VArPct\_WinTms und VArPct\_RvrtTms einstellen
- 3** Durch Schreiben von 1 in Register VArPct\_Ena den Betriebsart starten

# Basic Storage Control Model (IC124)

<b>Allgemeines</b>	Dieses Model ist nur für Fronius Hybrid Wechselrichter verfügbar.  Mit dem Basic Storage Control Model können folgende Einstellungen am Wechselrichter vorgenommen werden: <ul style="list-style-type: none"><li>- Vorgabe eines Leistungsfensters, in dem sich die Lade-/Entladeleistung vom Energiespeicher bewegen soll.</li><li>- Vorgabe eines minimalen Ladestandes, den der Energiespeicher nicht unterschreiten soll</li><li>- Ladung des Energiespeichers vom Netz erlauben/verbieten</li></ul>
	 <b>HINWEIS!</b> Alle Vorgaben verstehen sich als Empfehlungen! Der Wechselrichter kann von den Vorgaben abweichen, wenn dies aus Gründen der Betriebssicherheit erforderlich ist.
<b>Bereitgestellte Informationen</b>	Das Basic Storage Control Model stellt folgende Informationen lesend zu Verfügung: <ul style="list-style-type: none"><li>WChaMax<ul style="list-style-type: none"><li>- Wenn ein Energiespeicher verfügbar ist liefert dieses Register den Bezugswert für die Register OutWRt und InWRt zurück. <math>WChaMax := \max(\text{MaxChaRte}, \text{MaxDisChaRte})</math></li><li>- Wenn kein Energiespeicher verfügbar ist liefert das Register den Wert 0 zurück.</li></ul></li><li>ChaState<ul style="list-style-type: none"><li>- Ladestand des Energiespeicher in %: <math>\text{Estimated\_Capacity\_Remaining [Wh]} / \text{Estimated\_Capacity\_Maximum [Wh]}</math></li></ul></li><li>ChaSt<ul style="list-style-type: none"><li>Betriebsstatus des Energiespeichers<ul style="list-style-type: none"><li>- OFF: Energiespeicher ist nicht verfügbar</li><li>- EMPTY: Energiespeicher ist derzeit vollständig entladen</li><li>- DISCHARGING: Energiespeicher wird derzeit entladen</li><li>- CHARGING: Energiespeicher wird derzeit geladen</li><li>- FULL: Energiespeicher ist derzeit vollständig geladen</li><li>- HOLDING: Energiespeicher wird derzeit weder geladen noch entladen</li></ul></li></ul></li></ul>
<b>Leistungsfenster-Vorgaben</b>	Für die folgenden Beispiele wird WchaMax = 3300 W angenommen.  Für resultierende Leistungsfenster gilt: <ul style="list-style-type: none"><li>- negative Leistungswerte entsprechen einer Ladung des Energiespeichers</li><li>- positive Werte entsprechen einer Entladung des Energiespeichers</li></ul>
	 <b>HINWEIS!</b> Die Werte in den folgenden Beispielen müssen nach dem Lesen und vor dem Schreiben entsprechend ihren Skalierungsfaktoren in den angegebenen Skalierungsregistern skaliert werden.
	<b>Beispiel 1: Nur Laden des Energiespeichers erlauben</b>
	Dieses Verhalten kann durch Limitierung der maximalen Entladeleistung auf 0% erreicht werden => resultiert in Fenster [-3300 W, 0 W] <ul style="list-style-type: none"><li>- OutWRt = 0% (setze Entladelimit auf 0% von WchaMax)</li><li>- StorCtl_Mod = 1 (schaltet Entladegrenzwert aktiv, Bit-Muster: 01)</li><li>- InWRt ist in diesem Fall nicht relevant</li></ul>

## **Beispiel 2: Nur Entladen des Energiespeichers erlauben**

Dieses Verhalten kann durch Limitierung der maximalen Ladeleistung auf 0% erreicht werden => resultiert in Fenster [0 W, 3300 W]

- InWRte = 0% (setze Ladelimit auf 0% von WchaMax)
- StorCtl\_Mod = 2 (Bit 1 schaltet Ladegrenzwert aktiv, Bit-Muster: 10)
- OutWRte ist in diesem Fall nicht relevant

## **Beispiel 3: Weder Laden noch Entladen erlauben**

Dieses Verhalten kann durch Limitierung der maximalen Ladeleistung auf 0% und Limitierung der maximalen Entladeleistung auf 0% erreicht werden

=> resultiert in Fenster [0 W, 0 W]

- InWRte = 0% (setze Ladelimit auf 0% von WchaMax)
- OutWRte = 0% (setze Entladelimit auf 0% von WchaMax)
- StorCtl\_Mod = 3 (schalte beide Grenzwerte aktiv, Bit-Muster: 11)

## **Beispiel 4: Laden und Entladen mit maximal 50% der nominalen Leistung**

Dieses Verhalten kann durch Limitierung der maximalen Ladeleistung auf 50% und Limitierung der maximalen Entladeleistung auf 50% erreicht werden

=> resultiert in Fenster [-1650 W, 1650 W]

- InWRte = 50% (setze Ladelimit auf 50% von WchaMax)
- OutWRte = 50% (setze Entladelimit auf 50% von WchaMax)
- StorCtl\_Mod = 3 (schalte beide Grenzwerte aktiv, Bit-Muster: 11)

## **Beispiel 5: Laden im Bereich von 50% bis 75% der nominalen Leistung**

Dieses Verhalten kann durch Limitierung der maximalen Ladeleistung auf 75% und Limitierung der maximalen Entladeleistung auf -50% erreicht werden

=> resultiert in Fenster [1650 W, 2475 W]

- InWRte = 75% (setze Ladelimit auf 75% von WchaMax)
- OutWRte = -50% (setze Entladelimit auf -50% von WchaMax)
- StorCtl\_Mod = 3 (schalte beide Grenzwerte aktiv, Bit-Muster: 11)

## **Beispiel 6: Entladen mit 50% der nominalen Leistung**

Dieses Verhalten kann durch Limitierung der maximalen Ladeleistung auf -50% und Limitierung der maximalen Entladeleistung auf 50% erreicht werden

=> resultiert in Fenster [-1650 W, -1650 W]

- InWRte = -50% (setze Ladelimit auf -50% von WchaMax)
- OutWRte = 50% (setze Entladelimit auf 50% von WchaMax)
- StorCtl\_Mod = 3 (schalte beide Grenzwerte aktiv, Bit-Muster: 11)

## **Beispiel 7: Laden mit 50% bis 75% der nominalen Leistung**

Dieses Verhalten kann durch Limitierung der maximalen Ladeleistung auf 75% und Limitierung der maximalen Entladeleistung auf -50% erreicht werden

=> resultiert in Fenster [1650 W, 247 5W]

- InWRte = 75% (setze Ladelimit auf 75% von WchaMax)
- OutWRte = -50% (setze Entladelimit auf -50% von WchaMax)
- StorCtl\_Mod = 3 (schalte beide Grenzwerte aktiv, Bit-Muster: 11)

## **Beispiel 8: Laden mit 50% bis 100% der nominalen Leistung**

Dieses Verhalten kann durch Limitierung der maximalen Entladeleistung auf -50% erreicht werden => resultiert in Fenster [1650 W, 3300 W]

- OutWRte = -50% (setze Entladelimit auf -50% von WchaMax)
- StorCtl\_Mod = 1 (schaltet Entladegrenzwert aktiv, Bit-Muster: 01)
- InWRte ist in diesem Fall nicht relevant

---

<b>Vorgabe des minimalen Ladestandes</b>	Durch Setzen von Register MinRsvPct kann ein minimal zu erhaltender Ladezustand des Speichers festgelegt werden. Beispielsweise kann durch Setzen von MinRsvPct=20% eine Reserve von 20% des Ladezustandes reserviert werden, die der Speicher nicht unterschreiten soll.
<b>Laden des Energiespeichers vom Netz</b>	Mit dem Register ChaGriSet kann es dem Wechselrichter erlaubt oder verboten werden, den Speicher vom Netz zu laden.

---

<b>Basic Storage Controls Register</b>	Startadresse: - bei Einstellung „float“: <b>40313</b> - bei Einstellung „int+SF“: <b>40303</b>
----------------------------------------	------------------------------------------------------------------------------------------------------

Start Offset	End Offset	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
1	1	1	R	0x03	ID	uint16			A well-known value 124 Uniquely identifies this as a SunSpec Basic Storage Controls Model	124
2	2	1	R	0x03	L	uint16	Registers		Length of Basic Storage Controls	24
3	3	1	R	0x03	WchaMax	uint16	W	WCha-Max_SF	Setpoint for maximum charge.  *	

\* Additional Fronius description:

Reference Value for maximum Charge and Discharge. Multiply this value by InWRte to define maximum charging and OutWRte to define maximum discharging. Every rate between these two limits is allowed. Note that InWRte and OutWRte can be negative to define ranges for charging and discharging only.

4	4	1	R	0x03	WchaGra	uint16	% WCha-Max/sec	WCha-DisChaGra_SF	Setpoint for maximum charging rate. Default is MaxChaRte	100
5	5	1	R	0x03	WdisChaGra	unit16	% WCha-Max/sec	WCha-DisChaGra_SF	Setpoint for maximum discharge rate. Default is MaxDisChaRte.	100
6	6	1	RW	0x03 0x06 0x10	StorCtl_Mod	bitfield16	VA	WMaxLimPct_SF	Activate storage control mode. Bitfield value.  **	Bit 0: charge  Bit 1: discharge

\*\* Additional Fronius description:

Set charge bit to enable charge limit, set discharge bit to enable discharge limit, set both bits to enable both limits.

Start Offset	End Offset	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
7	7	1	R	0x03	VAChaMax	uint16	VA	VACha-Max_SF	Setpoint for maximum charging VA.	Not supported
8	8	1	RW	0x03 0x06 0x10	MinRsvPct	uint16	% WCha-Max	Min-RsvPct_SF	Setpoint for minimum reserve for storage as a percentage of the nominal maximum storage.	
9	9	1	R	0x03	ChaState	uint16	% AhrRtg	ChaState_SF	Currently available energy as a percent of the capacity rating	
10	10	1	R	0x03	StorAval	uint16	Ah	StorAval_SF	State of charge (ChaState) minus storage reserve (MinRsvPct) times capacity rating (AhrRtg).	Not supported
11	11	1	R	0x03	InBatV	uint16	V	InBatV_SF	Internal battery voltage.	Not supported
12	12	1	R	0x03	ChaSt	enum16			Charge status of storage device. Enumerated value.	1: OFF 2: EMPTY 3: DISCHARGING 4: CHARGING 5: FULL 6: HOLDING 7: TESTING
13	13	1	RW	0x03 0x06 0x10	OutWRte	int16	% WCha-Max	InOutWRte_SF	Percent of max discharge rate.  ***	

\*\*\* Additional Fronius description:

Defines the maximum discharge rate (discharge limit). Default is 100%

14	14	1	RW	0x03 0x06 0x10	InWRte	int16	% WCha-Max	InOutWRte_SF	Percent of max charging rate.  ****	
----	----	---	----	----------------------	--------	-------	------------	--------------	-------------------------------------------	--

\*\*\*\* Additional Fronius description:

Defines the maximum charge rate (charge limit). Default is 100%

15	15	1	R	0x03	InOutWRte_WinTms	uint16	Secs		Time window for charge/discharge rate change.	Not supported
16	16	1	R	0x03	InOutWRte_RvrtTms	uint16	Secs		Timeout period for charge/discharge rate.	Not supported
17	17	1	R	0x03	InOutWRte_RmpTms	uint16	Secs		Ramp time for moving from current setpoint to new setpoint.	Not supported

<b>Start Offset</b>	<b>End Offset</b>	<b>Size</b>	<b>R/W</b>	<b>Function codes</b>	<b>Name</b>	<b>Type</b>	<b>Units</b>	<b>Scale factor</b>	<b>Description</b>	<b>Range of values</b>
18	18	1	RW	0x03 0x06 0x10	ChaGriSet	enum16			Setpoint to enable/disable charging from grid  PV (Charging from grid disabled)  GRID (Charging from grid enabled)	
19	19	1	R	0x03	WchaMax_SF	sunssf			Scale factor for maximum charge.	0
20	20	1	R	0x03	Wcha-DisChaGra_SF	sunssf			Scale factor for maximum charge and discharge rate	0
21	21	1	R	0x03	VAChaMax_SF	sunssf			Scale factor for maximum charging VA	Not supported
22	22	1	R	0x03	MinRsvPct_SF	sunssf			Scale factor for minimum reserve percentage.	-2
23	23	1	R	0x03 0x06 0x10	ChaState_SF	sunssf			Scale factor for available energy percent.	-2
24	24	1	R	0x03	StorAval_SF	sunssf			Scale factor for state of charge.	-2
25	25	1	R	0x03	InBatV_SF	sunssf			Scale factor for battery voltage.	-2
26	26	1	R	0x03	InOutWRte_SF	sunssf			Scale factor for percent charge/discharge rate.	-2

# Multiple MPPT Inverter Extension Model (I160)

DE

Allgemeines	Das Multiple MPPT Inverter Extension Model beinhaltet die Werte von bis zu zwei DC Eingängen des Wechselrichters.
-------------	-------------------------------------------------------------------------------------------------------------------

Verfügt der Wechselrichter über zwei DC Eingänge, so werden Strom, Spannung, Leistung, Energie und Statusmeldungen der einzelnen Eingänge hier aufgelistet. Im Inverter Model (101 -103 oder 111 - 113) wird in diesem Fall nur die gesamte DC Leistung beider Eingänge ausgegeben. DC Strom und DC Spannung werden als "not implemented" angezeigt.

Sollte der Wechselrichter nur über einen DC Eingang verfügen, werden alle Werte des zweiten Strings auf "not implemented" gesetzt (ab Register 2\_DCA). Die Bezeichnung des zweiten Eingangs (Register 2\_IDStr) lautet in diesem Fall "Not supported". Die Werte des ersten (und einzigen) Eingangs werden normal angezeigt.

Multiple MPPT Inverter Extension Register	Startadresse: - bei Einstellung „float“: <b>40263</b> - bei Einstellung „int+SF“: <b>40253</b>
-------------------------------------------	------------------------------------------------------------------------------------------------------

Start Offset	End Offset	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
1	1	1	R	0x03	ID	unit16			A well-known value 160. Uniquely identifies this as a SunSpec Multiple MPPT Inverter Extension Model Mode	160
2	2	1	R	0x03	L	uint16			Length of Multiple MPPT Inverter Extension Model	48
3	3	1	R	0x03	DCA_SF	sunssf			Current Scale Factor	
4	4	1	R	0x03	DCV_SF	sunssf			Voltage Scale Factor	
5	5	1	R	0x03	DCW_SF	sunssf			Power Scale Factor	
6	6	1	R	0x03	DCWH_SF	sunssf			Energy Scale Factor	
7	8	2	R	0x03	Evt	bitfield32			Global Events	
9	9	1	R	0x03	N	uint16			Number of Modules	2
10	10	1	R	0x03	TmsPer	uint16			Timestamp Period	Not supported
11	11	1	R	0x03	1_ID	uint16			Input ID	1
12	19	8	R	0x03	1_IDStr	string8			Input ID String	"String 1" <sup>1)</sup>
20	20	1	R	0x03	1_DCA	uint16	A	DCA_SF	DC Current	
21	21	1	R	0x03	1_DCV	uint16	V	DC_V_SF	DC Voltage	
22	22	1	R	0x03	1_DCW	uint16	W	DCW_SF	DC Power	

Start Offset	End Offset	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
23	24	2	R	0x03	1_DCWH	acc32	Wh	DCWH_SF	Lifetime Energy	
25	26	2	R	0x03	1_Tms	uint32	Secs		Timestamp	
27	27	1	R	0x03	1_Tmp	int16	C		Temperature	
28	28	1	R	0x03	1_DCSt	enum16			Operating State	
Name					Value	Description				
I_STATUS_OFF					1	Off				
I_STATUS_SLEEPING					2	In operation (no feed-in)				
I_STATUS_STARTING					3	Run-up phase				
I_STATUS_MPPT					4	Normal operation				
I_STATUS_THROTTLED					5	Power Reduction				
I_STATUS_SHUTTING_DOWN					6	Switch-off phase				
I_STATUS_FAULT					7	Error exists				
I_STATUS_STANDBY					8	Standby				
29	30	2	R	0x03	1_DCEvt	bitfield32			Module Events	Not supported
31	31	1	R	0x03	2_ID	uint16			Input ID	2
32	39	8	R	0x03	2_IDStr	string8			Input ID Sting	"String 2" <sup>2)</sup> or "Not supported"
40	40	1	R	0x03	2_DCA	uint16	A	DCA_SF	DC Current	3)
41	41	1	R	0x03	2_DCV	uint16	V	DC_V_SF	DC Voltage	3)
42	42	1	R	0x03	2_DCW	uint16	W	DCW_SF	DC Power	3)
43	44	2	R	0x03	2_DCWH	acc32	Wh	DCWH_SF	Lifetime Energy	
45	46	2	R	0x03	2_Tms	uint32	Secs		Timestamp	
47	47	1	R	0x03	2_Tmp	int16	C		Temperature	
48	48	1	R	0x03	2_DCSt	enum16			Operating State	
Name					Value	Description				
I_STATUS_OFF					1	Off				
I_STATUS_SLEEPING					2	In operation (no feed-in)				
I_STATUS_STARTING					3	Run-up phase				
I_STATUS_MPPT					4	Normal operation				
I_STATUS_THROTTLED					5	Power Reduction				
I_STATUS_SHUTTING_DOWN					6	Switch-off phase				
I_STATUS_FAULT					7	Error exists				
I_STATUS_STANDBY					8	Standby				
49	50	2	R	0x03	2_DCEvt	bitfield32			Module Events	Not supported

Start Offset	End Offset	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
--------------	------------	------	-----	----------------	------	------	-------	--------------	-------------	-----------------

- 1) String 1 (MPPT1) = Register 1\_DCA, 1\_DCV und 1\_DCW
- 2) String 2 (MPPT2) = Register 2\_DCA, 2\_DCV und 2\_DCW
- 3) Summenwerte

DCW = Summe der DC-Leistungen

Bei Hybrid-Systemen:

String 1 = PV-Eingang

String 2 = Storage

Bei Entladung des Storage: DCW = 1\_DCW + 2\_DCW

Bei Ladung des Storage: DCW = 1\_DCW - 2\_DCW

### Beispiele

- a) PV-Eingang: 2000 W Produktion ==> 1\_DCW = 2000 W  
 Storage: 1000 W Entladen ==> 2\_DCW = 1000 W

$$\text{DCW} = 1_{\text{DCW}} + 2_{\text{DCW}} = 1000 \text{ W} + 2000 \text{ W} = 3000 \text{ W}$$

- b) PV-Eingang: 2000 W Produktion ==> 1\_DCW = 2000 W  
 Storage: - 1000 W Laden ==> 2\_DCW = 1000 W  
 (nur der Absolutwert kann über dieses Register angezeigt werden)

$$\text{DCW} = 1_{\text{DCW}} + 2_{\text{DCW}} = 2000 \text{ W} + (-1000 \text{ W}) = 1000 \text{ W}$$

# String Combiner Model (SC403)

## String Combiner Register

Start	End	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
40070	40070	1	R	0x03	ID	uint16			A well-known value 403. Uniquely identifies this as a SunSpec String Combiner (Current) Model	403
40071	40071	1	R	0x03	L	uint16	Registers		Length of string combiner model block	56
40072	40072	1	R	0x03	DCA_SF	sunssf			Current scale factor	
40073	40073	1	R	0x03	DCAhr_SF	sunssf			Amp-hour scale factor	
40074	40074	1	R	0x03	DCV_SF	sunssf			Voltage scale factor	Not supported
40075	40075	1	R	0x03	DCAMax	uint16	A		Maximum DC Current Rating	
40076	40076	1	R	0x03	N	uint16			Number of Inputs	Max. 5
40077	40078	2	R	0x03	Evt	bitfield32			Bitmask value. Events	
40079	40080	2	R	0x03	EvtVnd	bitfield32			Bitmask value. Vendor defined events	
40081	40081	1	R	0x03	DCA	int16	A		Total measured current	
40082	40083	2	R	0x03	DCAhr	acc32	Ah		Total metered Amp-hours	
40084	40084	1	R	0x03	DCV	int16	V		Output Voltage	Not supported
40085	40085	1	R	0x03	Tmp	int16	C		Internal operating temperature	Not supported
40086	40086	1	R	0x03	InDCA_SF	sunssf			Current scale factor for inputs	
40087	40087	1	R	0x03	InDCAhr_SF	sunssf			Amp-hour scale factor for inputs	
40088	40088	1	R	0x03	InID	uint16			Uniquely identifies this input set	
40089	40090	2	R	0x03	InEvt	bitfield32			String Input Event Flags	
40091	40092	2	R	0x03	InEvtVnd	bitfield32			String Input Vendor Event Flags	
40093	40093	1	R	0x03	InDCA	int16	A		String Input Current	
40094	40095	2	R	0x03	InDCAhr	acc32	Ah		String Input Amp-Hours	
40096	40096	1	R	0x03	InID	uint16			Uniquely identifies this input set	
40097	40098	2	R	0x03	InEvt	bitfield32			String Input Event Flags	
40099	40100	2	R	0x03	InEvtVnd	bitfield32			String Input Vendor Event Flags	

<b>Start</b>	<b>End</b>	<b>Size</b>	<b>R/W</b>	<b>Function codes</b>	<b>Name</b>	<b>Type</b>	<b>Units</b>	<b>Scale factor</b>	<b>Description</b>	<b>Range of values</b>
40101	40101	1	R	0x03	InDCA	int16	A		String Input Current	
40102	40103	2	R	0x03	InDCAhr	acc32	Ah		String Input Amp-Hours	
40104	40104	1	R	0x03	InID	uint16			Uniquely identifies this input set	
40105	40106	2	R	0x03	InEvt	bitfield32			String Input Event Flags	
40107	40108	2	R	0x03	InEvtVnd	bitfield32			String Input Vendor Event Flags	
40109	40109	1	R	0x03	InDCA	int16	A		String Input Current	
40110	40111	2	R	0x03	InDCAhr	acc32	Ah		String Input Amp-Hours	
40112	40112	1	R	0x03	InID	uint16			Uniquely identifies this input set	
40113	40114	2	R	0x03	InEvt	bitfield32			String Input Event Flags	
40115	40116	2	R	0x03	InEvtVnd	bitfield32			String Input Vendor Event Flags	
40117	40117	1	R	0x03	InDCA	int16	A		String Input Current	
40118	40119	2	R	0x03	InDCAhr	acc32	Ah		String Input Amp-Hours	
40120	40120	1	R	0x03	InID	uint16			Uniquely identifies this input set	
40121	40122	2	R	0x03	InEvt	bitfield32			String Input Event Flags	
40123	40124	2	R	0x03	InEvtVnd	bitfield32			String Input Vendor Event Flags	
40125	40125	1	R	0x03	InDCA	int16	A		String Input Current	
40126	40127	2	R	0x03	InDCAhr	acc32	Ah		String Input Amp-Hours	

# Meter Model

## Meter Model Register

Die Daten eines per Modbus RTU mit dem Fronius Datamanager verbundenen Energiezählers können per Modbus TCP über die entsprechenden SunSpec Models ausgelesen werden.

Ähnlich wie bei den Inverter Models gibt es auch hier zwei verschiedene SunSpec Models:

- das Meter Model mit Gleitkommadarstellung (Einstellung „float“; M211, M212 oder M213)
- das Meter Model mit ganzen Zahlen und Skalierungsfaktoren (Einstellung „int+SF“; M201, M202 oder M203)



**HINWEIS!** Die Registeranzahl der beiden Model-Typen ist unterschiedlich!

Die Modbus Geräte-ID des Energiezählers ist 240.

### Floating Point Darstellung – „float“

Start	End	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
40070	40070	1	R	0x03	ID	uint16			Uniquely identifies this as a SunSpec Meter Modbus Map; 211: single phase, 212: split phase, 213: three phase	211, 212, 213
40071	40071	1	R	0x03	L	uint16	Registers		Length of meter model block	124
40072	40073	2	R	0x03	A	float32	A		AC Total Current value	
40074	40075	2	R	0x03	AphA	float32	A		AC Phase-A Current value	
40076	40077	2	R	0x03	AphB	float32	A		AC Phase-B Current value	
40078	40079	2	R	0x03	AphC	float32	A		AC Phase-C Current value	
40080	40081	2	R	0x03	PhV	float32	V		AC Voltage Average Phase-to-neutral value	
40082	40083	2	R	0x03	PhVphA	float32	V		AC Voltage Phase-A-to-neutral value	
40084	40085	2	R	0x03	PhVphB	float32	V		AC Voltage Phase-B-to-neutral value	
40086	40087	2	R	0x03	PhVphC	float32	V		AC Voltage Phase-C-to-neutral value	
40088	40089	2	R	0x03	PPV	float32	V		AC Voltage Average Phase-to-phase value	
40090	40091	2	R	0x03	PPVphAB	float32	V		AC Voltage Phase-AB value	
40092	40093	2	R	0x03	PPVphBC	float32	V		AC Voltage Phase-BC value	

**Floating Point Darstellung – „float“**

<b>Start</b>	<b>End</b>	<b>Size</b>	<b>R/W</b>	<b>Function codes</b>	<b>Name</b>	<b>Type</b>	<b>Units</b>	<b>Scale factor</b>	<b>Description</b>	<b>Range of values</b>
40094	40095	2	R	0x03	PPVphCA	float32	V		AC Voltage Phase-CA value	
40096	40097	2	R	0x03	Hz	float32	Hz		AC Frequency value	
40098	40099	2	R	0x03	W	float32	W		AC Power value	
40100	40101	2	R	0x03	WphA	float32	W		AC Power Phase A value	
40102	40103	2	R	0x03	WphB	float32	W		AC Power Phase B value	
40104	40105	2	R	0x03	WphC	float32	W		AC Power Phase C value	
40106	40107	2	R	0x03	VA	float32	VA		AC Apparent Power value	
40108	40109	2	R	0x03	VApH A	float32	VA		AC Apparent Power Phase A value	
40110	40111	2	R	0x03	VApH B	float32	VA		AC Apparent Power Phase B value	
40112	40113	2	R	0x03	VApH C	float32	VA		AC Apparent Power Phase C value	
40114	40115	2	R	0x03	VAR	float32	VAr		AC Reactive Power value	
40116	40117	2	R	0x03	VARphA	float32	VAr		AC Reactive Power Phase A value	
40118	40119	2	R	0x03	VARphB	float32	VAr		AC Reactive Power Phase B value	
40120	40121	2	R	0x03	VARphC	float32	VAr		AC Reactive Power Phase C value	
40122	40123	2	R	0x03	PF	float32	PF		Power Factor value	
40124	40125	2	R	0x03	PFphA	float32	PF		Power Factor Phase A value	
40126	40127	2	R	0x03	PFphB	float32	PF		Power Factor Phase B value	
40128	40129	2	R	0x03	PFphC	float32	PF		Power Factor Phase C value	
40130	40131	2	R	0x03	TotWhExp	float32	Wh		Total Watt-hours Exported	
40132	40133	2	R	0x03	TotWhExpPhA	float32	Wh		Total Watt-hours Exported phase A	
40134	40135	2	R	0x03	TotWhExpPhB	float32	Wh		Total Watt-hours Exported phase B	
40136	40137	2	R	0x03	TotWhExpPhC	float32	Wh		Total Watt-hours Exported phase C	
40138	40139	2	R	0x03	TotWhImp	float32	Wh		Total Watt-hours Imported	
40140	40141	2	R	0x03	TotWhImpPhA	float32	Wh		Total Watt-hours Imported phase A	

### Floating Point Darstellung – „float“

Start	End	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
40142	40143	2	R	0x03	TotWhImpPhB	float32	Wh		Total Watt-hours Imported phase B	
40144	40145	2	R	0x03	TotWhImpPhC	float32	Wh		Total Watt-hours Imported phase C	
40146	40147	2	R	0x03	TotVAhExp	float32	VAh		Total VA-hours Exported	
40148	40149	2	R	0x03	TotVAhExpPhA	float32	VAh		Total VA-hours Exported phase A	
40150	40151	2	R	0x03	TotVAhExpPhB	float32	VAh		Total VA-hours Exported phase B	
40152	40153	2	R	0x03	TotVAhExpPhC	float32	VAh		Total VA-hours Exported phase C	
40154	40155	2	R	0x03	TotVAhImp	float32	VAh		Total VA-hours Imported	
40156	40157	2	R	0x03	TotVAhImpPhA	float32	VAh		Total VA-hours Imported phase A	
40158	40159	2	R	0x03	TotVAhImpPhB	float32	VAh		Total VA-hours Imported phase B	
40160	40161	2	R	0x03	TotVAhImpPhC	float32	VAh		Total VA-hours Imported phase C	
40162	40163	2	R	0x03	TotVArhImpQ1	float32	VArh		Total VAR-hours Imported Q1	
40164	40165	2	R	0x03	TotVArhImpQ1phA	float32	VArh		Total VAR-hours Imported Q1 phase A	
40166	40167	2	R	0x03	TotVArhImpQ1phB	float32	VArh		Total VAR-hours Imported Q1 phase B	
40168	40169	2	R	0x03	TotVArhImpQ1phC	float32	VArh		Total VAR-hours Imported Q1 phase C	
40170	40171	2	R	0x03	TotVArhImpQ2	float32	VArh		Total VAr-hours Imported Q2	
40172	40173	2	R	0x03	TotVArhImpQ2phA	float32	VArh		Total VAR-hours Imported Q2 phase A	
40174	40175	2	R	0x03	TotVArhImpQ2phB	float32	VArh		Total VAR-hours Imported Q2 phase B	
40176	40177	2	R	0x03	TotVArhImpQ2phC	float32	VArh		Total VAR-hours Imported Q2 phase C	
40178	40179	2	R	0x03	TotVArhExpQ3	float32	VArh		Total VAr-hours Exported Q3	
40180	40181	2	R	0x03	TotVArhExpQ3phA	float32	VArh		Total VAR-hours Exported Q3 phase A	
40182	40183	2	R	0x03	TotVArhExpQ3phB	float32	VArh		Total VAR-hours Exported Q3 phase B	
40184	40185	2	R	0x03	TotVArhExpQ3phC	float32	VArh		Total VAR-hours Exported Q3 phase C	

### Floating Point Darstellung – „float“

Start	End	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
40186	40187	2	R	0x03	TotVArhExpQ4	float32	VArh		Total VAr-hours Exported Q4	
40188	40189	2	R	0x03	TotVArhExpQ4phA	float32	VArh		Total VAR-hours Exported Q4 phase A	
40190	40191	2	R	0x03	TotVArhExpQ4phB	float32	VArh		Total VAR-hours Exported Q4 phase B	
40192	40193	2	R	0x03	TotVArhExpQ4phC	float32	VArh		Total VAR-hours Exported Q4 phase C	
40194	40195	2	R	0x03	Evt	uint32	Bitfield		Events	

### Ganzzahldarstellung – „int+SF“

Start	End	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
40070	40070	1	R	0x03	ID	uint16			Uniquely identifies this as a SunSpec Meter Modbus Map 201: single phase 202: split phase 203: three phase	201, 202, 203
40071	40071	1	R	0x03	L	uint16	Registers		Length of meter model block	105
40072	40072	1	R	0x03	A	int16	A		AC Total Current value	
40073	40073	1	R	0x03	AphA	int16	A		AC Phase-A Current value	
40074	40074	1	R	0x03	AphB	int16	A		AC Phase-B Current value	
40075	40075	1	R	0x03	AphC	int16	A		AC Phase-C Current value	
40076	40076	1	R	0x03	A_SF	int16	sunssf		AC Current Scale factor	
40077	40077	1	R	0x03	PhV	int16	V		AC Voltage Average Phase-to-neutral value	
40078	40078	1	R	0x03	PhVphA	int16	V		AC Voltage Phase-A-to-neutral value	
40079	40079	1	R	0x03	PhVphB	int16	V		AC Voltage Phase-B-to-neutral value	
40080	40080	1	R	0x03	PhVphC	int16	V		AC Voltage Phase-C-to-neutral value	
40081	40081	1	R	0x03	PPV	int16	V		AC Voltage Average Phase-to-phase value	
40082	40082	1	R	0x03	PPVphAB	int16	V		AC Voltage Phase-AB value	

### Ganzzahldarstellung – „int+SF“

Start	End	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
40083	40083	1	R	0x03	PPVphBC	int16	V		AC Voltage Phase-BC value	
40084	40084	1	R	0x03	PPVphCA	int16	V		AC Voltage Phase-CA value	
40085	40085	1	R	0x03	V_SF	int16	sunssf		AC Phase Voltage Scale factor	
40086	40086	1	R	0x03	Hz	int16	Hz		AC Frequency value	
40087	40087	1	R	0x03	Hz_SF	int16	sunssf		AV Frequency Scale factor	
40088	40088	1	R	0x03	W	int16	W		AC Power value	
40089	40089	1	R	0x03	WphA	int16	W		AC Power Phase A value	
40090	40090	1	R	0x03	WphB	int16	W		AC Power Phase B value	
40091	40091	1	R	0x03	WphC	int16	W		AC Power Phase C value	
40092	40092	1	R	0x03	W_SF	int16	sunssf		AC Phase Power Scale factor	
40093	40093	1	R	0x03	VA	int16	VA		AC Apparent Power value	
40094	40094	1	R	0x03	VApH A	int16	VA		AC Apparent Power Phase A value	
40095	40095	1	R	0x03	VApH B	int16	VA		AC Apparent Power Phase B value	
40096	40096	1	R	0x03	VApH C	int16	VA		AC Apparent Power Phase C value	
40097	40097	1	R	0x03	VA_SF	int16	sunssf		AC Phase Apparent Power Scale factor	
40098	40098	1	R	0x03	VAR	int16	VAr		AC Reactive Power value	
40099	40099	1	R	0x03	VARphA	int16	VAr		AC Reactive Power Phase A value	
40100	40100	1	R	0x03	VARphB	int16	VAr		AC Reactive Power Phase B value	
40101	40101	1	R	0x03	VARphC	int16	VAr		AC Reactive Power Phase C value	
40102	40102	1	R	0x03	VAR_SF	int16	sunssf		AC Phase Reactive Power Scale factor	
40103	40103	1	R	0x03	PF	int16	PF		Power Factor value	
40104	40104	1	R	0x03	PFphA	int16	PF		Power Factor Phase A value	
40105	40105	1	R	0x03	PFphB	int16	PF		Power Factor Phase B value	
40106	40106	1	R	0x03	PFphC	int16	PF		Power Factor Phase C value	

**Ganzzahldarstellung – „int+SF“**

<b>Start</b>	<b>End</b>	<b>Size</b>	<b>R/W</b>	<b>Function codes</b>	<b>Name</b>	<b>Type</b>	<b>Units</b>	<b>Scale factor</b>	<b>Description</b>	<b>Range of values</b>
40107	40107	1	R	0x03	PF_SF	int16	sunssf		Phase Power Factor Scale factor	
40108	40109	2	R	0x03	TotWhExp	uint32	Wh		Total Watt-hours Exported	
40110	40111	2	R	0x03	TotWhExpPhA	uint32	Wh		Total Watt-hours Exported phase A	
40112	40113	2	R	0x03	TotWhExpPhB	uint32	Wh		Total Watt-hours Exported phase B	
40114	40115	2	R	0x03	TotWhExpPhC	uint32	Wh		Total Watt-hours Exported phase C	
40116	40117	2	R	0x03	TotWhImp	uint32	Wh		Total Watt-hours Imported	
40118	40119	2	R	0x03	TotWhImpPhA	uint32	Wh		Total Watt-hours Imported phase A	
40120	40121	2	R	0x03	TotWhImpPhB	uint32	Wh		Total Watt-hours Imported phase B	
40122	40123	2	R	0x03	TotWhImpPhC	uint32	Wh		Total Watt-hours Imported phase C	
40124	40124	1	R	0x03	TotWh_SF	int16	sunssf		Total Watt Scale factor	
40125	40126	2	R	0x03	TotVAhExp	uint32	VAh		Total VA-hours Exported	
40127	40128	2	R	0x03	TotVAhExpPhA	uint32	VAh		Total VA-hours Exported phase A	
40139	40130	2	R	0x03	TotVAhExpPhB	uint32	VAh		Total VA-hours Exported phase B	
40131	40132	2	R	0x03	TotVAhExpPhC	uint32	VAh		Total VA-hours Exported phase C	
40133	40134	2	R	0x03	TotVAhImp	uint32	VAh		Total VA-hours Imported	
40135	40136	2	R	0x03	TotVAhImpPhA	uint32	VAh		Total VA-hours Imported phase A	
40137	40138	2	R	0x03	TotVAhImpPhB	uint32	VAh		Total VA-hours Imported phase B	
40139	40140	2	R	0x03	TotVAhImpPhC	uint32	VAh		Total VA-hours Imported phase C	
40141	40141	1	R	0x03	TotVAh_SF	int16	sunssf		Total VA-hours Scale factor	
40142	40143	2	R	0x03	TotVArhImpQ1	uint32	VArh		Total VAR-hours Imported Q1	
40144	40145	2	R	0x03	TotVArhImpQ1phA	uint32	VArh		Total VAR-hours Imported Q1 phase A	
40146	40147	2	R	0x03	TotVArhImpQ1phB	uint32	VArh		Total VAR-hours Imported Q1 phase B	
40148	40149	2	R	0x03	TotVArhImpQ1phC	uint32	VArh		Total VAR-hours Imported Q1 phase C	

### Ganzzahldarstellung – „int+SF“

<b>Start</b>	<b>End</b>	<b>Size</b>	<b>R/W</b>	<b>Function codes</b>	<b>Name</b>	<b>Type</b>	<b>Units</b>	<b>Scale factor</b>	<b>Description</b>	<b>Range of values</b>
40150	40151	2	R	0x03	TotVArhImpQ2	uint32	VArh		Total VAr-hours Imported Q2	
40152	40153	2	R	0x03	TotVArhImpQ2phA	uint32	VArh		Total VAR-hours Imported Q2 phase A	
40154	40155	2	R	0x03	TotVArhImpQ2phB	uint32	VArh		Total VAR-hours Imported Q2 phase B	
40156	40157	2	R	0x03	TotVArhImpQ2phC	uint32	VArh		Total VAR-hours Imported Q2 phase C	
40158	40159	2	R	0x03	TotVArhExpQ3	uint32	VArh		Total VAr-hours Exported Q3	
40160	40161	2	R	0x03	TotVArhExpQ3phA	uint32	VArh		Total VAR-hours Exported Q3 phase A	
40162	40163	2	R	0x03	TotVArhExpQ3phB	uint32	VArh		Total VAR-hours Exported Q3 phase B	
40164	40165	2	R	0x03	TotVArhExpQ3phC	uint32	VArh		Total VAR-hours Exported Q3 phase C	
40166	40167	2	R	0x03	TotVArhExpQ4	uint32	VArh		Total VAr-hours Exported Q4	
40168	40169	2	R	0x03	TotVArhExpQ4phA	uint32	VArh		Total VAR-hours Exported Q4 phase A	
40170	40171	2	R	0x03	TotVArhExpQ4phB	uint32	VArh		Total VAR-hours Exported Q4 phase B	
40172	40173	2	R	0x03	TotVArhExpQ4phC	uint32	VArh		Total VAR-hours Exported Q4 phase C	
40174	40174	1	R	0x03	TotVArh_SF	int16	sunssf		Total VAR-hours Scale factor	
40175	40176	2	R	0x03	Evt	uint32	Bitfield		Events	

# End Block

## Allgemeines

Zwei Register nach dem letzten Datenmodell zeigen an, dass keine weiteren SunSpec-Modelle mehr folgen.

Die Adressen dieser beiden Register sind je nach Gerätetyp (Wechselrichter, String Control, Energiezähler) und ausgewähltem Datentyp („float“ oder „int+SF“) verschieden.

- Wechselrichter:
  - Startadresse für bei Einstellung „float“: 40313
  - Startadresse bei Einstellung „int+SF“: 40303
- String Control:
  - Startadresse: 40127
- Energiezähler:
  - Startadresse für bei Einstellung „float“: 40195
  - Startadresse bei Einstellung „int+SF“: 40176

## End Block

Start Offset	End Offset	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
1	1	1	R	0x03	ID	uint16			Identifies this as End block	0xFFFF
2	2	1	R	0x03	L	uint16	Registers		Length of model block	0

# String Combiner Event Flags

---

String Combiner  
Event Flags

Name	Event Flags
LOW_VOLTAGE	0x00000001
LOW_POWER	0x00000002
LOW EFFICIENCY	0x00000004
CURRENT	0x00000008
VOLTAGE	0x00000010
POWER	0x00000020
PR	0x00000040
DISCONNECTED	0x00000080
FUSE_FAULT	0x00000100
COMBINER_FUSE_FAULT	0x00000200
COMBINER_CABINET_OPEN	0x00000400
TEMP	0x00000800
GROUNDFAULT	0x00001000
REVERSED_POLARITY	0x00002000
INCOMPATIBLE	0x00004000
COMM_ERROR	0x00008000
INTERNAL_ERROR	0x00010000
THEFT	0x00020000
ARC_DETECTED	0x00040000

# Contents

The Modbus Protocol .....	73
General .....	73
Structure of Modbus Messages .....	73
Modbus TCP – MBAP Header .....	74
Supported Function Codes .....	74
03 (0x03) Read Holding Registers .....	74
06 (0x06) Write Single Register .....	75
16 (0x10) Write Multiple Registers .....	75
Exception Codes .....	76
CRC Calculation for Modbus RTU .....	77
Calculating CRC Checksum .....	78
Adding CRC Checksum to the Message .....	79
General .....	80
Abbreviations Used .....	80
Communication with the Modbus Master .....	80
Register Maps .....	81
Response Times .....	81
Modbus Device ID for Inverters .....	82
Modbus Device ID for Fronius String Controls .....	82
Modbus Device ID for Energy Meters .....	83
Event Flags .....	83
Register Addresses .....	83
Unavailable Data Records .....	85
Time Response of the Supported Operating Modes .....	86
Sign Convention for the Power Factor .....	87
Values Saved on the Card .....	87
Scale Factors .....	88
Non-Writable Registers .....	88
Entering Invalid Values .....	88
Modbus Settings .....	89
General .....	89
Opening the Modbus Settings .....	89
Data Output via Modbus .....	90
Limit Control .....	92
Save or Reject Changes .....	93
Fronius Registers .....	94
Fronius Register .....	94
Inverter Status Code .....	94
Deleting Event Flags and Status Codes .....	94
Saving and Deleting Data .....	95
Changing the Data Type .....	95
System Totals .....	95
Common & Inverter Model .....	96
Common Block Register .....	96
Inverter Model Register .....	96
SunSpec State Codes .....	101
Fronius State Codes .....	102
Nameplate model (IC120) .....	103
General .....	103
Nameplate Register .....	103
Basic Settings Model (IC121) .....	105
Basic Settings Register .....	105
Reference Voltage .....	107
Deviation from Reference Voltage .....	107
Extended Measurements & Status Model (IC122) .....	108
General .....	108
Extended Measurements & Status Register .....	108
Immediate Control Model (IC123) .....	111
General .....	111
Immediate Controls Register .....	111

Standby.....	113
Power reduction .....	113
Example: Setting a Power Reduction .....	113
Example: Changing the Return Time When Power Reduction Has Been Activated .....	114
Effects of Reactive Power Specifications on Effective Power.....	114
Constant Power Factor .....	115
Example: Setting a Constant Power Factor .....	115
Constant Relative Reactive Power .....	116
Example: Setting Constant Reactive Power .....	116
<b>Basic Storage Control Model (IC124) .....</b>	<b>117</b>
General .....	117
Information Provided.....	117
Power Window Specifications.....	117
Setting the Minimum Charge Level.....	119
Charging the Energy Storage via the Grid .....	119
Basic Storage Controls Register.....	119
<b>Multiple MPPT Inverter Extension Model (I160) .....</b>	<b>122</b>
General .....	122
Multiple MPPT Inverter Extension Register .....	122
<b>String Combiner Model (SC403).....</b>	<b>125</b>
String Combiner Register.....	125
<b>Meter Model .....</b>	<b>127</b>
Meter Model Register.....	127
<b>End Block.....</b>	<b>134</b>
General .....	134
End Block.....	134
<b>String Combiner Event Flags .....</b>	<b>135</b>
String Combiner Event Flags .....	135

# The Modbus Protocol

## General

The description of the protocol is largely taken from the Modbus specifications, which are publicly available at [www.modbus.org/specs.php](http://www.modbus.org/specs.php).

Modbus is a simple, open communication protocol, with which master-slave or client-server communication can be carried out between the devices connected to the network. The basic principle of Modbus is: A master sends a request and a slave responds to this. In Modbus TCP, the master is referred to as the client and a slave as a server. The function is the same. The descriptions of the protocol functions provided below will use the more common names master and slave, irrespective of the RTU and TCP variants. In cases where there are differences between RTU and TCP, this will be specifically indicated.

Modbus can be used in two ways on the Fronius Datamanager:

- Modbus TCP  
using TCP/IP via Ethernet (connected by cable or via WLAN)
- Modbus RTU  
using asynchronous serial transmission via RS-485 (EIA/TIA-485-A), only for Fronius Datamanager 2.0.

In the case of Modbus RTU, there can only ever be one master in the system. In principle, only one master may initiate requests. A slave may only give a response if it has been addressed by the master; the slaves cannot communicate with each other. If a broadcast request (request to all available slaves via slave ID or unit ID 0) is sent, none of the slaves can respond. Broadcasts can therefore only be used for write commands.

If a master sends a request to a slave, then it expects a response. In the event of a request from a master, there are five options:

- If the slave receives the request without communication errors and can process this request without errors, then a normal response will be sent with the required data.
- If the slave does not receive the request due to a communication error, then no response is sent. This leads to a timeout on the master.
- If the slave receives the request, but discovers a communication error (parity, CRC, etc.), then no response is sent. This leads to a timeout on the master.
- If the slave receives the request without communication errors, but cannot process it without errors (e.g., if a register that is not available needs to be read), then an error message (exception response) is returned with the reason for the error.
- If the slave receives a broadcast request, which also goes to all other devices, then no response will be sent either in the event of an error or if the request has been successfully processed. Broadcast requests are therefore only suitable for write commands.

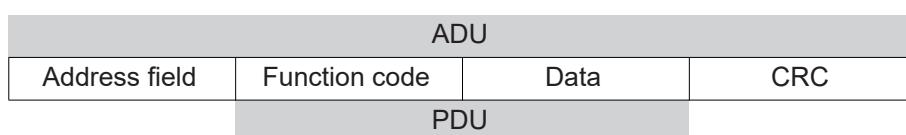
Modbus devices provide data in 16 bit large data blocks (registers).

In certain cases, individual data points may also cover several data blocks (e.g., 2 registers = 32 bit value).

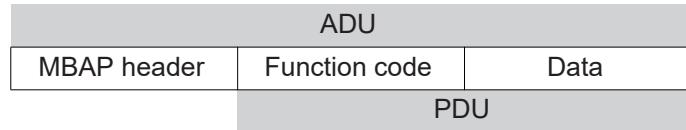
## Structure of Modbus Messages

In principle, a Modbus message is made up of the protocol data unit (PDU). This is independent of the underlying communication layers.

Depending on the bus or network that is used, additional fields can also be added. This structure is then referred to as the application data unit (ADU).



Structure of a Modbus message for Modbus RTU



*Structure of a Modbus message for Modbus TCP*

Modbus TCP uses its own header to identify the application data unit. This header is called MBAP header (MODBUS application protocol header).

The size of the protocol data unit (PDU) is limited due to the first Modbus implementations in a serial network (max. RS-485 ADU = 256 bytes). This results in the following for the size of the protocol data unit PDU: PDU = 256 – slave ID (1 byte) – CRC (2 bytes) = 253 bytes. This results in:

- Modbus RTU ADU = 253 + slave ID (1 byte) + CRC (2 bytes) = 256 bytes
- Modbus TCP ADU = 253 bytes + MBAP (7 bytes) = 260 bytes

#### Modbus TCP – MBAP Header

The MBAP header includes 7 bytes:

- **Transaction ID** (2 bytes): Is used in order to synchronize request and response. The slave adopts the transaction ID from the request into the response.
- **Protocol ID** (2 bytes): Is always 0 (Modbus protocol).
- **Length** (2 bytes): The length field includes the number of bytes of the subsequent fields, including unit ID and data fields.
- **Unit ID** (1 byte): This field is used for addressing devices connected to the Fronius Datamanager (gateway function of the Fronius Datamanager). The unit ID corresponds to the slave ID in Modbus RTU. The value is specified by the master and is returned unchanged by the slave with the response.

For details about the addressing of the devices, see:

- Modbus Device ID for Inverters on page 82
- Modbus Device ID for Fronius String Controls on page 82
- Modbus Device ID for Energy Meters on page 83

**IMPORTANT:** The correct unit ID must always be specified, even if the Fronius Data-manager is only connected to one individual inverter.

#### Supported Func- tion Codes

The function code determines the action to be carried out on the slave. The Fronius Data-manager supports three function codes for read and write operations:

- 03 (0x03)<sup>1)</sup> read holding registers
- 06 (0x06)<sup>1)</sup> write single register
- 16 (0x10)<sup>1)</sup> write multiple registers.

If an error occurs on the slave during the processing of a request, an error message is sent as the response (exception response). In the event of this kind of response, the most significant bit of the function code is set to 1 (corresponds to adding 0x80 to the function code) <sup>1)</sup> and an exception code is added, which indicates the reason for the error.

<sup>1)</sup> The prefix "0x" stands for hexadecimal numbers.

#### 03 (0x03) Read Holding Registers

This function code is used to read the content of one or more successive registers of a device. The request contains the address of the first register to be read and the number of registers to be read. Registers are addressed in the request starting at 0. This means that registers 1 to 16 will be addressed via addresses 0 to 15.

**Request**

Function code	1 byte	0x03
Start address	2 bytes	0x0000 to 0xFFFF (0 to 65535)
Number of registers	2 bytes	1 to 125

**Response**

Function code	1 byte	0x03
Number of bytes	1 byte	2 x N*
Register values	N* x 2 bytes	

\*N = number of registers

**Error**

Error code	1 byte	0x83
Exception code	1 byte	01 or 02 or 03 or 04 or 11

**06 (0x06) Write Single Register**

This function code is used in order to write a single register. The request only contains the address of the register to be written. Registers are addressed starting at 0. This means that register 1 is addressed via address 0. The normal response is a copy of the request, which is sent after successfully writing the register.

**Request**

Function code	1 byte	0x06
Register address	2 bytes	0x0000 to 0xFFFF (0 to 65535)
Register value	2 bytes	

**Response**

Function code	1 byte	0x06
Register address	2 bytes	0x0000 to 0xFFFF (0 to 65535)
Register value	2 bytes	

**Error**

Error code	1 byte	0x86
Exception code	1 byte	01 or 02 or 03 or 04 or 11

**16 (0x10) Write Multiple Registers**

This function code is used in order to write a block of successive registers. The request contains the address of the first register to be written, the number of registers to be written, the number of bytes to be written, and the values to be written (2 bytes per register). The normal response contains the function code, the start address, and the number of registers written.

### Request

Function code	1 byte	0x10
Start address	2 bytes	0x0000 to 0xFFFF (0 to 65535)
Number of registers	2 bytes	1 to 123
Number of bytes	1 byte	2 x N*
Register values	N* x 2 bytes	

\*N = number of registers

### Response

Function code	1 byte	0x10
Start address	2 bytes	0x0000 to 0xFFFF (0 to 65535)
Number of registers	2 bytes	1 to 123

### Error

Error code	1 byte	0x90
Exception code	1 byte	01 or 02 or 03 or 04 or 11

- Exception Codes** An error message (exception response) has two fields, which distinguishes it from a normal response:
- **Function code field**  
In a normal response, the function code of the request is adopted into the function code field of the response. In all function codes, the most significant bit (MSB) is 0 (the values of the function codes are all lower than 0x80). In an error message, the MSB is set to 1. This means that 0x80 is added to the value for the function code. The master can identify the response as an error message due to the set MSB.
  - **Data field**  
A normal response contains data or statistical values in the data field. In an error message, an exception code is returned in the data field. This exception code indicates the reason for the error message.

Modbus Exception Codes		
Code	Name	Meaning
01	ILLEGAL FUNCTION	The function code in the request is not supported by the slave.
02	ILLEGAL DATA ADDRESS	Invalid register addresses have been requested.
03	ILLEGAL DATA VALUE	A value in the request is outside of the valid range. This applies both for the fields of a request (e.g., invalid number of registers) and for invalid setting values for the SunSpec inverter control models.
04	SLAVE DEVICE FAILURE	An error occurred during an attempt to write one or more registers.
11	GATEWAY TARGET DEVICE FAILED TO RESPOND	Only for Modbus TCP. The addressed device cannot be found: a) the device is not in the SolarNet Ring or b) the device is switched off or c) the SolarNet Ring is open.

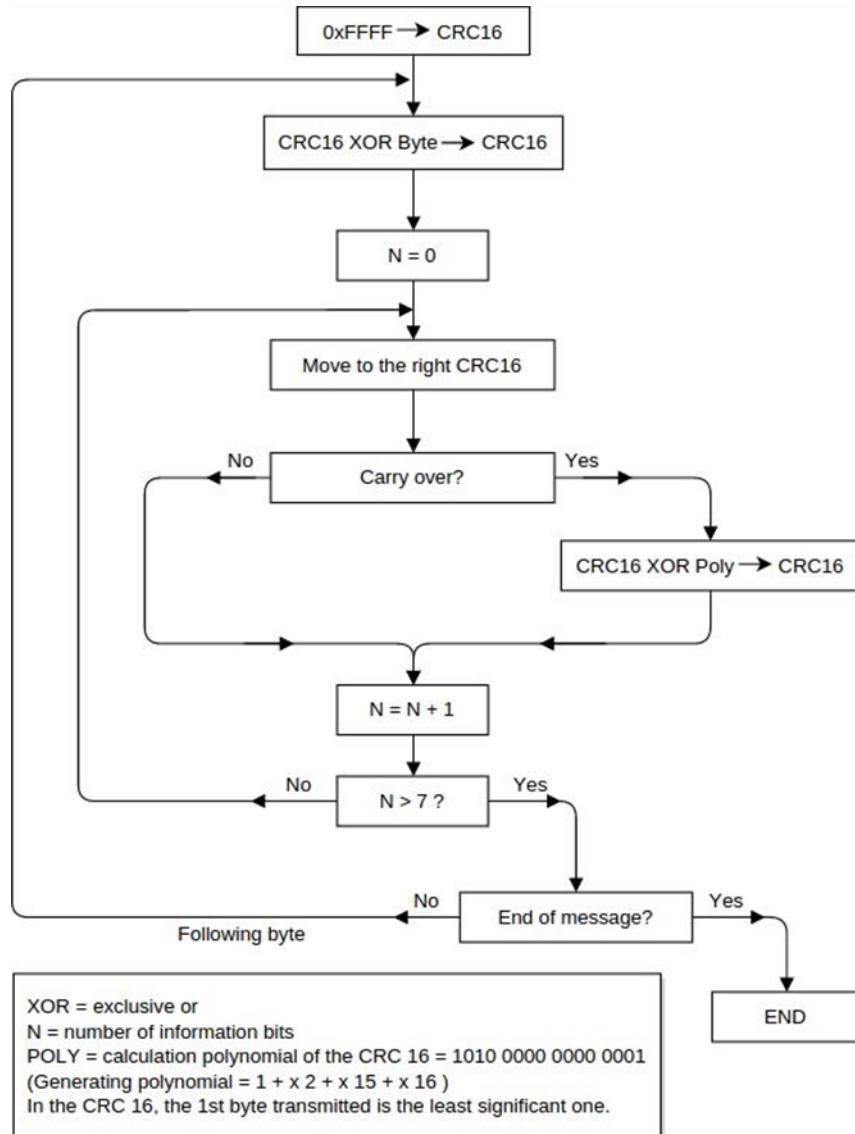
## CRC Calculation for Modbus RTU

Each Modbus RTU message is equipped with a checksum (CRC, Cyclic Redundancy Check) in order to be able to identify transmission errors. The size of the checksum is 2 bytes. It is calculated by the sending device and attached to the message to be sent. For its part, the receiver calculates the checksum from all bytes of the received message (without CRC) and compares this with the received checksum. If these two checksums are different, then an error has occurred.

The calculation of the checksum starts with setting all bits of a 16 bit register (CRC register) to 1 (0xFFFF). All bytes of the message are then individually processed with the CRC register. Only the data bytes of one message are used for the calculation. Start, stop, and parity bits are not considered.

During the calculation of the CRC, each byte is XOR-linked with the CRC register. The result is then moved in the direction of the least significant bit (LSB) and the most significant bit (MSB) is set to 0. The LSB is considered. If the LSB was previously 1, then the CRC register is XOR-linked with a fixed assigned value. If the LSB was 0, then nothing needs to be done.

This process is repeated until the CRC register has been moved eight times. After the last (eighth) movement, the next byte is taken and XOR-linked to the current CRC register. The write process then starts from the beginning; it is again moved eight times. After dealing with all bytes of the message, the value of the CRC register is the checksum.



*Calculation algorithm of the CRC16*

### Calculating CRC Checksum

- 1 Initialize a 16 bit register (2 bytes) with 0xFFFF. This register is referred to as the CRC16 register.
- 2 XOR-link the first byte of the message with the less significant byte of the CRC16 register. The result is saved in the CRC16 register.
- 3 Move the CRC16 register 1 bit to the right (in the direction of the LSB), fill MSB with 0. Look at LSB.
- 4 Check LSB value
  - If the LSB was 0: Go to step 3 (move again).
  - If the LSB was 1: XOR-link the CRC16 register with the CRC polynomial 0xA001 (1010 0000 0000 0001).
- 5 Repeat steps 3 and 4 until eight movement operations have been carried out. When these have been carried out, a complete byte of the message will have been processed.
- 6 Repeat steps 3 to 5 for the next byte of the message. Repeat everything until all bytes of the message have been processed.
- 7 After the last byte, the CRC16 register contains the checksum.

- 8** When the checksum is added to the message to be sent, then the two bytes must be inverted as described below.

---

#### Adding CRC Checksum to the Message

If the 16 bit (2 bytes) CRC checksum is sent with a message, then the less significant byte is transferred before the more significant one.

For example, if the CRC checksum is 0x1241 (0001 0010 0100 0001):

Addr	Func	Data Count	Data	Data	Data	Data	CRC Lo	CRC Hi
							0x41	0x12

# General

## Abbreviations Used

AC	Alternating current
DC	Direct current
FW	Firmware
MBC	Fronius Modbus Card
PF	Power factor ( $\cos \varphi$ )
PV	Photovoltaics
RTC	Real-time clock
SF	Scale factor
SW	Software
V	Voltage (volts)
VA	Apparent power
VAr	Reactive power
VMax	Maximum voltage
VMin	Minimum voltage
VRef	Reference voltage
W	Power (watts)
IN	Inverter

## Communication with the Modbus Master

The Fronius Datamanager communicates with the Modbus master using register addresses in accordance with the SunSpec Alliance specifications.  
(<http://www.sunspec.org/>)



**NOTE!** The Fronius Datamanager also supports the integration of Fronius String Controls via Fronius Solar Net.

Fronius String Controls are displayed by an integrated Common Block and the subsequent String Combiner Model.

In addition, the Fronius Datamanager offers the option of providing via Modbus TCP data of an energy meter connected via Modbus RTU. The meter is displayed via an integrated Common Block and the subsequent Meter Model.

The allocation of register addresses to the corresponding function can be found in the following tables:

- For all devices:
  - Common Block (C001)
- For inverters:
  - Fronius Register
  - Inverter model (I101, I102, I103, I111, I112, or I113)
  - Inverter Controls:
    - Nameplate (IC120)
    - Basic Settings (IC121)
    - Extended Measurements & Status (IC122)
    - Immediate Controls (IC123)
  - Multiple MPPT Inverter Extension (I160)
- For Fronius String Controls:
  - String Combiner Model (SC403)
- For energy meters:
  - Meter Model (M201, M202, M203, M211, M212, or M213).



**NOTE!** Only applies for Modbus RTU and only if no energy meter is connected: If no data exchange takes place on the RS-485 bus, noise and interference may affect the lines. In order for a receiver to remain in a defined status when there are no data signals, bias resistors should be used in order to maintain a defined idle state on the data lines.

The Fronius Datamanager does not have any bias resistors. Detailed information about the use of these resistors can be found in the document "MODBUS over serial line specification and implementation guide V1.02" ([http://modbus.org/docs/Modbus\\_over\\_serial\\_line\\_V1\\_02.pdf](http://modbus.org/docs/Modbus_over_serial_line_V1_02.pdf)).

## Register Maps

Inverter	Fronius String Control	Energy Meter
<b>SID</b> Identification as a SunSpec device	<b>SID</b> Identification as a SunSpec device	<b>SID</b> Identification as a SunSpec device
<b>Common Block</b> Device information	<b>Common Block</b> Device information	<b>Common Block</b> Device information
<b>Inverter Model</b> Inverter data	<b>String Combiner Model</b> Fronius String Control data	<b>Meter Model</b> Energy meter data
<b>Nameplate Model</b>	<b>End Block</b>	<b>End Block</b>
<b>Basic Settings Model</b>		
<b>Ext. Measurement Model</b>		
<b>Immediate Controls Model</b>		
<b>Multi. MPPT Inv. Ext. Model</b>		
<b>End Block</b>		

The map lists can be downloaded in xlsx format from the Fronius website:

[www.fronius.com – Solar Energy – Info & Support – Documents & Downloads – Third-Party Downloads](http://www.fronius.com – Solar Energy – Info & Support – Documents & Downloads – Third-Party Downloads)

## Response Times

The response times depend on factors such as the number of devices in the Fronius Solar Net ring. The higher the number of devices used, the longer the timeout for responses needs to be.



**NOTE!** If there are several devices in the Fronius Solar Net Ring, a timeout of at least 1 second should be used when querying inverter data.

### Recommendation for Timeout Values

When using Fronius String Controls, a single Modbus request might result in two requests being sent via Fronius Solar Net; this can lead to longer response times than when using inverter requests. If Fronius String Controls are present, you should therefore use a higher timeout value for responses.

When first requesting common block data after restarting the Fronius Datamanager, the information about the Fronius String Control must first be requested using Fronius Solar Net. For this reason, this first request will take a little more time than subsequent requests.

If there are more devices in a Fronius Solar Net Ring, it is advisable to split these between several Fronius Solar Net Rings, which each have their own Fronius Datamanager, in order to speed up responses further.

---

<b>Modbus Device ID for Inverters</b>	The inverter's Modbus device ID is the same as its inverter number, which can be set using the control panel on the inverter. (See the inverter operating instructions.)
---------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------



**NOTE!** There is only one exception to this rule:

The inverter number 00 converts to device ID 100 because Modbus reserves device ID 0 for broadcast messages.

Example:

Inverter number	Modbus device ID
00	100
01	001
02	002
03	003
99	099

---

<b>Modbus Device ID for Fronius String Controls</b>	The Modbus device ID of a Fronius String Control is derived from - its address in Fronius Solar Net - a String Control offset value.
	<p>The default value for the String Control offset is 101 because the range reserved for inverters goes up to Modbus device ID 100.</p> <p>The offset value can, however, be adjusted via the Fronius Datamanager web page. =&gt; see section "Data Output via Modbus"</p>

**Example 1:** String Control offset = 101 (standard value)

<i>Fronius String Control address</i>	<i>Modbus device ID</i>
0	101
1	102
2	103
99	200

A Fronius Solar Net Ring allows up to 100 inverters and up to 200 Fronius String Controls. The available Modbus device IDs are reserved for other functions (e.g., for energy meters) from 240.

With the standard String Control offset of 101, it would therefore not be possible to have Fronius String Control addresses from 139 (which corresponds to Modbus ID 240) upwards.

For this reason, it is possible to adjust the String Control offset on the Fronius Datamanager website if fewer than 100 inverters are being used.

**Example 2:** 30 inverters, 200 Fronius String Controls, String Control offset = 40

<i>Fronius String Control address</i>	<i>Modbus device ID</i>
0	40
1	41
2	42
199	239

---

<b>Modbus Device ID for Energy Meters</b>	If an energy meter (e.g., Fronius Smart Meter 63A) is connected to the Fronius Datamanager via Modbus RTU, it can be read out via the fixed Modbus device ID 240 using Modbus TCP.
-------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

---

<b>Event Flags</b>	Status changes and faults in the inverters and Fronius String Controls are shown as event flags.
--------------------	--------------------------------------------------------------------------------------------------

Detailed information and lists can be downloaded in various formats (xlsx, csv, json) from the Fronius website:

[www.fronius.com – Solar Energy – Info & Support – Documents & Downloads – Third-Party Downloads](http://www.fronius.com – Solar Energy – Info & Support – Documents & Downloads – Third-Party Downloads)



**NOTE!** It is also possible to combine several state codes for one event.

#### For inverters:

An accurate description of the state codes can be found in the operating instructions of the relevant inverter.

If the inverter generates a state code, the relevant event flag is set in the Fronius Datamanager.



**NOTE!** In addition, the relevant state code is also displayed in register F\_Active\_State\_Code (214).

The event flag and state code will remain active for as long as the state code is displayed on the inverter. If another state code is generated, it will also be displayed in the event flags. In this case, there is a chance that the previous event flag will not be deleted.

It is therefore possible to manually delete the event flags and the state code by entering 0xFFFF in register F\_Reset\_All\_Event\_Flags (215).

---

<b>Register Addresses</b>
---------------------------

#### IMPORTANT!

- Register addresses do not remain constant.
- The actual register addresses depend on the composition of the dynamic SunSpec register list.

Correct procedure:

- Search for the model by making a request (determine start address)
- Then work with offsets

To read a register, the register's start address must be specified in the Modbus request.

Fronius Basic Register: 212

SunSpec Basic Register: 40001

Registers begin at 1 and do not represent a function code.

Do not confuse the registers with the Modicon address scheme:

In the Modicon address scheme, 40001 is displayed as 4x40001.

To read register 40001, use address 40000 (0x9C40).



**NOTE!** The register address that is output therefore always has 1 number less than the actual register number.

**IMPORTANT!** The lengths of individual models may vary due to the data types used. Start addresses are therefore specified for SunSpec models in the case of some register tables. This start address, together with the offset from the table, then produces the value of the actual register number.

**Example:** Table Nameplate model (IC120) on page 103:

The register *WRtg* of the nameplate model has an offset of 4. The start address is specified as 40131 with the setting “float”.

Therefore, the correct register number is:  $40131 + 4 = 40135$ .

#### Examples for Modbus RTU:

##### 1. Request for four registers starting from register 40005 (Mn, Manufacturer)

Send (bytes in hexadecimal)

01	03	9C 44	00 04	2A 4C				
Device ID	Function code	Address 40004 (corresponds to register 40005)	Number of registers to be read	Checksum Low byte	High byte			

Receive (bytes in hexadecimal)

01	03	08	46 72	6F 6E	69 75	73 00	8A 2A	
Device ID	Function code	Num- ber of bytes	Address 40005 “F” and “r”	Address 40006 “o” and “n”	Address 40007 “i” and “u”	Address 40008 “s” and 0	Checksum Low byte	High byte

##### 2. Enter one register starting from register 40242 (WmaxLimPct)

01	10	9D 32	00 01	02	13 88	E3 DD	
Device ID	Function code	Address 40242	Number of registers to be entered	Number of data bytes still to follow	Register value to be entered $0x1388 = 5000$	Checksum Low byte	High byte

01	10	9D 32	00 01	8F AA	
Device ID	Function code	Address 40242	Number of registers entered	Checksums Low byte	High byte

#### Examples for Modbus TCP:

## 1. Request for four registers starting from register 40005 (Mn, Manufacturer)

Send (bytes in hexadecimal)

MBAP header	01	03	9C 44	00 04
For details, see description of MBAP header	Device ID	Function code	Address 40004 (corresponds to register 40005)	Number of registers to be read

Receive (bytes in hexadecimal)

MBAP header	01	03	08	46 72	6F 6E	69 75	73 00
For details, see description of MBAP header	Device ID	Function code	Number of bytes	Address 40005 "F" and "r"	Address 40006 "o" and "n"	Address 40007 "i" and "u"	Address 40008 "s" and 0

## 2. Enter one register starting from register 40242 (WmaxLimPct)

MBAP header	01	10	9D 32	00 01	02	13 88
For details, see description of MBAP header	Device ID	Function code	Address 40242	Number of registers to be entered	Number of data bytes still to follow	Register value to be entered 0x1388 = 5000

MBAP header	01	10	9D 32	00 01
For details, see description of MBAP header	Device ID	Function code	Address 40242	Number of registers entered

### Unavailable Data Records

Fronius inverters cannot always provide all the data specified in the SunSpec data models. Depending on the data type, this data is represented by the following values in accordance with the SunSpec specification:

- int16 (-32767 to 32767): 0x8000<sup>1)</sup>
- uint16 (0 to 65534): 0xFFFF
- acc16 (0 to 65535): 0
- enum16(0 to 65534): 0xFFFF
- bitfield16 (0 to 0x7FFF): 0xFFFF
- pad (0x8000): always 0x8000
- int32 (-2147483647 to 2147483647): 0x80000000
- uint32 (0 to 4294967294): 0xFFFFFFFF
- acc32 (0 to 4294967295): 0
- enum32(0 to 4294967294): 0xFFFFFFFF
- bitfield32 (0 to 0xFFFFFFFF): 0xFFFFFFFF
- int64 (-9223372036854775807 to 9223372036854775807): 0x8000000000000000
- acc64 (0 to 18446744073709551615): 0
- stringX: all X registers filled with 0x0000
- float32 (range see IEEE 754): 0x7FC00000 (NaN)
- sunssf (scaling factors; -10 to 10): 0x8000

<sup>1)</sup> The prefix "0x" stands for hexadecimal numbers.

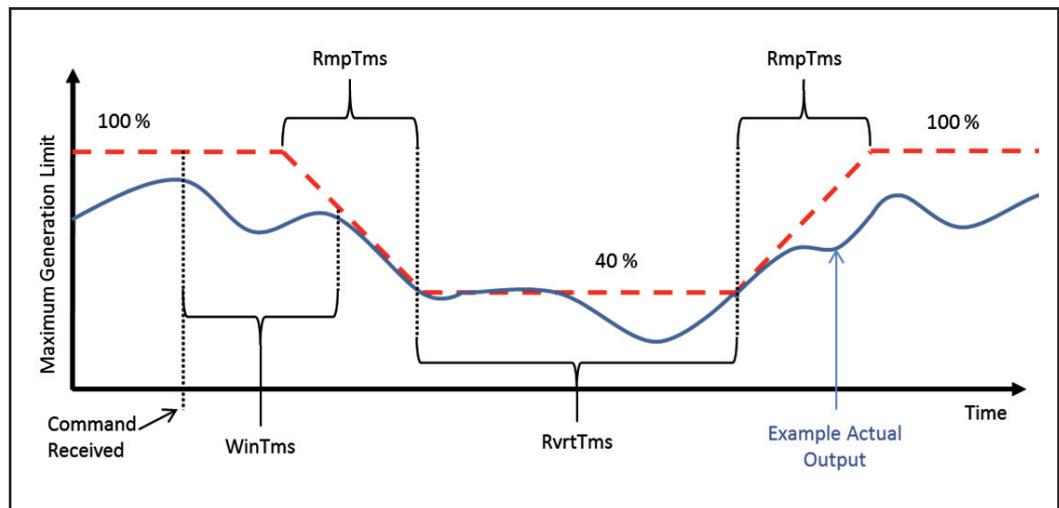


**NOTE!** Data points not supported by the data manager are marked with "Not supported" in the "Range of values" column in the register tables.

In this case, during reading, the corresponding value from the list above is obtained depending on the data type.

In certain instances, registers which are basically listed as supported may also return this value. This is because some values depend on the device type, e.g., currents AphB and AphC in the case of a single-phase inverter.

## Time Response of the Supported Operating Modes



Time response illustrated by power reduction

The inverter's time response in an operating mode can be defined by several time values. Three possible time values are shown in the figure "Time response illustrated by power reduction":

- **WinTms 0–300 [seconds]**

Specifies a time window in which the operating mode is randomly started. The time window starts when the start command for the operating mode is issued (e.g., *OutPF\_Set\_Ena* = 1).

*WinTms* can be used to prevent all the inverters in the system from applying the changes at the same time. If the time window is set to 0 (the default value), the operating mode will start immediately.

- **RvrTms 0–28800 [seconds]**

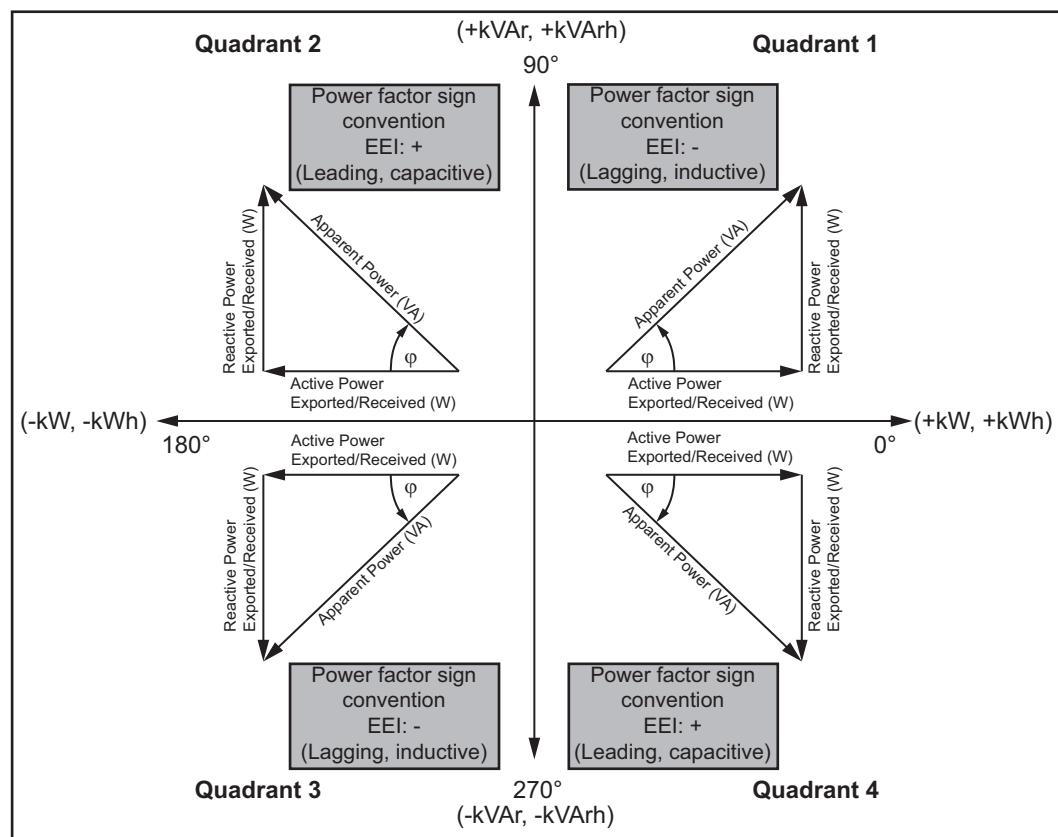
Determines how long the operating mode will remain active. After the time expires, the operating mode is automatically deactivated. If *RvrTms* is 0 (the default value), the operating mode remains active until it is manually deactivated via the corresponding register.

- **RmpTms (currently not supported by the Datamanager)**

Specifies how quickly the changes are to be made. The corresponding value gradually changes during the specified time period from the old to the new value.

If *RmpTms* is 0 (the default value) or if this value is not supported, the new value will be valid immediately.

## Sign Convention for the Power Factor



The EEI sign convention<sup>1)</sup> for the power factor is in line with the SunSpec specification and is based on the information contained in the "Handbook for Electricity Metering" and IEC 61557-12 (2007).

The power factor is:

- negative if the reactive power is positive (over-excited, quadrant 1)
- positive if the reactive power is negative (under-excited, quadrant 4)

<sup>1)</sup> EEI = Edison Electrical Institute

## Values Saved on the Card

Nameplate Model (IC120):

- **WRtg**  
AC nominal output of inverter.
- **VArRtg**  
AC nominal apparent output of inverter.  
Default value = WRtg
- **VArRtgQ1**  
Maximum AC reactive power in the first quadrant (over-excited).  
Default value is calculated based on the available cos Phi (0.85) and the nominal apparent power. Note the scaling factor VArRtg\_SF.
- **VArRtgQ4**  
Maximum AC reactive power in the fourth quadrant (under-excited).  
Default value is calculated based on the available cos Phi (0.85) and the nominal apparent power. Note the scaling factor VArRtg\_SF.
- **ARtg**  
AC nominal current of inverter.

### Basic Settings Model (IC121):

- **WMax**  
Maximum AC power  
Default value = WRtg
- **VRef**  
Reference voltage at the feed-in point
- **VRefOfs**  
Deviation from reference voltage
- **VMax**  
Maximum AC voltage
- **VMin**  
Minimum AC voltage
- **VAMax**  
Maximum AC apparent power  
Default value = VARtg

### Saving Values

If data is not available or is incorrectly displayed, the values listed above can be adjusted and saved on the Datamanager.

Changes currently have no influence on the way the Datamanager or the inverters function and are merely used to display device-specific information.

In order to save the values, the register *F\_Store\_Data* (213) of any inverter must be written with 0xFFFF. The values for all inverters are then permanently saved and are also available after an AC reset of the Datamanager.

### Deleting Values

It is only possible to delete values for an individual inverter. To do this, enter 0xFFFF into the register *F\_Delete\_Data* (212) of the relevant inverter.

---

### Scale Factors

**IMPORTANT!** Scale factors (also possible when selecting “Float”!) are not static, even if they are entered as a fixed value in these operating instructions.

Scale factors can change every time the firmware is changed (e.g.: scale factor for power specification).

Scale factors with constant values are listed in the tables in the column “Range of values”. Current data (data of inverters, Fronius String Controls, and energy meters) may have variable scale factors. These must be read from the corresponding registers.

---

### Non-Writable Registers

The following registers cannot be written:

- Read-only (R) registers
- Registers which are currently not supported



**NOTE!** If you try to write these registers, the Fronius Datamanager will not return an exception code!

The values entered into these registers will be ignored without any error message being issued by the Fronius Datamanager.

---

### Entering Invalid Values

Some registers only permit certain values. The valid values can be found in the relevant register table.

If an invalid value is entered into a register, the Fronius Datamanager will return exception code 3 (illegal data value). The invalid value is ignored.

If several registers are written at the same time, all the valid values will be entered up to the register containing the invalid value. The write operation will then be canceled.

# Modbus Settings

## General

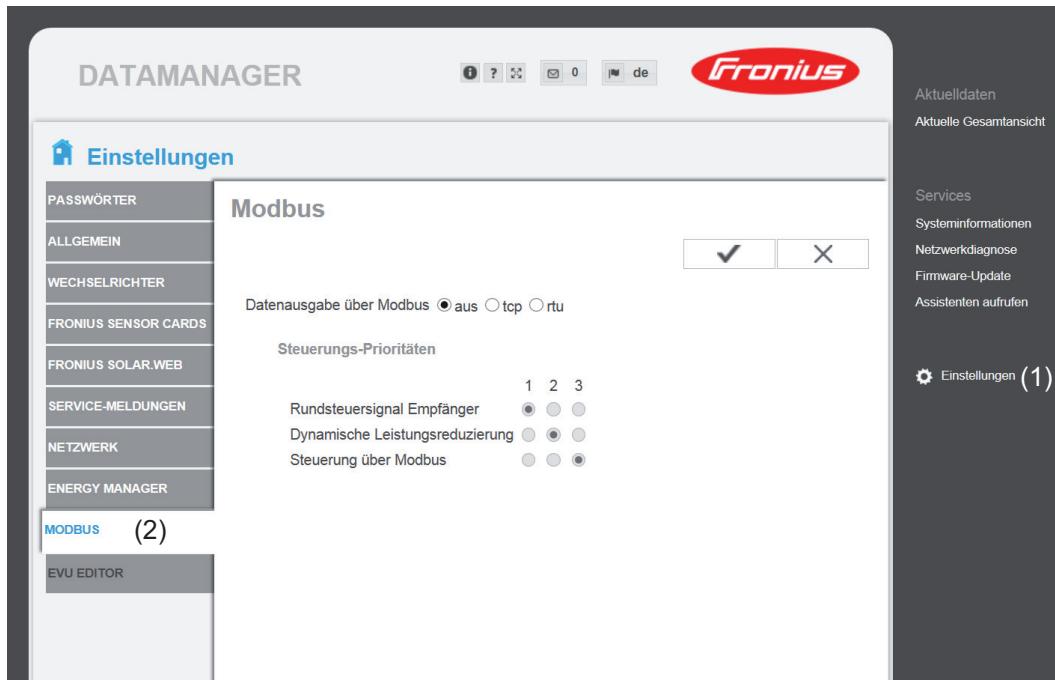
From your web browser, you can use the Fronius Datamanager web interface to apply the Modbus connection settings which cannot be accessed via the Modbus protocol.



**NOTE!** When transferring data via Modbus RTU the use of the web interface is usually not necessary because Modbus RTU is factory-adjusted.

## Opening the Modbus Settings

- 1 Install Fronius Datamanager  
=> see the Fronius Datamanager operating instructions.
- 2 Open Internet browser.
- 3 Enter the following in the address field of the Internet browser:
  - the IP address of the Fronius Datamanager (can be accessed via *System Information*)
  - or host name and domain name of the Fronius Datamanager.
 The web interface's start page is displayed.
- 4 Select the "Settings" section (1).
- 5 Open the "Modbus" section (2).



**NOTE!** In the case of Fronius Datamanager 2.0, the "Data output via Modbus" is set to RTU in the factory.

The RTU option is not available for the Fronius Datamanager.

## Data Output via Modbus

### Modbus

<p>(1) (2) (3)</p> <p>Data export via Modbus <input type="radio"/> off <input checked="" type="radio"/> tcp <input type="radio"/> rtu</p> <p>Modbus port <input type="text" value="502"/></p> <p>String control address offset <input type="text" value="101"/></p> <p>SunSpec Model Type <input checked="" type="radio"/> float <input type="radio"/> int + SF</p> <p>Demo mode <input type="checkbox"/></p> <p>Inverter control via Modbus <input checked="" type="checkbox"/></p> <p>Restrict the control <input type="checkbox"/></p>	<input checked="" type="checkbox"/> (5) <input type="checkbox"/> (6)												
<p><b>Controlling priority</b></p> <table border="0"><tr><td style="text-align: center;">1</td><td style="text-align: center;">2</td><td style="text-align: center;">3</td></tr><tr><td>IO control <input checked="" type="radio"/></td><td><input type="radio"/></td><td><input type="radio"/></td></tr><tr><td>Dynamic power reduction <input type="radio"/></td><td><input checked="" type="radio"/></td><td><input type="radio"/></td></tr><tr><td>Controlling via Modbus <input type="radio"/></td><td><input type="radio"/></td><td><input checked="" type="radio"/></td></tr></table> <p><small>Notification: a change of control priorities is possible only in the dno editor with the service password.</small></p> <p><b>Legend:</b> 1 ... highest priority 2 ... medium priority 3 ... lowest priority</p>		1	2	3	IO control <input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	Dynamic power reduction <input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	Controlling via Modbus <input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
1	2	3											
IO control <input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>											
Dynamic power reduction <input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>											
Controlling via Modbus <input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>											

### Data Output via Modbus

Activation of the Modbus service and selection of the transmission protocol.  
If the Modbus service is activated, additional entry fields are available.

The Modbus RTU transmission protocol is only available for Fronius Datamanager 2.0.



**NOTE!** If there is a Modbus energy meter (e.g., Fronius Smart Meter) configured under Settings/Meter on the system, it will not be possible to use the "RTU" setting.

In this case, data output via Modbus will be deactivated automatically upon selection of "RTU." This change will only be visible once the Datamanager web page has been reloaded.

Any energy meter connected to the Datamanager via RS-485 can also be read by Modbus TCP via the corresponding SunSpec models. The Modbus ID for the meter is 240.

(1) **off**

No data output via Modbus.

If the data output via Modbus is deactivated, control commands sent to the inverter via Modbus are reset, e.g., no power reduction or no reactive power specification.

(2) **tcp**

Data output via Modbus TCP.

<p>Data export via Modbus <input type="radio"/> off <input checked="" type="radio"/> tcp <input type="radio"/> rtu</p> <p>Modbus port <input type="text" value="502"/></p> <p>String control address offset <input type="text" value="101"/></p> <p>SunSpec Model Type <input checked="" type="radio"/> float <input type="radio"/> int + SF</p> <p>Demo mode <input type="checkbox"/></p> <p>Inverter control via Modbus <input checked="" type="checkbox"/></p>	<p>(2) (2a) (2b) (2c) (2d) (2e) (2f)</p>
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------

(2a) **Modbus port**  
Number of the TCP port which must be used for Modbus communication.

Presetting: 502  
Port 80 cannot be used for this purpose.

(2b) **Fronius String Control address offset**  
Offset value used to assign addresses to Fronius String Controls via Modbus.  
For further details, see the section entitled “Modbus Device ID for Fronius String Controls.”

#### **SunSpec model type**

Used to select the data type of data models for inverters and energy meters.

(2c) **float**  
Display as floating-point numbers.  
SunSpec inverter model I111, I112, or I113.  
SunSpec meter model M211, M212, or M213.

(2d) **int+SF**  
Display as integers with scaling factors.  
SunSpec inverter model I101, I102, or I103.  
SunSpec meter model M201, M202, or M203.

**IMPORTANT!** Since the different models have different numbers of registers, the register addresses of all the subsequent models also change when the data type is changed.

(2e) **Demo mode**  
The demo mode is used to implement or validate a Modbus master. It enables you to read inverter, energy meter, and Fronius String Control data without actually having to connect or activate a device. The same data is always sent back for all the registers.

(2f) **Inverter control via Modbus**  
If this option is activated, the inverter can be controlled via Modbus.  
The “Limit Control” selection field is displayed.  
Inverter control includes the following functions:  

- On/off
- Power reduction
- Setting a constant power factor (cos phi)
- Setting a constant reactive power

(3) **RTU**  
Data output via Modbus RTU.

Data export via Modbus	(3)	
<input type="radio"/> off	<input type="radio"/> tcp	<input checked="" type="radio"/> rtu
Baud rate	(3a)	
9600	<input type="button" value="▼"/>	
Parity	(3b)	
no	<input type="button" value="▼"/>	
String control address offset	(3c)	
101	<input type="button" value="▼"/>	
Sunspec Model Type	(3d)	
<input checked="" type="radio"/> float	<input type="radio"/> int + SF	
Demo mode	(3f)	
<input type="checkbox"/>		
Inverter control via Modbus	(3g)	
<input checked="" type="checkbox"/>		

**Notification:** when connecting a Fronius Smart Meter, Modbus RTU is automatically disabled

- (3a) **Baud rate**  
Used to enter the baud rate.
- (3b) **Parity**  
Selection field for entering the parity.
- (3c) **Fronius String Control address offset**  
Offset value used to assign addresses to Fronius String Controls via Modbus.  
For further details, see the section entitled “Modbus Device ID for Fronius String Controls.”

**SunSpec model type**

Used to select the data type of data models for inverters.

- (3d) **float**  
Display as floating-point numbers.  
SunSpec inverter model I111, I112, or I113.
- (3e) **int+SF**  
Display as integers with scaling factors.  
SunSpec inverter model I101, I102, or I103.

**IMPORTANT!** Since the different models have different numbers of registers, the register addresses of all the subsequent models also change when the data type is changed.

- (3f) **Demo mode**  
The demo mode is used to implement and validate a Modbus master. It enables you to read inverter, energy meter, and Fronius String Control data without actually having to connect or activate a device. The same data is always sent back for all the registers.
- (3g) **Inverter Control via Modbus**  
If this option is activated, the inverter is controlled via Modbus.  
Inverter control includes the following functions:  
  - On/off
  - Power reduction
  - Setting a constant power factor (cos phi)
  - Setting a constant reactive power

(4) **Control priorities**

Used to specify which service is given priority by the inverter control unit.

1 = highest priority, 3 = lowest priority.

The control priorities can only be changed in the **UC EDITOR** menu item.

(5) **“Apply/Save” button**

(6) **“Cancel/Discard entries” button**

**Limit Control**

The "Limit Control" option is only available for the TCP transmission protocols.  
It is used to block inverter control commands from unauthorized users by only permitting control for specific devices.

Inverter control via Modbus	<input checked="" type="checkbox"/>	
Restrict the control	<input checked="" type="checkbox"/>	(1)
IP address	<input type="text" value="10.5.34.1"/>	(2)

(1) **Limit Control**

If this option is activated, only certain devices will be able to send control commands.

(2) **IP address**

To limit inverter control to one or more devices, enter the IP addresses of the devices which are permitted to send commands to Fronius Datamanager in this field. Multiple entries are separated by commas.

Examples:

- one IP address: **98.7.65.4**
  - Control only permitted by IP address 98.7.65.4
- several IP addresses: **98.7.65.4, 222.44.33.1**
  - Control only permitted by IP addresses 98.7.65.4 and 222.44.33.1
- IP address range, e.g., from 98.7.65.1 to 98.7.65.254 (CIDR notation):  
**98.7.65.0/24**
  - Control only permitted through IP addresses 98.7.65.1 to 98.7.65.254

---

**Save or Reject Changes**



Saves the changes and displays a message confirming this.

If you exit the "Modbus" section without saving your changes, all the changes you have made will be rejected.



Prompts you to confirm whether or not you wish to reject the changes you have made and then reinstates the most recently saved values.

# Fronius Registers

## Fronius Register



**NOTE!** These registers only apply to inverters. These registers are not relevant to Fronius String Controls and energy meters.

Start	End	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
212	212	1	RW	0x03 0x06 0x10	F_Delete_Data	uint16			Delete stored data (e.g., ratings, curve data) of the current inverter by writing 0xFFFF	0xFFFF
213	213	1	RW	0x03 0x06 0x10	F_Store_Data	uint16			Rating data of all inverters connected to the Fronius Datamanager are persistently stored by writing 0xFFFF	0xFFFF
214	214	1	R	0x03	F_Active_State_Code	uint16			Current active state code of inverter – description can be found in inverter manual	
215	215	1	RW	0x03 0x06 0x10	F_Reset_All_Event_Flags	uint16			Write 0xFFFF to reset all event flags and active state code	0xFFFF
216	216	1	RW	0x03 0x06 0x10	F_ModelType	uint16				1: Float 2: Int & SF
500	501	2	R	0x03	F_Site_Power	uint32	W			
502	505	4	R	0x03	F_Site_Energy_Day	uint64	Wh			
506	509	4	R	0x03	F_Site_Energy_Year	uint64	Wh			
510	513	4	R	0x03	F_Site_Energy_Total	uint64	Wh			

## Inverter Status Code

Register **F\_Active\_State\_Code (214)** displays the inverter status code which has just been generated. This may also be displayed on the inverter's display. This code is also displayed as an event flag in the inverter model. The displayed code remains active for as long the inverter has the corresponding status. Alternatively, the status can also be deleted by using register **F\_Reset\_All\_Event\_Flags**.

## Deleting Event Flags and Status Codes

The event flags in the inverter models (101, 102, 103 and 111, 112, 113) remain active until the corresponding status is no longer present on the inverter. There are a few exceptional cases in which the event flags are not deleted. For this reason, it is possible to reset the event flags and the displayed status code by issuing the Modbus command.

- 1 Enter 0xFFFF in register **F\_Reset\_All\_Event\_Flags (215)**

The content of the following registers is deleted:

- *F\_Active\_State\_Code* (214)
- *Evt1*
- *Evt2*
- *EvtVnd1* to *EvtVnd4*

## Saving and Deleting Data

If the value 0xFFFF is written in the register ***F\_Store\_Data(213)***, then all nominal values (ratings) for all inverters are saved on the Fronius Datamanager. These values can be changed in the corresponding registers of the Nameplate Model and the Basic Settings Model. This can be useful if, for example, no nominal values could be automatically determined for a device and you want to enter the values manually.

If you want to delete the saved values for a particular inverter, you must write the value 0xFFFF in the ***F\_Delete\_Data(212)*** register. The values are then only deleted for this inverter. The deletion can only ever be applied to the inverter with which there is currently communication.

## Changing the Data Type

The data type for the data models for inverters and energy meters can be selected via the ***F\_ModelType(216)*** register. It is possible to select either display as floating point numbers (float, standard) or as integers with scale factors (int+SF).



**NOTE!** This setting only relates to the inverter model (inverter) and the meter model (energy meter). All other models continue to use integers and scale factors.

This setting functions in the same way as the web interface Modbus settings – SunSpec model type.

Setting options:

- Float = 1 (standard): Inverter model I111, I112, or I113; meter model M211, M212, or M213
- int+SF = 2: Inverter model I101, I102, or I103; meter model M201, M202, or M203.



**NOTE!** Since the different models have different numbers of registers, the register addresses of all the subsequent models also change when the data type is changed.

## System Totals

The following registers can be used to query power and energy data from all inverters connected to this Fronius Datamanager via Fronius Solar Net.

These values are displayed in Watt (W) or Watt hours (Wh) and do not require scale factors.

- ***F\_Site\_Power(500–501)***: Power
- ***F\_Site\_Energy\_Day(502–505)***: Daily Energy
- ***F\_Site\_Energy\_Year(506–509)***: Yearly Energy
- ***F\_Site\_Energy\_Total(510–513)***: Total energy of the entire system.

# Common & Inverter Model

## Common Block Register

The description of the Common Block including the SID register (register 40001–40002) for identification as a SunSpec device applies for each device type (inverter, Fronius String Control, energy meter). Each device has its own Common Block, which lists information about the device (model, serial number, SW version, etc.).

Start	End	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
40001	40002	2	R	0x03	SID	uint32			Well-known value. Uniquely identifies this as a SunSpec Modbus Map	0x53756e53 (SunS)
40003	40003	1	R	0x03	ID	uint16			Well-known value. Uniquely identifies this as a SunSpec Common Model block	1
40004	40004	1	R	0x03	L	uint16	Registers		Length of Common Model block	65
40005	40020	16	R	0x03	Mn	String32			Manufacturer	e.g., Fronius
40021	40036	16	R	0x03	Md	String32			Device model	e.g., IG+150V
40037	40044	8	R	0x03	Opt	String16			SW version of Datamanager	e.g., 3.3.6-13
40045	40052	8	R	0x03	Vr	String16			SW version of inverter	
40053	40068	16	R	0x03	SN	String32			Serial number of inverter, String Control, or energy meter	
40069	40069	1	R	0x03	DA	uint16			Modbus Device Address	1–247

## Inverter Model Register

Two different SunSpec Models are supported for the inverter data:

- the default set inverter model with floating point display (setting “float”; I111, I112, or I113)
- the inverter model with integers and scaling factors (setting “int+SF”; I101, I102, or I103)



**NOTE!** The register number of the two model types is different!

### Floating point display – “float”

Start	End	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
40070	40070	1	R	0x03	ID	uint16			Uniquely identifies this as a SunSpec Inverter Modbus Map 111: single phase 112: split phase 113: three phase	111, 112, 113
40071	40071	1	R	0x03	L	uint16	Registers		Length of inverter model block	60
40072	40073	2	R	0x03	A	float32	A		AC total current value	
40074	40075	2	R	0x03	AphA	float32	A		AC phase A current value	
40076	40077	2	R	0x03	AphB	float32	A		AC phase B current value	
40078	40079	2	R	0x03	AphC	float32	A		AC phase C current value	
40080	40081	2	R	0x03	PPVphAB	float32	V		AC voltage phase AB value	
40082	40083	2	R	0x03	PPVphBC	float32	V		AC voltage phase BC value	
40084	40085	2	R	0x03	PPVphCA	float32	V		AC voltage phase CA value	
40086	40087	2	R	0x03	PhVphA	float32	V		AC voltage phase-A-to-neutral value	
40088	40089	2	R	0x03	PhVphB	float32	V		AC voltage phase-B-to-neutral value	
40090	40091	2	R	0x03	PhVphC	float32	V		AC voltage phase-C-to-neutral value	
40092	40093	2	R	0x03	W	float32	W		AC power value	
40094	40095	2	R	0x03	Hz	float32	Hz		AC frequency value	
40096	40097	2	R	0x03	VA	float32	VA		Apparent power	
40098	40099	2	R	0x03	VAr	float32	VAr		Reactive power	
40100	40101	2	R	0x03	PF	float32	%		Power factor	
40102	40103	2	R	0x03	WH	float32	Wh		AC lifetime energy production	
40104	40105	2	R	0x03	DCA	float32	A		DC current value	DC current only if one MPPT available; with multiple MPPT “not implemented”

### Floating point display – “float”

Start	End	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
40106	40107	2	R	0x03	DCV	float32	V		DC voltage value	DC voltage only if one MPPT available; with multiple MPPT “not implemented”
40108	40109	2	R	0x03	DCW	float32	W		DC power value	Total DC power of all available MPPT
40110	40111	2	R	0x03	TmpCab	float32	°C		Cabinet temperature	
40112	40113	2	R	0x03	TmpSnk	float32	°C		Coolant or heat sink temperature	
40114	40115	2	R	0x03	TmpTrns	float32	°C		Transformer temperature	
40116	40117	2	R	0x03	TmpOt	float32	°C		Other temperature	
40118	40118	1	R	0x03	St	enum16	Enum-merated		Operating state	1)
40119	40119	1	R	0x03	StVnd	enum16	Enum-merated		Vendor defined operating state	2)
40120	40121	2	R	0x03	Evt1	unit32	Bit field		Event flags (bits 0–31)	3)
40122	40123	2	R	0x03	Evt2	unit32	Bit field		Event flags (bits 32–63)	
40124	40125	2	R	0x03	EvtVnd1	unit32	Bit field		Vendor defined event flags (bits 0–31)	
40126	40127	2	R	0x03	EvtVnd2	unit32	Bit field		Vendor defined event flags (bits 32–63)	3)
40128	40129	2	R	0x03	EvtVnd3	unit32	Bit field		Vendor defined event flags (bits 64–95)	
40130	40131	2	R	0x03	EvtVnd4	unit32	Bit field		Vendor defined event flags (bits 96–127)	

1) see SunSpec State Codes

2) see Fronius State Codes

3) Detailed information and lists can be downloaded in various formats (xlsx, csv, json) from the Fronius website:

[www.fronius.com – Solar Energy – Info & Support – Documents & Downloads – Third-Party Downloads](http://www.fronius.com – Solar Energy – Info & Support – Documents & Downloads – Third-Party Downloads)

**Integer Display – “int+SF”**

<b>Start</b>	<b>End</b>	<b>Size</b>	<b>R/W</b>	<b>Function codes</b>	<b>Name</b>	<b>Type</b>	<b>Units</b>	<b>Scale factor</b>	<b>Description</b>	<b>Range of values</b>
40070	40070	1	R	0x03	ID	uint16			Uniquely identifies this as a SunSpec Inverter Modbus Map 101: single phase 102: split phase 103: three phase	101, 102, 103
40071	40071	1	R	0x03	L	uint16	Registers		Length of inverter model block	50
40072	40072	2	R	0x03	A	uint16	A	A_SF	AC total current value	
40073	40073	2	R	0x03	AphA	uint16	A	A_SF	AC phase A current value	
40074	40074	2	R	0x03	AphB	uint16	A	A_SF	AC phase B current value	
40075	40075	2	R	0x03	AphC	uint16	A	A_SF	AC phase C current value	
40076	40076	2	R	0x03	A_SF	sunssf			AC current scale factor	
40077		1	R	0x03	PPVphAB	uint16	V	V_SF	AC voltage phase AB value	
40078	40078	1	R	0x03	PPVphBC	uint16	V	V_SF	AC voltage phase BC value	
40079	40079	1	R	0x03	PPVphCA	uint16	V	V_SF	AC voltage phase CA value	
40080	40080	1	R	0x03	PhVphA	uint16	V	V_SF	AC voltage phase-A-to-neutral value	
40081	40081	1	R	0x03	PhVphB	uint16	V	V_SF	AC voltage phase-B-to-neutral value	
40082	40082	1	R	0x03	PhVphC	uint16	V	V_SF	AC voltage phase-C-to-neutral value	
40083	40083	1	R	0x03	V_SF	sunssf			AC voltage scale factor	
40084	40084	1	R	0x03	W	int16	W	W_SF	AC power value	
40085	40085	1	R	0x03	W_SF	sunssf			AC power scale factor	
40086	40086	1	R	0x03	Hz	uint16	Hz	Hz_SF	AC frequency value	
40087	40087	1	R	0x03	Hz_SF	sunssf			Scale factor	
40088	40088	1	R	0x03	VA	int16	VA	VA_SF	Apparent power	
40089	40089	1	R	0x03	VA_SF	sunssf			Scale factor	
40090	40090	1	R	0x03	VAr	int16	VAr	VAr_SF	Reactive power	
40091	40091	1	R	0x03	VAr_SF	sunssf			Scale factor	
40092	40092	1	R	0x03	PF	int16	%	PF_SF	Power factor	
40093	40093	1	R	0x03	PF_SF	sunssf			Scale factor	
40094	40095	2	R	0x03	WH	uint32	Wh	WH_SF	AC lifetime energy production	
40096	40096	1	R	0x03	WH_SF	sunssf			AC lifetime energy production scale factor	

**Integer Display – “int+SF”**

<b>Start</b>	<b>End</b>	<b>Size</b>	<b>R/W</b>	<b>Function codes</b>	<b>Name</b>	<b>Type</b>	<b>Units</b>	<b>Scale factor</b>	<b>Description</b>	<b>Range of values</b>
40097	40097	1	R	0x03	DCA	uint16	A	DCA_SF	DC current value	DC current only if one MPPT available; with multiple MPPT “not implemented”
40098	40098	1	R	0x03	DCA_SF	sunssf			Scale factor	Scale factor for DC current only if one MPPT available; with multiple MPPT “not implemented”
40099	40099	1	R	0x03	DCV	uint16	V	DCV_SF	DC voltage value	DC voltage only if one MPPT available; with multiple MPPT “not implemented”
40100	40100	1	R	0x03	DCV_SF	sunssf			Scale factor	Scale factor for DC voltage only if one MPPT available; with multiple MPPT “not implemented”
40101	40101	1	R	0x03	DCW	int16	W	DCW_SF	DC power value	Total DC power of all available MPPT
40102	40102	1	R	0x03	DCW_SF	sunssf			Scale factor	
40103	40103	1	R	0x03	TmpCab	int16	°C	Tmp_SF	Cabinet temperature	Not supported
40104	40104	1	R	0x03	TmpSnk	int16	°C	Tmp_SF	Coolant or heat sink temperature	Not supported
40105	40105	1	R	0x03	TmpTrns	int16	°C	Tmp_SF	Transformer temperature	Not supported
40106	40106	1	R	0x03	TmpOt	int16	°C	Tmp_SF	Other temperature	Not supported
40107	40107	1	R	0x03	Tmp_SF	sunssf			Scale factor	Not supported

## Integer Display – “int+SF”

EN-US

<b>Start</b>	<b>End</b>	<b>Size</b>	<b>R/W</b>	<b>Function codes</b>	<b>Name</b>	<b>Type</b>	<b>Units</b>	<b>Scale factor</b>	<b>Description</b>	<b>Range of values</b>
40108	40108	1	R	0x03	St	enum16	Enum-erated		Operating state	1)
40109	40109	1	R	0x03	StVnd	enum16	Enum-erated		Vendor defined operating state	2)
40110	40111	2	R	0x03	Evt1	uint32	Bit field		Event flags (bits 0–31)	3)
40112	40113	2	R	0x03	Evt2	uint32	Bit field		Event flags (bits 32–63)	
40114	40115	2	R	0x03	EvtVnd1	uint32	Bit field		Vendor defined event flags (bits 0–31)	
40116	40117	2	R	0x03	EvtVnd2	uint32	Bit field		Vendor defined event flags (bits 32–63)	3)
40118	40119	2	R	0x03	EvtVnd3	uint32	Bit field		Vendor defined event flags (bits 64–95)	
40120	40121	2	R	0x03	EvtVnd4	uint32	Bit field		Vendor defined event flags (bits 96–127)	

1) see SunSpec State Codes

2) see Fronius State Codes

3) Detailed information and lists can be downloaded in various formats (xlsx, csv, json) from the Fronius website:

[www.fronius.com – Solar Energy – Info & Support – Documents & Downloads – Third-Party Downloads](http://www.fronius.com – Solar Energy – Info & Support – Documents & Downloads – Third-Party Downloads)

### SunSpec State Codes

<b>Name</b>	<b>Value</b>	<b>Description</b>
I_STATUS_OFF	1	Inverter is off
I_STATUS_SLEEPING	2	Auto shutdown
I_STATUS_STARTING	3	Inverter starting
I_STATUS_MPPT	4	Inverter working normally
I_STATUS_THROTTLED	5	Power reduction active
I_STATUS_SHUTTING_DOWN	6	Inverter shutting down
I_STATUS_FAULT	7	One or more faults present, see St*or Evt* register
I_STATUS_STANDBY	8	Standby

\* Inverter model register

---

**Fronius State Codes**

Name	Value	Description
I_STATUS_OFF	1	Inverter is off
I_STATUS_SLEEPING	2	Auto shutdown
I_STATUS_STARTING	3	Inverter starting
I_STATUS_MPPT	4	Inverter working normally
I_STATUS_THROTTLED	5	Power reduction active
I_STATUS_SHUTTING_DOWN	6	Inverter shutting down
I_STATUS_FAULT	7	One or more faults present, see St*or Evt* register
I_STATUS_STANDBY	8	Standby
I_STATUS_NO_BUSINIT	9	No SolarNet communication
I_STATUS_NO_COMM_INV	10	No communication with inverter possible
I_STATUS_SN_OVERCURRENT	11	Overcurrent detected on SolarNet plug
I_STATUS_BOOTLOAD	12	Inverter is currently being updated
I_STATUS_AFCI	13	AFCI event (arc detection)

\* Inverter model register

# Nameplate model (IC120)

## General

This model corresponds to a rating plate. The following data can be read:

- ***DERType (3)***  
Type of device. The register returns the value 4 (PV device).
- ***WRtg (4)***  
Nominal power of inverter.
- ***VARtg (6)***  
Nominal apparent power of inverter.
- ***VArRtgQ1 (8) – VArRtgQ4 (11)***  
Nominal reactive power values for the four quadrants.
- ***ARtg (13)***  
Nominal current of inverter.
- ***PFRtgQ1 (15) – PFRtgQ4 (18)***  
Minimal power factor values for the four quadrants.

## Nameplate Register

Start address:

- for “float” setting: **40131**
- for “int+SF” setting: **40121**

Start offset	End offset	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
1	1	1	R	0x03	ID	uint16			A well-known value 120. Uniquely identifies this as a SunSpec Nameplate Model	120
2	2	1	R	0x03	L	uint16	Registers		Length of Nameplate Model	26
3	3	1	R	0x03	DERTyp	enum16			Type of DER device. Default value is 4 to indicate PV device	4
4	4	1	R	0x03	WRtg	uint16	W	WRtg_SF	Continuous power output capability of the inverter	
5	5	1	R	0x03	WRtg_SF	sunssf			Scale factor	1
6	6	1	R	0x03	VARtg	uint16	VA	VARtg_SF	Continuous volt-ampere capability of the inverter	
7	7	1	R	0x03	VARtg_SF	sunssf			Scale factor	1
8	8	1	R	0x03	VArRtgQ1	int16	var	VArRtg_SF	Continuous VAR capability of the inverter in quadrant 1	
9	9	1	R	0x03	VArRtgQ2	int16	var	VArRtg_SF	Continuous VAR capability of the inverter in quadrant 2	Not supported
10	10	1	R	0x03	VArRtgQ3	int16	var	VArRtg_SF	Continuous VAR capability of the inverter in quadrant 3	Not supported

Start offset	End offset	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
11	11	1	R	0x03	VArRtgQ4	int16	var	VArRtg_SF	Continuous VAR capability of the inverter in quadrant 4	
12	12	1	R	0x03	VArRtg_SF	sunssf			Scale factor	1
13	13	1	R	0x03	ARtg	uint16	A	ARtg_SF	Maximum RMS AC current level capability of the inverter	
14	14	1	R	0x03	ARtg_SF	sunssf			Scale factor	-2
15	15	1	R	0x03	PFRtgQ1	int16	cos()	PFRtg_SF	Minimum power factor capability of the inverter in quadrant 1	
16	16	1	R	0x03	PFRtgQ2	int16	cos()	PFRtg_SF	Minimum power factor capability of the inverter in quadrant 2	Not supported
17	17	1	R	0x03	PFRtgQ3	int16	cos()	PFRtg_SF	Minimum power factor capability of the inverter in quadrant 3	Not supported
18	18	1	R	0x03	PFRtgQ4	int16	cos()	PFRtg_SF	Minimum power factor capability of the inverter in quadrant 4	
19	19	1	R	0x03	PFRtg_SF	sunssf			Scale factor	-3
20	20	1	R	0x03	WHRtg	uint16	Wh	WHRtg_SF	Nominal energy rating of storage device	*
21	21	1	R	0x03	WHRtg_SF	sunssf			Scale factor	0*
22	22	1	R	0x03	AhrRtg	uint16	AH	AhrRtg_SF	The usable capacity of the battery. Maximum charge minus minimum charge from a technology perspective (amp-hour capacity rating)	Not supported
23	23	1	R	0x03	AhrRtg_SF	sunssf			Scale factor for amp-hour rating	Not supported
24	24	1	R	0x03	MaxChaRte	uint16	W	MaxChaRte_SF	Maximum rate of energy transfer into the storage device	*
25	25	1	R	0x03	MaxChaRte_SF	sunssf			Scale factor	0*
26	26	1	R	0x03	MaxDisChaRte	uint16	W	MaxDisChaRte_SF	Maximum rate of energy transfer out of the storage device	*
27	27	1	R	0x03	MaxDisChaRte_SF	sunssf			Scale factor	0*
28	28	1	R	0x03	Pad				Pad register	

\* Only supported by Fronius Hybrid inverters

# Basic Settings Model (IC121)

**Basic Settings Register**

Start address:

- for "float" setting: **40159**
- for "int+SF" setting: **40149**

Start offset	End offset	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
1	1	1	R	0x03	ID	uint16			A well-known value 121. Uniquely identifies this as a SunSpec Basic Settings Model	121
2	2	1	R	0x03	L	uint16	Registers		Length of Basic Settings Model	30
3	3	1	R	0x03	WMax	uint16	W	WMax_SF	Setting for maximum power output. Default to WRtg	
4	4	1	RW	0x03 0x06 0x10	VRef	uint16	V	VRef_SF	Voltage at the PCC	
5	5	1	RW	0x03 0x06 0x10	VRefOfs	int16	V	VRefOfs_SF	Offset from PCC to inverter	
6	6	1	R	0x03	VMax	uint16	V	VMin-Max_SF	Currently not supported	
7	7	1	R	0x03	VMin	uint16	V	VMin-Max_SF	Currently not supported	
8	8	1	R	0x03	VAMax	unit16	VA	VAMax_SF	Setpoint for maximum apparent power. Default to VARtg	
9	9	1	R	0x03	VARMaxQ1	int16	var	VAR-Max_SF	Setting for maximum reactive power in quadrant 1. Default to VArRtgQ1	
10	10	1	R	0x03	VARMaxQ2	int16	var	VAR-Max_SF	Setting for maximum reactive power in quadrant 2. Default to VArRtgQ2	Not supported
11	11	1	R	0x03	VARMaxQ3	int16	var	VAR-Max_SF	Setting for maximum reactive power in quadrant 3. Default to VArRtgQ3	Not supported
12	12	1	R	0x03	VARMaxQ4	int16	var	VAR-Max_SF	Setting for maximum reactive power in quadrant 4. Default to VArRtgQ4	
13	13	1	R	0x03	WGra	uint16	% WMax/min	WGra_SF	Default ramp rate of change of active power due to command or internal action	Not supported

<b>Start offset</b>	<b>End offset</b>	<b>Size</b>	<b>R/W</b>	<b>Function codes</b>	<b>Name</b>	<b>Type</b>	<b>Units</b>	<b>Scale factor</b>	<b>Description</b>	<b>Range of values</b>
14	14	1	R	0x03	PFMinQ1	int16	cos()	PFMin_SF	Setpoint for minimum power factor value in quadrant 1. Default to PFRtgQ1	
15	15	1	R	0x03	PFMinQ2	int16	cos()	PFMin_SF	Setpoint for minimum power factor value in quadrant 2. Default to PFRtgQ2	Not supported
16	16	1	R	0x03	PFMinQ3	int16	cos()	PFMin_SF	Setpoint for minimum power factor value in quadrant 3. Default to PFRtgQ3	Not supported
17	17	1	R	0x03	PFMinQ4	int16	cos()	PFMin_SF	Setpoint for minimum power factor value in quadrant 4. Default to PFRtgQ4	
18	18	1	R	0x03	VArAct	enum16			VAR action on change between charging and discharging: 1=switch 2=maintain VAR characterization	Not supported
19	19	1	R	0x03	ClcTotVA	enum16			Calculation method for total apparent power. 1=vector 2=arithmetic	Not supported
20	20	1	R	0x03	MaxRmpRte	uint16	% WGra	Max-RmpRte_SF	Setpoint for maximum ramp rate as percentage of nominal maximum ramp rate. This setting will limit the rate that watts delivery to the grid can increase or decrease in response to intermittent PV generation	Not supported
21	21	1	R	0x03	ECPNomHz	uint16	Hz	ECP-Nom-Hz_SF	Setpoint for nominal frequency at the ECP	Not supported
22	22	1	R	0x03	ConnPh	enum16			Identity of connected phase for single phase inverters. A=1 B=2 C=3	Not supported
23	23	1	R	0x03	WMax_SF	sunssf			Scale factor for maximum power output	1
24	24	1	R	0x03	VRef_SF	sunssf			Scale factor for voltage at the PCC	0
25	25	1	R	0x03	VRefOfs_SF	sunssf			Scale factor for offset voltage	0
26	26	1	R	0x03	VMinMax_SF	sunssf			Scale factor for min/max voltages	0
27	27	1	R	0x03	VAMax_SF	sunssf			Scale factor for voltage at the PCC	1

Start offset	End offset	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
28	28	1	R	0x03	VARMax_SF	sunssf			Scale factor for reactive power	1
29	29	1	R	0x03	WGra_SF	sunssf			Scale factor for default ramp rate	Not supported
30	30	1	R	0x03	PFMin_SF	sunssf			Scale factor for minimum power factor	-3
31	31	1	R	0x03	MaxRmpRte_SF	sunssf			Scale factor for maximum ramp percentage	Not supported
32	32	1	R	0x03	ECPNom-Hz_SF	sunssf			Scale factor for nominal frequency	Not supported

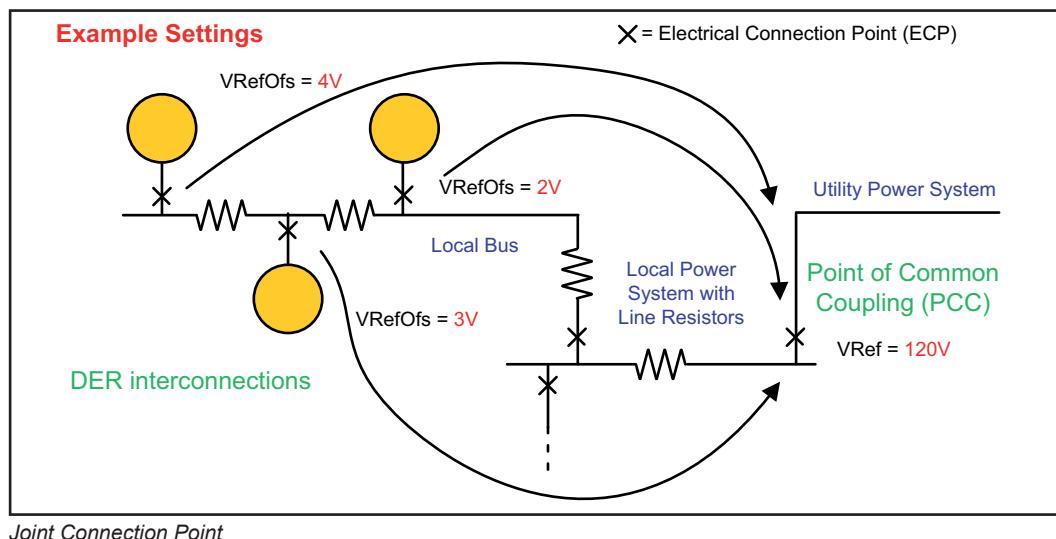
## Reference Voltage

### VRef (4)

The reference voltage is the voltage at the joint connection point where the local grid is connected to the public grid. The reference voltage is the same as the inverter's nominal voltage.

=> See figure "Joint Connection Point."

The value is given in volts in the range of 0 (0x0000) to 400 (0x0190).



## Deviation from Reference Voltage

### VRefOfs (5)

Depending on the wiring of the local grid, there may be a deviation from the reference voltage at the point where each individual inverter is connected to the local grid (see "Joint connection point" diagram).

The value is given in volts in the range of -20 (0xFFEC) to 20 (0x0014).

# Extended Measurements & Status Model (IC122)

---

<b>General</b>	This model provides some additional measurement and status values which the normal inverter model does not cover:
-	<b>PVConn (3)</b> This bit field displays the inverter's status <ul style="list-style-type: none"> <li>- Bit 0: Connected</li> <li>- Bit 1: Responsive</li> <li>- Bit 2: Operating (inverter feeds energy in)</li> </ul>
-	<b>ECPConn (5)</b> This register displays the status of connection to the grid <ul style="list-style-type: none"> <li>- <i>ECPConn</i> = 1: Inverter is currently feeding power into the grid</li> <li>- <i>ECPConn</i> = 0: Inverter is not feeding power into the grid</li> </ul>
-	<b>ActWH (6–9)</b> Active energy meter
-	<b>StActCtl (36–37)</b> Bit field for currently active inverter modes <ul style="list-style-type: none"> <li>- Bit 0: Power reduction (FixedW; corresponds to WMaxLimPct specification)</li> <li>- Bit 1: Constant reactive power specification (FixedVAR; corresponds to VArMax-Pct)</li> <li>- Bit 2: Setting a constant power factor (FixedPF; corresponds to OutPFSet)</li> </ul>
-	<b>TmSrc (38–41)</b> Source for the time synchronization, the register returns the string "RTC"
-	<b>Tms (42–43)</b> Current time and date of the RTC The seconds are specified from January 1, 2000 00:00 (UTC) to the current time.

---

## Extended Measurements & Status Register

Start address:

- for "float" setting: **40191**
- for "int+SF" setting: **40181**

Start offset	End offset	Size	RW	Function codes	Name	Type	Units	Scale factor	Description	Range of values
1	1	1	R	0x03	ID	uint16			A well-known value 122. Uniquely identifies this as a SunSpec Measurements_Status Model	122
2	2	1	R	0x03	L	uint16	Registers		Length of Measurements_Status Model	44

Start offset	End offset	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
3	3	1	R	0x03	PVConn	bitfield16			PV inverter present/available status. Enumerated value  Connected Available Operating Test	Bit 0 = 1 Bit 1 = 1 Bit 2 = 1 Bit 3 = 1
4	4	1	R	0x03	StorConn	bitfield16			Storage inverter present/available status. Enumerated value  Connected Available Operating	Bit 0 = 1* Bit 1 = 1* Bit 2 = 1*
5	5	1	R	0x03	ECPConn	bitfield16			ECP connection status  Connected	Bit 0 = 1
6	9	4	R	0x03	ActWh	acc64	Wh		AC lifetime active (real) energy output	
10	13	4	R	0x03	ActVAh	acc64	VAh		AC lifetime apparent energy output	Not supported
14	17	4	R	0x03	ActVArhQ1	acc64	varh		AC lifetime reactive energy output in quadrant 1	Not supported
18	21	4	R	0x03	ActVArhQ2	acc64	varh		AC lifetime reactive energy output in quadrant 2	Not supported
22	25	4	R	0x03	ActVArhQ3	acc64	varh		AC lifetime negative energy output in quadrant 3	Not supported
26	29	4	R	0x03	ActVArhQ4	acc64	varh		AC lifetime reactive energy output in quadrant 4	Not supported
30	30	1	R	0x03	VArAval	int16	var	VArAval_SF	Number of VARs available without impacting watts output	Not supported
31	31	1	R	0x03	VArAval_SF	sunssf			Scale factor for available VARs	Not supported
32	32	1	R	0x03	WAval	uint16	W	WAval_SF	Number of watts available	Not supported
33	33	1	R	0x03	WAval_SF	sunssf			Scale factor for available watts	Not supported
34	35	2	R	0x03	StSetLimMsk	bitfield32			Bit mask indicating set-point limit(s) reached. Bits are persistent and must be cleared by the controller	Not supported

<b>Start offset</b>	<b>End offset</b>	<b>Size</b>	<b>R/W</b>	<b>Function codes</b>	<b>Name</b>	<b>Type</b>	<b>Units</b>	<b>Scale factor</b>	<b>Description</b>	<b>Range of values</b>
36	37	2	R	0x03	StActCtl	bitfield32			Bit mask indicating which inverter controls are currently active FixedW FixedVAR FixedPF	Bit 0 = 1 Bit 1 = 1 Bit 2 = 1
38	41	4	R	0x03	TmSrc	string			Source of time synchronization	RTC
42	43	2	R	0x03	Tms	uint32	Secs		Seconds since 01-01-2000 00:00 UTC	
44	44	1	R	0x03	RtSt	bitfield16			Bit mask indicating which voltage ride through modes are currently active	0
45	45	1	R	0x03	Riso	uint16	Ohm	Riso_SF	Isolation resistance	Not supported
46	46	1	R	0x03	Riso_SF	int16			Scale factor for isolation resistance	Not supported

\* Only supported by Fronius Hybrid inverters

# Immediate Control Model (IC123)

EN-US

<b>General</b>		The immediate controls can be used to make the following settings on the inverter:								
		<ul style="list-style-type: none"> <li>- deactivation of inverter's grid power feed operation (standby)</li> <li>- constant reduction of output power</li> <li>- specification of a constant power factor</li> <li>- specification of a constant relative reactive power</li> </ul>								
<b>Immediate Controls Register</b>		Start address: <ul style="list-style-type: none"> <li>- for "float" setting: <b>40237</b></li> <li>- for "int+SF" setting: <b>40227</b></li> </ul>								
Start offset	End offset	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
1	1	1	R	0x03	ID	uint16			A well-known value 123. Uniquely identifies this as a SunSpec Immediate Controls Model	123
2	2	1	R	0x03	L	uint16	Registers		Length of Immediate Controls Model	24
3	3	1	RW	0x03 0x06 0x10	Conn_WinTms	uint16	Secs		Time window for connect/disconnect	0–300
4	4	1	RW	0x03 0x06 0x10	Conn_RvrTms	uint16	Secs		Timeout period for connect/disconnect	0–28800
5	5	1	RW	0x03 0x06 0x10	Conn	enum16			Enumerated value. Connection control	
									Disconnected	0
									Connected	1
6	6	1	RW	0x03 0x06 0x10	WMaxLimPct	uint16	% WMax	WMax-LimPct_SF	Set power output to specified level	
7	7	1	RW	0x03 0x06 0x10	WMax-LimPct_Win-Tms	uint16	Secs		Time window for power limit change	0–300
8	8	1	RW	0x03 0x06 0x10	WMax-LimPct_RvrT-ms	uint16	Secs		Timeout period for power limit	0–28800
9	9	1	R	0x03	WMax-LimPct_RmpT-ms	uint16	Secs		Ramp time for moving from current setpoint to new setpoint	Not supported
10	10	1	RW	0x03 0x06 0x10	WMaxLim_Ena	enum16			Enumerated value. Throttle enable/disable control	
									Disabled	0
									Enabled	1

Start offset	End offset	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
11	11	1	RW	0x03 0x06 0x10	OutPFSet	int16	cos()	OutPF- Set_SF	Set power factor to specific value – cosine of angle	
12	12	1	RW	0x03 0x06 0x10	OutPFSet_Win- Tms	uint16	Secs		Time window for power factor change	0–300
13	13	1	RW	0x03 0x06 0x10	OutPF- Set_RvrTms	uint16	Secs		Timeout period for power factor	0–28800
14	14	1	R	0x03	OutPF- Set_RmpTms	uint16	Secs		Ramp time for moving from current setpoint to new setpoint	Not supported
15	15	1	RW	0x03 0x06 0x10	OutPFSet_Ena	enum16			Enumerated value. Fixed power factor enable/disable control	
									Disabled Enabled	0 1
16	16	1	R	0x03	VArWMaxPct	int16	% WMax	VArW- Max- Pct_SF	Reactive power in percent of WMax	Not supported
17	17	1	RW	0x03 0x06 0x10	VArMaxPct	int16	% VAr- Max	VArPct_ SF	Reactive power in percent of VArMax	
18	18	1	R	0x03	VArAvalPct	int16	% VAr- Aval	VArPct_ SF	Reactive power in percent of VArAval	Not supported
19	19	1	RW	0x03 0x06 0x10	VArPct_Win- Tms	uint16	Secs		Time window for VAR limit change	0–300
20	20	1	R	0x03	VArPct_RvrT- ms	uint16	Secs		Timeout period for VAR limit	0–28800
21	21	1	RW	0x03 0x06 0x10	VArPct_RmpT- ms	uint16	Secs		Ramp time for moving from current setpoint to new setpoint	Not supported
22	22	1	R	0x03	VArPct_Mod	enum16			Enumerated value. VAR limit mode	2: VAR limit as a % of VArMax
23	23	1	RW	0x03 0x06 0x10	VArPct_Ena	enum16			Enumerated value. Fixed VAR enable/disable control	
									Disabled Enabled	0 1
24	24	1	R	0x03	WMax- LimPct_SF	sunssf			Scale factor for power output percent	-2
25	25	1	R	0x03	OutPFSet_SF	sunssf			Scale factor for power factor	-3
26	26	1	R	0x03	VArPct_SF	sunssf			Scale factor for reactive power	0

<b>Standby</b>	<p><b><i>Conn_WinTms</i> (3) to <i>Conn</i> (5)</b></p> <p>These registers are used to control the standby mode (no grid power feed operation) of the inverter.</p> <p><b><i>Conn_WinTms</i> (3) and <i>Conn_RvrtTms</i> (4)</b></p> <p>These registers can be used to control the inverter's time response. =&gt; See section "Time Response of the Supported Operating Modes". 0 is set as the default for all registers.</p> <p><b><i>Conn</i> (5)</b></p> <p>Register <i>Conn</i> indicates whether or not the inverter is currently feeding power into the grid (0 = standby, 1 = grid power feed operation).</p> <ul style="list-style-type: none"> <li>- In order to switch the inverter to standby, enter the value 0 into this register.</li> <li>- In order to reactivate the inverter, enter the value 1 into this register.</li> </ul> <p> <b>NOTE!</b> To find out whether or not the inverter is feeding power into the grid, you can also use the ECPConn register and check the extended measurements and status model.</p>
<b>Power reduction</b>	<p><b><i>WMaxLimPct</i> (6) to <i>WMaxLim_Ena</i> (10)</b></p> <p>These registers can be used to set an output power reduction in the inverter.</p> <p><b><i>WMaxLimPct</i> (6)</b></p> <p>In register <i>WMaxLimPct</i> you can enter values between 0% and 100%. Depending on the inverter's software version, values below 10 may force the inverter into standby (no grid power feed operation). The values limit the device's maximum possible output power and therefore may not necessarily affect the real-time power.</p> <p><b>IMPORTANT!</b> Observe the scale factor for this register. Further information can be found at: <a href="http://sunspec.org/wp-content/uploads/2015/06/SunSpec-Information-Models-12041.pdf">http://sunspec.org/wp-content/uploads/2015/06/SunSpec-Information-Models-12041.pdf</a></p> <p><b><i>WMaxLimPct_WinTms</i> (7), <i>WMaxLimPct_RvrtTms</i> (8)</b></p> <p>These registers can be used to control the inverter's time response for this operating mode. =&gt; See section "Time Response of the Supported Operating Modes." 0 is set as the default for all registers.</p> <p><b><i>WMaxLim_Ena</i> (10)</b></p> <p>Used to start and end this operating mode</p> <ul style="list-style-type: none"> <li>- Enter value 1 into register <i>WMaxLim_Ena</i> = start operating mode</li> <li>- Enter value 0 into register <i>WMaxLim_Ena</i> = end operating mode</li> </ul> <p> <b>NOTE!</b> Proceed as follows to change values when an operating mode is active (e.g., when setting a different power limit or return time):       <ul style="list-style-type: none"> <li>- Enter the new value into the relevant register</li> <li>- Restart the operating mode using register <i>WMaxLim_Ena</i></li> </ul> </p> <p><b>Example: Setting a Power Reduction</b></p> <ol style="list-style-type: none"> <li><b>1</b> Enter the value for the output power reduction in register <i>WMaxLimPct</i> (e.g., 30 for 30%).</li> <li><b>2</b> As an option, you can set the start and return time using registers <i>WMaxLimPct_WinTms</i> and <i>WMaxLimPct_RvrtTms</i>.</li> <li><b>3</b> Start the operating mode by entering 1 in register <i>WMaxLim_Ena</i>.</li> </ol>

**IMPORTANT!** Observe the scale factor for this register.

Further information can be found at:

<http://sunspec.org/wp-content/uploads/2015/06/SunSpec-Information-Models-12041.pdf>

**Example:  
Changing the Re-  
turn Time When  
Power Reduction  
Has Been Activat-  
ed**

If the power reduction was originally started using  $WMaxLimPct_RvrtTms = 0$ , the operating mode must be manually deactivated.

- 1** Set  $WMaxLimPct_RvrtTms$  to 30, for example
- 2** Apply the change by entering 1 in register  $WMaxLim_Ena$ 
  - The operating mode is automatically deactivated after 30 seconds.

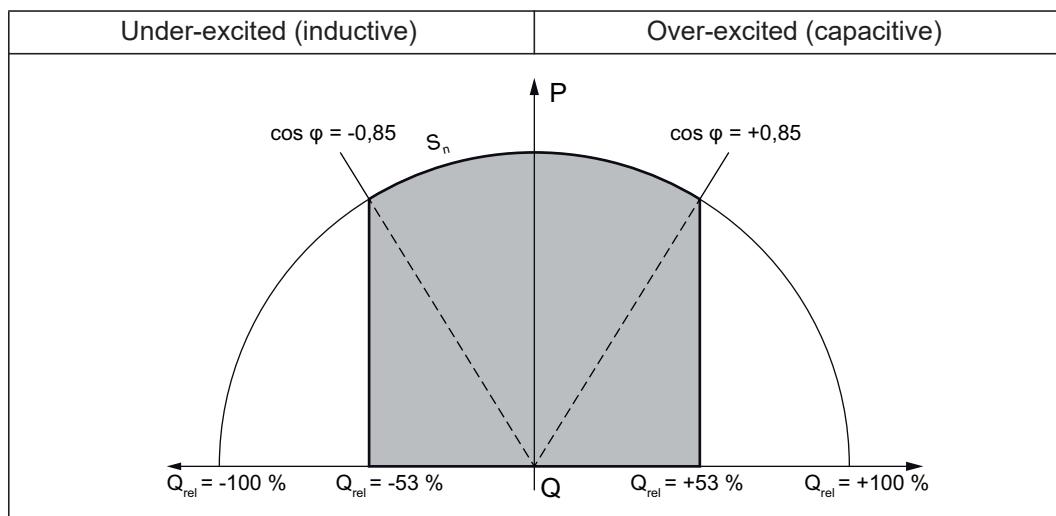
**Effects of Reac-  
tive Power Spec-  
ifications on  
Effective Power**

In principle, reactive power operation is limited by the maximum output current (the maximum apparent power) and by the operative reactive power limit of the inverter:

- Fronius IG Plus, CL cos phi = 0.85, VArrel = 100%
- Fronius Galvo cos phi = 0.85, VArrel = 53%
- Fronius Symo cos phi = 0.7, VArrel = 71%.

 **NOTE!** Due to the current technical conditions, only a cos phi up to a maximum of  $\pm 0.85$  can be specified per Modbus. In some circumstances, however, VAr<sub>rel</sub> specifications may demand a lower value.

The following diagram shows the possible working area of the inverter. All valid operating points defined by effective power P and reactive power Q are within the gray area.



Reactive Power and Power Factor

**Legend:**

W	Power	VAr <sub>max</sub>	Nominal reactive power
$W_{\max}$	Nominal power	VAr <sub>rel</sub>	Relative reactive power
VAr	Reactive power		(VAr/VArmax)

## Constant Power Factor

### ***OutPFSet (11) to OutPFSet\_Ena (15)***

These registers can be used to set a constant power factor in the inverter.

#### ***OutPFSet (11)***

- In register *OutPFSet* it is possible to enter both positive and negative values for the power factor.
- The values must be scaled up by a factor of 1000 , e.g., power factor 0.95 = register value 950.
- The lowest possible values depend on the inverter type and can be found in the Nameplate Model.



**NOTE!** The power factor value must be entered with the correct sign => see section "Sign Convention for the Power Factor"

- positive for under-excited
- negative for over-excited.

### ***OutPFSet\_WinTms (12), OutPFSet\_RvrtTms (13)***

These registers can be used to control the inverter's time response for this operating mode.

=> See section "Time Response of the Supported Operating Modes".

0 is set as the default for all registers.

#### ***OutPFSet\_Ena (15)***

Used to start and end this operating mode

- Enter value 1 into register *OutPFSet\_Ena* = start operating mode
- Enter value 0 into register *OutPFSet\_Ena* = end operating mode.



**NOTE!** Proceed as follows to change values when an operating mode is active (e.g., when setting a different power factor or return time):

- Enter the new value into the relevant register.
- Restart the operating mode using register *OutPFSet\_Ena*.

---

## Example: Setting a Constant Power Factor

- 1** Enter the power factor value in register *OutPFSet* (e.g., 950 for 0.95).
- 2** As an option, you can set the start and return time using registers *OutPFSet\_WinTms* and *OutPFSet\_RvrtTms*.
- 3** Start the operating mode by entering 1 in register *OutPFSet\_Ena*.

---

**Constant Relative  
Reactive Power****VArMaxPct (17) to VArPct\_Ena (23)**

These registers can be used to set on the inverter a constant value for the reactive power to be produced by the inverter.

**VArMaxPct (17)**

- Used to set a value for constant reactive power.
- The minimum and maximum limits depend on the type of inverter.



**NOTE!** In practical operation, the reactive power that is actually available is specified by the inverter's operating limits. For this reason, the reactive power specification can only be reached if enough effective power is fed into the grid. If too little effective power is fed into the grid, the inverter will operate at its operating limit.

**VArPct\_WinTms (19), VArPct\_RvrtTms (20)**

These registers can be used to control the inverter's time response for this operating mode.

=> See section "Time Response of the Supported Operating Modes".

0 is set as the default for all registers.

**VArPct\_Mod (22)**

- This register cannot be changed.
- It returns the (currently) supported operating mode.  
Reactive power as a percentage of the maximum possible reactive power.

**VArPct\_Ena (23)**

Used to start and end this operating mode

- Enter value 1 into register VArPct\_Ena = start operating mode
- Enter value 0 into register VArPct\_Ena = end operating mode.



**NOTE!** Proceed as follows to change values when an operating mode is active (e.g., when setting a different reactive power value or return time):

- Enter the new value into the relevant register.
- Restart the operating mode using register VArPct\_Ena.

---

**Example:  
Setting Constant  
Reactive Power**

**1** Enter the relative reactive power value in register VArMaxPct  
(e.g., 80 for 80%).

**2** As an option, you can set the start and return time using registers VArPct\_WinTms and VArPct\_RvrtTms.

**3** Start the operating mode by entering 1 in register VArPct\_Ena.

# Basic Storage Control Model (IC124)

<b>General</b>	<p>This model is only available for Fronius Hybrid inverters.</p> <p>The Basic Storage Control Model can be used to make the following settings on the inverter:</p> <ul style="list-style-type: none"> <li>- Setting a power window within which the charge/discharge capacity of the energy storage may fluctuate.</li> <li>- Setting a minimum charge level that the energy storage must not fall below.</li> <li>- Permitting/preventing grid charging of the energy storage.</li> </ul> <p> <b>NOTE!</b> All specifications are to be considered recommendations. The inverter may deviate from the specifications if this is necessary for operational safety reasons.</p>
<b>Information Provided</b>	<p>The Basic Storage Control Model provides the following read-only information:</p> <p><b>WChaMax</b></p> <ul style="list-style-type: none"> <li>- If energy storage is available, this register feeds back the baseline value for the registers OutWRte and InWRt. <math>WChaMax := \max(\text{MaxChaRte}, \text{MaxDisChaRte})</math></li> <li>- If energy storage is not available, the register feeds back a value of 0.</li> </ul> <p><b>ChaState</b></p> <ul style="list-style-type: none"> <li>- Energy storage charge level in %: <math>\text{Estimated\_Capacity\_Remaining [Wh]} / \text{Estimated\_Capacity\_Maximum [Wh]}</math></li> </ul> <p><b>ChaSt</b></p> <p>Energy storage operating status</p> <ul style="list-style-type: none"> <li>- OFF: Energy storage is not available</li> <li>- EMPTY: Energy storage is currently fully discharged</li> <li>- DISCHARGING: Energy storage is in the process of being discharged</li> <li>- CHARGING: Energy storage is in the process of being charged</li> <li>- FULL: Energy storage is currently fully charged</li> <li>- HOLDING: Energy storage is currently neither charged nor discharged</li> </ul>
<b>Power Window Specifications</b>	<p>The following examples assume that WchaMax = 3300 W.</p> <p>The following applies for the resulting power windows:</p> <ul style="list-style-type: none"> <li>- Negative power values indicate that the energy storage is charging</li> <li>- Positive values indicate that the energy storage is discharging</li> </ul> <p> <b>NOTE!</b> The values in the following examples must be scaled according to their scale factors in the specified scale registers after reading and before writing.</p> <p><b>Example 1: Only permit energy storage charging</b></p> <p>This behavior can be achieved by limiting the maximum discharge capacity to 0% =&gt; results in window [-3300 W, 0 W]</p> <ul style="list-style-type: none"> <li>- OutWRte = 0% (set discharge limit of WchaMax to 0%)</li> <li>- StorCtl_Mod = 1 (activates discharge limit, bit pattern: 01)</li> <li>- InWRte is not relevant in this case</li> </ul> <p><b>Example 2: Only permit energy storage discharging</b></p>

This behavior can be achieved by limiting the maximum charge capacity to 0% => results in window [0 W, 3300 W]

- InWRte = 0% (set charge limit of WchaMax to 0%)
- StorCtl\_Mod = 2 (bit 1 activates charge limit, bit pattern: 10)
- OutWRte is not relevant in this case

#### **Example 3: Do not permit charging or discharging**

This behavior can be achieved by limiting the maximum charge capacity to 0% and the maximum discharge capacity to 0%

=> results in window [0 W, 0 W]

- InWRte = 0% (set charge limit of WchaMax to 0%)
- OutWRte = 0% (set discharge limit of WchaMax to 0%)
- StorCtl\_Mod = 3 (activate both limit values, bit pattern: 11)

#### **Example 4: Charging and discharging with maximum 50% of the nominal power**

This behavior can be achieved by limiting the maximum charge capacity to 50% and the maximum discharge capacity to 50%

=> results in window [-1650 W, 1650 W]

- InWRte = 50% (set charge limit of WchaMax to 50%)
- OutWRte = 50% (set discharge limit of WchaMax to 50%)
- StorCtl\_Mod = 3 (activate both limit values, bit pattern: 11)

#### **Example 5: Charging in the range of 50% to 75% of the nominal power**

This behavior can be achieved by limiting the maximum charge capacity to 75% and the maximum discharge capacity to -50%

=> results in window [1650 W, 2475 W]

- InWRte = 75% (set charge limit of WchaMax to 75%)
- OutWRte = -50% (set discharge limit of WchaMax to -50%)
- StorCtl\_Mod = 3 (activate both limit values, bit pattern: 11)

#### **Example 6: Discharging with 50% of the nominal power**

This behavior can be achieved by limiting the maximum charge capacity to -50% and the maximum discharge capacity to 50%

=> results in window [-1650 W, -1650 W]

- InWRte = -50% (set charge limit of WchaMax to -50%)
- OutWRte = 50% (set discharge limit of WchaMax to 50%)
- StorCtl\_Mod = 3 (activate both limit values, bit pattern: 11)

#### **Example 7: Charging with 50% to 75% of the nominal power**

This behavior can be achieved by limiting the maximum charge capacity to 75% and the maximum discharge capacity to -50%

=> results in window [1650 W, 2475 W]

- InWRte = 75% (set charge limit of WchaMax to 75%)
- OutWRte = -50% (set discharge limit of WchaMax to -50%)
- StorCtl\_Mod = 3 (activate both limit values, bit pattern: 11)

#### **Example 8: Charging with 50% to 100% of the nominal power**

This behavior can be achieved by limiting the maximum discharge capacity to -50% => results in window [1650 W, 3300 W]

- OutWRte = -50% (set discharge limit of WchaMax to -50%)
- StorCtl\_Mod = 1 (activates discharge limit, bit pattern: 01)
- InWRte is not relevant in this case

**Setting the Minimum Charge Level** By setting register MinRsvPct, a minimum state of charge of the energy storage can be set. For example, by setting MinRsvPct to 20%, a reserve of 20% of the state of charge can be reserved that the memory should not fall below.

**Charging the Energy Storage via the Grid** The ChaGriSet register can be used to allow or prevent inverter storage charging via the grid.

**Basic Storage Controls Register** Start address:  
 - for "float" setting: **40313**  
 - for "int+SF" setting: **40303**

Start offset	End offset	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
1	1	1	R	0x03	ID	uint16			A well-known value 124. Uniquely identifies this as a SunSpec Basic Storage Controls Model	124
2	2	1	R	0x03	L	uint16	Registers		Length of Basic Storage Controls	24
3	3	1	R	0x03	WchaMax	uint16	W	WCha-Max-SF	Setpoint for maximum charge. *	

\* Additional Fronius description:

Reference value for maximum charge and discharge. Multiply this value by InWRte to define maximum charging and OutWRte to define maximum discharging. Every rate between these two limits is allowed. Note that InWRte and OutWRte can be negative to define ranges for charging and discharging only.

4	4	1	R	0x03	WchaGra	uint16	% WCha-Max/sec	WChaD isCha-Gra_SF	Setpoint for maximum charging rate. Default is MaxChaRte	100
5	5	1	R	0x03	WdisChaGra	unit16	% WCha-Max/sec	WChaD isCha-Gra_SF	Setpoint for maximum discharge rate. Default is MaxDisChaRte	100
6	6	1	RW	0x03 0x06 0x10	StorCtl_Mod	bitfield16	VA	WMax-LimPct_SF	Activate storage control mode. Bit field value. **	Bit 0: charge  Bit 1: discharge

\*\* Additional Fronius description:

Set charge bit to enable charge limit, set discharge bit to enable discharge limit, set both bits to enable both limits.

Start offset	End offset	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
7	7	1	R	0x03	VAChaMax	uint16	VA	VACha-Max_SF	Setpoint for maximum charging VA	Not supported
8	8	1	RW	0x03 0x06 0x10	MinRsvPct	uint16	% WCh-Max	MinRsvPct_SF	Setpoint for minimum reserve for storage as a percentage of the nominal maximum storage	
9	9	1	R	0x03	ChaState	uint16	% Ahr-Rtg	ChaState_SF	Currently available energy as a percent of the capacity rating	
10	10	1	R	0x03	StorAval	uint16	Ah	StorAval_SF	State of charge (ChaState) minus storage reserve (MinRsvPct) times capacity rating (AhrRtg).	Not supported
11	11	1	R	0x03	InBatV	uint16	V	InBatV_SF	Internal battery voltage	Not supported
12	12	1	R	0x03	ChaSt	enum16			Charge status of storage device. Enumerated value	1: OFF 2: EMPTY 3: DIS-CHARGING 4: CHARGING 5: FULL 6: HOLDING 7: TESTING
13	13	1	RW	0x03 0x06 0x10	OutWRte	int16	% WCh-Max	InOut-WRte_SF	Percent of max. discharge rate.  ***	

\*\*\* Additional Fronius description:

Defines the maximum discharge rate (discharge limit). Default is 100%

14	14	1	RW	0x03 0x06 0x10	InWRte	int16	% WCh-Max	InOut-WRte_SF	Percent of max charging rate.  ****	
----	----	---	----	----------------------	--------	-------	-----------	---------------	-------------------------------------------	--

\*\*\*\* Additional Fronius description:

Defines the maximum charge rate (charge limit). Default is 100%

15	15	1	R	0x03	InOut-WRte_WinTms	uint16	Secs		Time window for charge/discharge rate change	Not supported
16	16	1	R	0x03	InOut-WRte_RvrTms	uint16	Secs		Timeout period for charge/discharge rate	Not supported
17	17	1	R	0x03	InOut-WRte_RmpTm s	uint16	Secs		Ramp time for moving from current setpoint to new setpoint	Not supported

Start offset	End offset	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
18	18	1	RW	0x03 0x06 0x10	ChaGriSet	enum16			Setpoint to enable/disable charging from grid  PV (charging from grid disabled)  GRID (charging from grid enabled)	0 0 1
19	19	1	R	0x03	WchaMax_SF	sunssf			Scale factor for maximum charge	0
20	20	1	R	0x03	WchaDisCha-Gra_SF	sunssf			Scale factor for maximum charge and discharge rate	0
21	21	1	R	0x03	VAChaMax_SF	sunssf			Scale factor for maximum charging VA	Not supported
22	22	1	R	0x03	MinRsvPct_SF	sunssf			Scale factor for minimum reserve percentage	-2
23	23	1	R	0x03 0x06 0x10	ChaState_SF	sunssf			Scale factor for available energy percent	-2
24	24	1	R	0x03	StorAval_SF	sunssf			Scale factor for state of charge	-2
25	25	1	R	0x03	InBatV_SF	sunssf			Scale factor for battery voltage	-2
26	26	1	R	0x03	InOutWRte_SF	sunssf			Scale factor for percent charge/discharge rate	-2

# Multiple MPPT Inverter Extension Model (I160)

## General

The Multiple MPPT Inverter Extension Model contains the values of up to two DC inverter inputs.

If the inverter has two DC inputs, then this is where the current, voltage, power, energy, and status codes for the individual inputs are listed. In the inverter model (101–103 or 111–113), only the full DC power of both inputs is output in this case. DC current and DC voltage are displayed as “not implemented”.

If the inverter only has one DC input, all values for the second string are set to “not implemented” (from register 2\_DCA). The description of the second input (register 2\_IDStr) appears as “not supported” in this case. The values for the first (and only) input are displayed normally.

## Multiple MPPT Inverter Extension Register

Start address:

- for “float” setting: **40263**
- for “int+SF” setting: **40253**

Start offset	End offset	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
1	1	1	R	0x03	ID	unit16			A well-known value 160. Uniquely identifies this as a SunSpec Multiple MPPT Inverter Extension Model Mode	160
2	2	1	R	0x03	L	uint16			Length of Multiple MPPT Inverter Extension Model	48
3	3	1	R	0x03	DCA_SF	sunssf			Current scale factor	
4	4	1	R	0x03	DCV_SF	sunssf			Voltage scale factor	
5	5	1	R	0x03	DCW_SF	sunssf			Power scale factor	
6	6	1	R	0x03	DCWH_SF	sunssf			Energy scale factor	
7	8	2	R	0x03	Evt	bitfield32			Global events	
9	9	1	R	0x03	N	uint16			Number of modules	2
10	10	1	R	0x03	TmsPer	uint16			Timestamp period	Not supported
11	11	1	R	0x03	1_ID	uint16			Input ID	1
12	19	8	R	0x03	1_IDStr	string8			Input ID string	“String 1” <sup>1)</sup>
20	20	1	R	0x03	1_DCA	uint16	A	DCA_SF	DC current	
21	21	1	R	0x03	1_DCV	uint16	V	DCV_SF	DC voltage	
22	22	1	R	0x03	1_DCW	uint16	W	DCW_SF	DC power	

Start offset	End offset	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
23	24	2	R	0x03	1_DCWH	acc32	Wh	DCWH_SF	Lifetime energy	
25	26	2	R	0x03	1_Tms	uint32	Secs		Timestamp	
27	27	1	R	0x03	1_Tmp	int16	C		Temperature	
28	28	1	R	0x03	1_DCSt	enum16			Operating state	
				Name		Value	Description			
				I_STATUS_OFF		1	Off			
				I_STATUS_SLEEPING		2	In operation (no feed-in)			
				I_STATUS_STARTING		3	Run-up phase			
				I_STATUS_MPPT		4	Normal operation			
				I_STATUS_THROTTLED		5	Power reduction			
				I_STATUS_SHUTTING_DOWN		6	Switch-off phase			
				I_STATUS_FAULT		7	Error exists			
				I_STATUS_STANDBY		8	Standby			
29	30	2	R	0x03	1_DCEvt	bitfield32			Module events	Not supported
31	31	1	R	0x03	2_ID	uint16			Input ID	2
32	39	8	R	0x03	2_IDStr	string8			Input ID string	"String 2" 2) or "Not supported"
40	40	1	R	0x03	2_DCA	uint16	A	DCA_SF	DC current	3)
41	41	1	R	0x03	2_DCV	uint16	V	DCV_SF	DC voltage	3)
42	42	1	R	0x03	2_DCW	uint16	W	DCW_SF	DC power	3)
43	44	2	R	0x03	2_DCWH	acc32	Wh	DCWH_SF	Lifetime energy	
45	46	2	R	0x03	2_Tms	uint32	Secs		Timestamp	
47	47	1	R	0x03	2_Tmp	int16	C		Temperature	
48	48	1	R	0x03	2_DCSt	enum16			Operating state	
				Name		Value	Description			
				I_STATUS_OFF		1	Off			
				I_STATUS_SLEEPING		2	In operation (no feed-in)			
				I_STATUS_STARTING		3	Run-up phase			
				I_STATUS_MPPT		4	Normal operation			
				I_STATUS_THROTTLED		5	Power reduction			
				I_STATUS_SHUTTING_DOWN		6	Switch-off phase			
				I_STATUS_FAULT		7	Error exists			
				I_STATUS_STANDBY		8	Standby			
49	50	2	R	0x03	2_DCEvt	bitfield32			Module events	Not supported

Start offset	End offset	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
--------------	------------	------	-----	----------------	------	------	-------	--------------	-------------	-----------------

- 1) String 1 (MPPT1) = Register 1\_DCA, 1\_DCV, and 1\_DCW
- 2) String 2 (MPPT2) = Register 2\_DCA, 2\_DCV, and 2\_DCW
- 3) Total values

DCW = Total DC power

In hybrid systems:

String 1 = PV input

String 2 = Storage

When discharging the storage: DCW = 1\_DCW + 2\_DCW

When charging the storage: DCW = 1\_DCW - 2\_DCW

### Examples

- a) PV input: 2000 W production ==> 1\_DCW = 2000 W  
 Storage: 1000 W discharge ==> 2\_DCW = 1000 W

$$\text{DCW} = 1_{\text{DCW}} + 2_{\text{DCW}} = 1000 \text{ W} + 2000 \text{ W} = 3000 \text{ W}$$

- b) PV input: 2000 W production ==> 1\_DCW = 2000 W  
 Storage: - 1000 W charge ==> 2\_DCW = 1000 W  
 (only the absolute value can be shown in this register)

$$\text{DCW} = 1_{\text{DCW}} + 2_{\text{DCW}} = 2000 \text{ W} + (-1000 \text{ W}) = 1000 \text{ W}$$

# String Combiner Model (SC403)

## String Combiner Register

Start	End	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
40070	40070	1	R	0x03	ID	uint16			A well-known value 403. Uniquely identifies this as a SunSpec String Combiner (Current) Model	403
40071	40071	1	R	0x03	L	uint16	Registers		Length of string combiner model block	56
40072	40072	1	R	0x03	DCA_SF	sunssf			Current scale factor	
40073	40073	1	R	0x03	DCAhr_SF	sunssf			Amp-hour scale factor	
40074	40074	1	R	0x03	DCV_SF	sunssf			Voltage scale factor	Not supported
40075	40075	1	R	0x03	DCAMax	uint16	A		Maximum DC current rating	
40076	40076	1	R	0x03	N	uint16			Number of inputs	Max. 5
40077	40078	2	R	0x03	Evt	bitfield32			Bitmask value. Events	
40079	40080	2	R	0x03	EvtVnd	bitfield32			Bitmask value. Vendor defined events	
40081	40081	1	R	0x03	DCA	int16	A		Total measured current	
40082	40083	2	R	0x03	DCAhr	acc32	Ah		Total metered amp-hours	
40084	40084	1	R	0x03	DCV	int16	V		Output voltage	Not supported
40085	40085	1	R	0x03	Tmp	int16	C		Internal operating temperature	Not supported
40086	40086	1	R	0x03	InDCA_SF	sunssf			Current scale factor for inputs	
40087	40087	1	R	0x03	InDCAhr_SF	sunssf			Amp-hour scale factor for inputs	
40088	40088	1	R	0x03	InID	uint16			Uniquely identifies this input set	
40089	40090	2	R	0x03	InEvt	bitfield32			String input event flags	
40091	40092	2	R	0x03	InEvtVnd	bitfield32			String input vendor event flags	
40093	40093	1	R	0x03	InDCA	int16	A		String input current	
40094	40095	2	R	0x03	InDCAhr	acc32	Ah		String input amp-hours	
40096	40096	1	R	0x03	InID	uint16			Uniquely identifies this input set	
40097	40098	2	R	0x03	InEvt	bitfield32			String input event flags	
40099	40100	2	R	0x03	InEvtVnd	bitfield32			String input vendor event flags	

<b>Start</b>	<b>End</b>	<b>Size</b>	<b>R/W</b>	<b>Function codes</b>	<b>Name</b>	<b>Type</b>	<b>Units</b>	<b>Scale factor</b>	<b>Description</b>	<b>Range of values</b>
40101	40101	1	R	0x03	InDCA	int16	A		String input current	
40102	40103	2	R	0x03	InDCAhr	acc32	Ah		String input amp-hours	
40104	40104	1	R	0x03	InID	uint16			Uniquely identifies this input set	
40105	40106	2	R	0x03	InEvt	bitfield32			String input event flags	
40107	40108	2	R	0x03	InEvtVnd	bitfield32			String input vendor event flags	
40109	40109	1	R	0x03	InDCA	int16	A		String input current	
40110	40111	2	R	0x03	InDCAhr	acc32	Ah		String input amp-hours	
40112	40112	1	R	0x03	InID	uint16			Uniquely identifies this input set	
40113	40114	2	R	0x03	InEvt	bitfield32			String input event flags	
40115	40116	2	R	0x03	InEvtVnd	bitfield32			String input vendor event flags	
40117	40117	1	R	0x03	InDCA	int16	A		String input current	
40118	40119	2	R	0x03	InDCAhr	acc32	Ah		String input amp-hours	
40120	40120	1	R	0x03	InID	uint16			Uniquely identifies this input set	
40121	40122	2	R	0x03	InEvt	bitfield32			String input event flags	
40123	40124	2	R	0x03	InEvtVnd	bitfield32			String input vendor event flags	
40125	40125	1	R	0x03	InDCA	int16	A		String input current	
40126	40127	2	R	0x03	InDCAhr	acc32	Ah		String input amp-hours	

# Meter Model

EN-US

## Meter Model Register

The data of an energy meter connected with the Fronius Datamanager via Modbus RTU can be read by the relevant SunSpec models via Modbus TCP.

In a similar way to the inverter models, there are also two different SunSpec models in this case:

- the meter model with floating point display (setting “float”; M211, M212, or M213)
- the meter model with integers and scale factors (setting “int+SF”; M201, M202, or M203)



**NOTE!** The register number of the two model types is different!

The Modbus device ID of the energy meter is 240.

### Floating point display – “float”

Start	End	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
40070	40070	1	R	0x03	ID	uint16			Uniquely identifies this as a SunSpec Meter Modbus Map 211: single phase 212: split phase 213: three phase	211, 212, 213
40071	40071	1	R	0x03	L	uint16	Registers		Length of meter model block	124
40072	40073	2	R	0x03	A	float32	A		AC total current value	
40074	40075	2	R	0x03	AphA	float32	A		AC phase A current value	
40076	40077	2	R	0x03	AphB	float32	A		AC phase B current value	
40078	40079	2	R	0x03	AphC	float32	A		AC phase C current value	
40080	40081	2	R	0x03	PhV	float32	V		AC voltage average phase-to-neutral value	
40082	40083	2	R	0x03	PhVphA	float32	V		AC voltage phase-A-to-neutral value	
40084	40085	2	R	0x03	PhVphB	float32	V		AC voltage phase-B-to-neutral value	
40086	40087	2	R	0x03	PhVphC	float32	V		AC voltage phase-C-to-neutral value	
40088	40089	2	R	0x03	PPV	float32	V		AC voltage average phase-to-phase value	
40090	40091	2	R	0x03	PPVphAB	float32	V		AC voltage phase AB value	
40092	40093	2	R	0x03	PPVphBC	float32	V		AC voltage phase BC value	

### Floating point display – “float”

Start	End	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
40094	40095	2	R	0x03	PPVphCA	float32	V		AC voltage phase CA value	
40096	40097	2	R	0x03	Hz	float32	Hz		AC frequency value	
40098	40099	2	R	0x03	W	float32	W		AC power value	
40100	40101	2	R	0x03	WphA	float32	W		AC power phase A value	
40102	40103	2	R	0x03	WphB	float32	W		AC power phase B value	
40104	40105	2	R	0x03	WphC	float32	W		AC power phase C value	
40106	40107	2	R	0x03	VA	float32	VA		AC apparent power value	
40108	40109	2	R	0x03	V AphA	float32	VA		AC apparent power phase A value	
40110	40111	2	R	0x03	V AphB	float32	VA		AC apparent power phase B value	
40112	40113	2	R	0x03	V AphC	float32	VA		AC apparent power phase C value	
40114	40115	2	R	0x03	VAR	float32	VAr		AC reactive power value	
40116	40117	2	R	0x03	VARphA	float32	VAr		AC reactive power phase A value	
40118	40119	2	R	0x03	VARphB	float32	VAr		AC reactive power phase B value	
40120	40121	2	R	0x03	VARphC	float32	VAr		AC reactive power phase C value	
40122	40123	2	R	0x03	PF	float32	PF		Power factor value	
40124	40125	2	R	0x03	PFphA	float32	PF		Power factor phase A value	
40126	40127	2	R	0x03	PFphB	float32	PF		Power factor phase B value	
40128	40129	2	R	0x03	PFphC	float32	PF		Power factor phase C value	
40130	40131	2	R	0x03	TotWhExp	float32	Wh		Total watt-hours exported	
40132	40133	2	R	0x03	TotWhExpPhA	float32	Wh		Total watt-hours exported phase A	
40134	40135	2	R	0x03	TotWhExpPhB	float32	Wh		Total watt-hours exported phase B	
40136	40137	2	R	0x03	TotWhExpPhC	float32	Wh		Total watt-hours exported phase C	
40138	40139	2	R	0x03	TotWhImp	float32	Wh		Total watt-hours imported	
40140	40141	2	R	0x03	TotWhImpPhA	float32	Wh		Total watt-hours imported phase A	
40142	40143	2	R	0x03	TotWhImpPhB	float32	Wh		Total watt-hours imported phase B	

**Floating point display – “float”**

<b>Start</b>	<b>End</b>	<b>Size</b>	<b>R/W</b>	<b>Function codes</b>	<b>Name</b>	<b>Type</b>	<b>Units</b>	<b>Scale factor</b>	<b>Description</b>	<b>Range of values</b>
40144	40145	2	R	0x03	TotWhImpPhC	float32	Wh		Total watt-hours imported phase C	
40146	40147	2	R	0x03	TotVAhExp	float32	VAh		Total VA-hours exported	
40148	40149	2	R	0x03	TotVAhExpPhA	float32	VAh		Total VA-hours exported phase A	
40150	40151	2	R	0x03	TotVAhExpPhB	float32	VAh		Total VA-hours exported phase B	
40152	40153	2	R	0x03	TotVAhExpPhC	float32	VAh		Total VA-hours exported phase C	
40154	40155	2	R	0x03	TotVAhImp	float32	VAh		Total VA-hours imported	
40156	40157	2	R	0x03	TotVAhImpPhA	float32	VAh		Total VA-hours imported phase A	
40158	40159	2	R	0x03	TotVAhImpPhB	float32	VAh		Total VA-hours imported phase B	
40160	40161	2	R	0x03	TotVAhImpPhC	float32	VAh		Total VA-hours imported phase C	
40162	40163	2	R	0x03	TotVArhImpQ1	float32	VArh		Total VAR-hours imported Q1	
40164	40165	2	R	0x03	TotVArhImpQ1phA	float32	VArh		Total VAR-hours imported Q1 phase A	
40166	40167	2	R	0x03	TotVArhImpQ1phB	float32	VArh		Total VAR-hours imported Q1 phase B	
40168	40169	2	R	0x03	TotVArhImpQ1phC	float32	VArh		Total VAR-hours imported Q1 phase C	
40170	40171	2	R	0x03	TotVArhImpQ2	float32	VArh		Total VAR-hours imported Q2	
40172	40173	2	R	0x03	TotVArhImpQ2phA	float32	VArh		Total VAR-hours imported Q2 phase A	
40174	40175	2	R	0x03	TotVArhImpQ2phB	float32	VArh		Total VAR-hours imported Q2 phase B	
40176	40177	2	R	0x03	TotVArhImpQ2phC	float32	VArh		Total VAR-hours imported Q2 phase C	
40178	40179	2	R	0x03	TotVArhExpQ3	float32	VArh		Total VAR-hours exported Q3	
40180	40181	2	R	0x03	TotVArhExpQ3phA	float32	VArh		Total VAR-hours exported Q3 phase A	
40182	40183	2	R	0x03	TotVArhExpQ3phB	float32	VArh		Total VAR-hours exported Q3 phase B	
40184	40185	2	R	0x03	TotVArhExpQ3phC	float32	VArh		Total VAR-hours exported Q3 phase C	
40186	40187	2	R	0x03	TotVArhExpQ4	float32	VArh		Total VAR-hours exported Q4	
40188	40189	2	R	0x03	TotVArhExpQ4phA	float32	VArh		Total VAR-hours exported Q4 phase A	

### Floating point display – “float”

Start	End	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
40190	40191	2	R	0x03	TotVArhExpQ4phB	float32	VArh		Total VAR-hours exported Q4 phase B	
40192	40193	2	R	0x03	TotVArhExpQ4phC	float32	VArh		Total VAR-hours exported Q4 phase C	
40194	40195	2	R	0x03	Evt	uint32	Bit field		Events	

### Integer Display – “int+SF”

Start	End	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
40070	40070	1	R	0x03	ID	uint16			Uniquely identifies this as a SunSpec Meter Modbus Map 201: single phase 202: split phase 203: three phase	201, 202, 203
40071	40071	1	R	0x03	L	uint16	Registers		Length of meter model block	105
40072	40072	1	R	0x03	A	int16	A		AC total current value	
40073	40073	1	R	0x03	AphA	int16	A		AC phase A current value	
40074	40074	1	R	0x03	AphB	int16	A		AC phase B current value	
40075	40075	1	R	0x03	AphC	int16	A		AC phase C current value	
40076	40076	1	R	0x03	A_SF	int16	sunssf		AC current scale factor	
40077	40077	1	R	0x03	PhV	int16	V		AC voltage average phase-to-neutral value	
40078	40078	1	R	0x03	PhVphA	int16	V		AC voltage phase-A-to-neutral value	
40079	40079	1	R	0x03	PhVphB	int16	V		AC voltage phase-B-to-neutral value	
40080	40080	1	R	0x03	PhVphC	int16	V		AC voltage phase-C-to-neutral value	
40081	40081	1	R	0x03	PPV	int16	V		AC voltage average phase-to-phase value	
40082	40082	1	R	0x03	PPVphAB	int16	V		AC voltage phase AB value	
40083	40083	1	R	0x03	PPVphBC	int16	V		AC voltage phase BC value	
40084	40084	1	R	0x03	PPVphCA	int16	V		AC voltage phase CA value	

**Integer Display – “int+SF”**

<b>Start</b>	<b>End</b>	<b>Size</b>	<b>R/W</b>	<b>Function codes</b>	<b>Name</b>	<b>Type</b>	<b>Units</b>	<b>Scale factor</b>	<b>Description</b>	<b>Range of values</b>
40085	40085	1	R	0x03	V_SF	int16	sunssf		AC phase voltage scale factor	
40086	40086	1	R	0x03	Hz	int16	Hz		AC frequency value	
40087	40087	1	R	0x03	Hz_SF	int16	sunssf		AV frequency scale factor	
40088	40088	1	R	0x03	W	int16	W		AC power value	
40089	40089	1	R	0x03	WphA	int16	W		AC power phase A value	
40090	40090	1	R	0x03	WphB	int16	W		AC power phase B value	
40091	40091	1	R	0x03	WphC	int16	W		AC power phase C value	
40092	40092	1	R	0x03	W_SF	int16	sunssf		AC phase power scale factor	
40093	40093	1	R	0x03	VA	int16	VA		AC apparent power value	
40094	40094	1	R	0x03	VApH_A	int16	VA		AC apparent power phase A value	
40095	40095	1	R	0x03	VApH_B	int16	VA		AC apparent power phase B value	
40096	40096	1	R	0x03	VApH_C	int16	VA		AC apparent power phase C value	
40097	40097	1	R	0x03	VA_SF	int16	sunssf		AC phase apparent power scale factor	
40098	40098	1	R	0x03	VAR	int16	VAr		AC reactive power value	
40099	40099	1	R	0x03	VARphA	int16	VAr		AC reactive power phase A value	
40100	40100	1	R	0x03	VARphB	int16	VAr		AC reactive power phase B value	
40101	40101	1	R	0x03	VARphC	int16	VAr		AC reactive power phase C value	
40102	40102	1	R	0x03	VAR_SF	int16	sunssf		AC phase reactive power scale factor	
40103	40103	1	R	0x03	PF	int16	PF		Power factor value	
40104	40104	1	R	0x03	PFphA	int16	PF		Power factor phase A value	
40105	40105	1	R	0x03	PFphB	int16	PF		Power factor phase B value	
40106	40106	1	R	0x03	PFphC	int16	PF		Power factor phase C value	
40107	40107	1	R	0x03	PF_SF	int16	sunssf		Phase power factor scale factor	
40108	40109	2	R	0x03	TotWhExp	uint32	Wh		Total watt-hours exported	
40110	40111	2	R	0x03	TotWhExpPhA	uint32	Wh		Total watt-hours exported phase A	

**Integer Display – “int+SF”**

<b>Start</b>	<b>End</b>	<b>Size</b>	<b>R/W</b>	<b>Function codes</b>	<b>Name</b>	<b>Type</b>	<b>Units</b>	<b>Scale factor</b>	<b>Description</b>	<b>Range of values</b>
40112	40113	2	R	0x03	TotWhExpPhB	uint32	Wh		Total watt-hours exported phase B	
40114	40115	2	R	0x03	TotWhExpPhC	uint32	Wh		Total watt-hours exported phase C	
40116	40117	2	R	0x03	TotWhImp	uint32	Wh		Total watt-hours imported	
40118	40119	2	R	0x03	TotWhImpPhA	uint32	Wh		Total watt-hours imported phase A	
40120	40121	2	R	0x03	TotWhImpPhB	uint32	Wh		Total watt-hours imported phase B	
40122	40123	2	R	0x03	TotWhImpPhC	uint32	Wh		Total watt-hours imported phase C	
40124	40124	1	R	0x03	TotWh_SF	int16	sunssf		Total watt scale factor	
40125	40126	2	R	0x03	TotVAhExp	uint32	VAh		Total VA-hours exported	
40127	40128	2	R	0x03	TotVAhExpPhA	uint32	VAh		Total VA-hours exported phase A	
40139	40130	2	R	0x03	TotVAhExpPhB	uint32	VAh		Total VA-hours exported phase B	
40131	40132	2	R	0x03	TotVAhExpPhC	uint32	VAh		Total VA-hours exported phase C	
40133	40134	2	R	0x03	TotVAhImp	uint32	VAh		Total VA-hours imported	
40135	40136	2	R	0x03	TotVAhImpPhA	uint32	VAh		Total VA-hours imported phase A	
40137	40138	2	R	0x03	TotVAhImpPhB	uint32	VAh		Total VA-hours imported phase B	
40139	40140	2	R	0x03	TotVAhImpPhC	uint32	VAh		Total VA-hours imported phase C	
40141	40141	1	R	0x03	TotVAh_SF	int16	sunssf		Total VA-hours scale factor	
40142	40143	2	R	0x03	TotVArhImpQ1	uint32	VArh		Total VAR-hours imported Q1	
40144	40145	2	R	0x03	TotVArhImpQ1phA	uint32	VArh		Total VAR-hours imported Q1 phase A	
40146	40147	2	R	0x03	TotVArhImpQ1phB	uint32	VArh		Total VAR-hours imported Q1 phase B	
40148	40149	2	R	0x03	TotVArhImpQ1phC	uint32	VArh		Total VAR-hours imported Q1 phase C	
40150	40151	2	R	0x03	TotVArhImpQ2	uint32	VArh		Total VAR-hours imported Q2	
40152	40153	2	R	0x03	TotVArhImpQ2phA	uint32	VArh		Total VAR-hours imported Q2 phase A	
40154	40155	2	R	0x03	TotVArhImpQ2phB	uint32	VArh		Total VAR-hours imported Q2 phase B	
40156	40157	2	R	0x03	TotVArhImpQ2phC	uint32	VArh		Total VAR-hours imported Q2 phase C	

**Integer Display – “int+SF”**

<b>Start</b>	<b>End</b>	<b>Size</b>	<b>R/W</b>	<b>Function codes</b>	<b>Name</b>	<b>Type</b>	<b>Units</b>	<b>Scale factor</b>	<b>Description</b>	<b>Range of values</b>
40158	40159	2	R	0x03	TotVArhExpQ3	uint32	VArh		Total VAR-hours exported Q3	
40160	40161	2	R	0x03	TotVArhExpQ3phA	uint32	VArh		Total VAR-hours exported Q3 phase A	
40162	40163	2	R	0x03	TotVArhExpQ3phB	uint32	VArh		Total VAR-hours exported Q3 phase B	
40164	40165	2	R	0x03	TotVArhExpQ3phC	uint32	VArh		Total VAR-hours exported Q3 phase C	
40166	40167	2	R	0x03	TotVArhExpQ4	uint32	VArh		Total VAR-hours exported Q4	
40168	40169	2	R	0x03	TotVArhExpQ4phA	uint32	VArh		Total VAR-hours exported Q4 phase A	
40170	40171	2	R	0x03	TotVArhExpQ4phB	uint32	VArh		Total VAR-hours exported Q4 phase B	
40172	40173	2	R	0x03	TotVArhExpQ4phC	uint32	VArh		Total VAR-hours exported Q4 phase C	
40174	40174	1	R	0x03	TotVArh_SF	int16	sunssf		Total VAR-hours scale factor	
40175	40176	2	R	0x03	Evt	uint32	Bit field		Events	

# End Block

---

<b>General</b>	Two registers according to the last data model indicate that no further SunSpec models will follow. The addresses of these two registers are different depending on the device type (inverter, String Control, energy meter) and selected data type ("float" or "int+SF"). <ul style="list-style-type: none"><li>- Inverter:<ul style="list-style-type: none"><li>- Start address for setting "float": 40313</li><li>- Start address for setting "int+SF": 40303</li></ul></li><li>- Fronius String Control:<ul style="list-style-type: none"><li>- Start address: 40127</li></ul></li><li>- Energy meter:<ul style="list-style-type: none"><li>- Start address for setting "float": 40195</li><li>- Start address for setting "int+SF": 40176</li></ul></li></ul>
----------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## End Block

Start offset	End offset	Size	R/W	Function codes	Name	Type	Units	Scale factor	Description	Range of values
1	1	1	R	0x03	ID	uint16			Identifies this as end block	0xFFFF
2	2	1	R	0x03	L	uint16	Registers		Length of model block	0

# String Combiner Event Flags

String Combiner  
Event Flags

Name	Event Flags
LOW_VOLTAGE	0x00000001
LOW_POWER	0x00000002
LOW EFFICIENCY	0x00000004
CURRENT	0x00000008
VOLTAGE	0x00000010
POWER	0x00000020
PR	0x00000040
DISCONNECTED	0x00000080
FUSE_FAULT	0x00000100
COMBINER_FUSE_FAULT	0x00000200
COMBINER_CABINET_OPEN	0x00000400
TEMP	0x00000800
GROUNDFAULT	0x00001000
REVERSED_POLARITY	0x00002000
INCOMPATIBLE	0x00004000
COMM_ERROR	0x00008000
INTERNAL_ERROR	0x00010000
THEFT	0x00020000
ARC_DETECTED	0x00040000

# **Fronius Worldwide - [www.fronius.com/addresses](http://www.fronius.com/addresses)**

**Fronius International GmbH**  
4600 Wels, Froniusplatz 1, Austria  
E-Mail: [pv-sales@fronius.com](mailto:pv-sales@fronius.com)  
<http://www.fronius.com>

**Fronius USA LLC** Solar Electronics Division  
6797 Fronius Drive, Portage, IN 46368  
E-Mail: [pv-us@fronius.com](mailto:pv-us@fronius.com)  
<http://www.fronius-usa.com>

Under <http://www.fronius.com/addresses> you will find all addresses of our sales branches and partner firms!