

FAST EnergyCam Modbus Slave Protokoll

Inhaltsverzeichnis

Einführung.....	1
Punkt zu Punkt Kommunikation.....	2
Lese M-Bus secondary address.....	2
Wake up.....	2
Generelles Modbus Timing.....	3
Unterstützte Aktionen.....	3
Modbus Register Adressen.....	4
Unterstützte Input-Register.....	4
Beispiele.....	6
Lese AppRevision.....	6
Lese ResultOCR.....	7
Unterstützte Holding-Register.....	8
Beispiele.....	11
Get OCRConfig.....	11
Set ActionPowerDown.....	12
Set ActionOCRInstallation.....	13
Fehlerfälle.....	14
Automatisches Verhalten.....	14
Historie.....	15

Tabellen

Tabelle 1: M-Bus secondary address.....	2
Tabelle 2: Unterstützte Input-Register.....	6
Tabelle 3: Beispiel für das Lesen von mehreren Input-Registern "AppRevision".....	7
Tabelle 4: Beispiel Antwort für erfolgreiches Lesen von mehreren Input-Registern "AppRevision".....	7
Tabelle 5: Beispiel für das Lesen von mehreren Input-Registern ResultOCR.....	8
Tabelle 6: Beispiel Antwort für erfolgreiches Lesen von mehreren Input-Registern ResultOCR.....	8
Tabelle 7: Unterstützte Holding-Register.....	11
Tabelle 8: Beispiel zum Lesen von mehreren Holding-Registern.....	12
Tabelle 9: Beispielantwort für erfolgreiches Lesen von mehreren Holding-Registern.....	12
Tabelle 10: Beispiel für das Schreiben auf ein Holding-Register.....	12
Tabelle 11: Beispielantwort für erfolgreiches Schreiben eines Holding-Registers.....	13
Tabelle 12: Beispiel für das Schreiben mehrerer Holding-Register.....	13
Tabelle 13: Beispielantwort für erfolgreiches Schreiben mehrerer Holding-Register.....	14
Tabelle 14: Beispiel Antwort für Schreiben auf eine unbekannte Holding-Register Adresse.....	14
Tabelle 15: Timings für automatisches Verhalten.....	15
Tabelle 16: Historie.....	15

Bilder

Bild 1: Blockschaltbild Punkt zu Punkt Kommunikation.....	2
---	---

Einführung

FAST EnergyCam kann über eine Zweidrahtleitung mittels Modbus-Protokoll kommunizieren. Eine generelle Beschreibung des Modbus-Protokolls ist in diesem Dokument zu finden: "EnergyCam Modbus General" (FAST_EnergyCam-Protokoll-MODBUS-General.pdf)

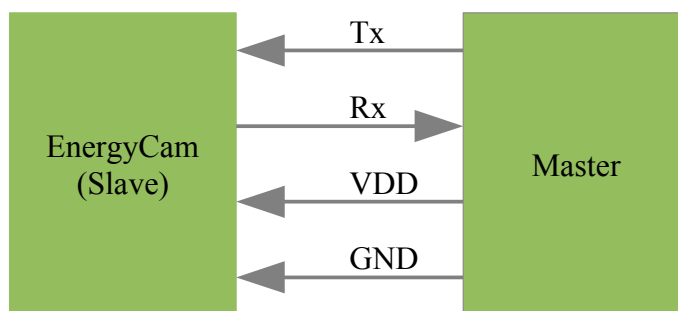
Die Grundeinstellung benutzt 115200 Baud, gerade Parität und 1 Stop-Bit, Slave-Adresse 1.

Punkt zu Punkt Kommunikation

Im Falle einer Punkt zu Punkt Verbindung sind zwei Leitungen ausreichend (Tx, Rx). EnergyCam fungiert als Slave und wartet auf Kommandos, die vom Master gesendet werden. Neben den beiden Datenleitungen ist noch eine Masseleitung und eine Spannungsversorgung nötig.

EnergyCam arbeitet intern mit 2.1V, so dass auch die Master-Rx Leitung diesen gegenüber der Spannungsversorgung niedrigeren Spannungswert aufweist.

Bild 1: Blockschaltbild Punkt zu Punkt Kommunikation



Lese M-Bus secondary address

Die standard M-Bus secondary address, wie sie auch bei wired M-Bus und wireless M-Bus (wM-Bus) benutzt wird, kann mit den folgenden Modbus-Registern gelesen werden.

Name	Zugriff	Modbus-Register	Länge [Byte]	Beschreibung
Manufacturer ID	read-only	ManufacturerIdentification	2	Drei-Buchstaben-Code "FFD", kodiert als 0x18C4.
Identnumber	read-only	MBusIdentNumber	4	Identnumber kodiert mit 8 BCD Zahlen, intern abgeleitet aus einer eindeutigen 64 Bit Gerätenummer
Device type	read-write	OMSCfg[7:0] SetOMSCfg[7:0]	1	Nur diese Werte werden unterstützt: 2: Strom (kWh) 3: Gas (m³) 7: Wasser (m³)
Version	N/A	N/A	1	Immer 0x01

Tabelle 1: M-Bus secondary address

Nur der Device type ist durch Benutzer änderbar.

Wake up

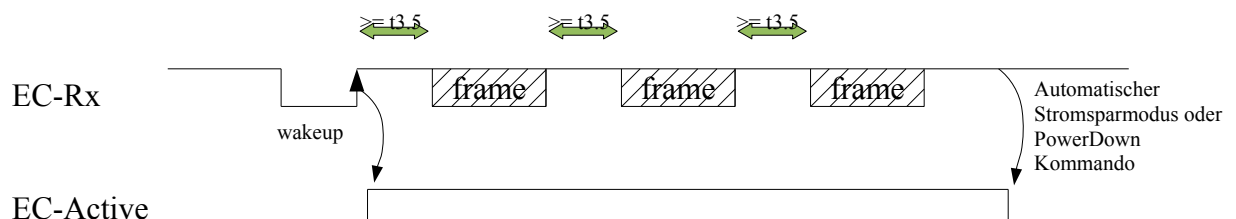
Als Erweiterung zum Modbus hat EnergyCam ein spezielles „Wake up“-Verfahren implementiert, falls der Stromsparmmodus verwendet wird. In diesem Fall muss der Modbus Master EnergyCam vor dem Zugriff ein „Wake up“-Kommando schicken.

Der folgende Abschnitt illustriert die Spannungspegel, wie sie der interne Prozessor empfängt und berücksichtigt keine eventuell benutzten Pegelwandler (wie RS232).

Die Master-Tx Leitung ist mit EnergyCams Rx Leitung verbunden. Diese Leitung ist, wenn keine Daten übertragen werden High. Im Stromsparmmodus wird auf eine steigende Flanke an dieser Leitung gewartet.

Jede steigende Flanke beendet den Stromsparmodes und nach einer gewissen Startzeit (t.b.d) können Modbus Frames empfangen werden. Der Master kann jede Methode verwenden, um eine steigende Flanke zu erzeugen. Zum Beispiel kann ein einzelnes Byte mit dem Wert 0x00 gesendet werden, was genau eine steigende Flanke zur Folge hat. Wenn die Startzeit und danach ein korrektes „silent interval“ (t3.5) eingehalten wird, werden nachfolgende Modbus Frames ohne CRC-Fehler erkannt und ein entsprechendes Antwort-Frame kann erwartet werden. Wird ein Wake up Byte gesendet während sich EnergyCam nicht mehr im Stromsparmodes befindet, wird dieses ignoriert, solange danach ein „silent interval“ (t3.5) eingehalten wird. Dadurch wird das Wake up Byte als fehlerhaftes Frame (CRC-Fehler) erkannt und ignoriert.

Auf den Wert „silent interval“ (t3.5) wird im Dokument *FAST_EnergyCam-Protokoll-MODBUS-General.pdf* eingegangen.



Generelles Modbus Timing

Durch EnergyCam interne Echtzeitanforderungen kann nicht immer garantiert werden, dass auf jede Modbus-Anfrage reagiert wird. Wenn also eine Modbus-Antwort ausbleibt ist die Anfrage zu wiederholen.

Unterstützte Aktionen

EnergyCam unterstützt verschiedene Aktionen, welche mit Modbus-Kommandos ausgelöst werden können:

- **ActionOCRInstallation:** Startet EnergyCam OCR Installation. Diese muss durchgeführt werden, wenn EnergyCam auf einem Zähler montiert wird oder EnergyCam neu gestartet wurde.
- **ActionOCR:** Standardmäßig liest EnergyCam automatisch alle 15 Minuten den Zähler ab. Durch dieses Kommando kann eine zusätzliche Ablesung durchgeführt werden. Diese Aktion wird aber nur durchgeführt, falls eine Installation erfolgreich war. Diese kann durch Lesen des *InputReg ResultInstallation* überprüft werden.
- **ActionPowerDown:** Normalerweise aktiviert EnergyCam nach einer gewissen Zeit (siehe Kapitel Automatisches Verhalten) automatisch den Stromsparmodes. Um Strom zu sparen kann der Stromsparmodes durch dieses Kommando sofort ausgelöst werden.

Um Aktionen auszulösen ist folgende Abfolge einzuhalten:

- Sicherstellen, dass EnergyCam gerade keine Aktion ausführt:
Wiederholtes Lesen des InputReg StatusEnergyCam bis der Wert ungleich ActionOngoing ist.
- Starten der Aktion:
Schreibe 0x0001 in HoldingReg Action...
- Prüfen auf Ende der Aktion:
Wiederholtes Lesen des InputReg StatusEnergyCam bis der Wert ungleich ActionOngoing ist.
- Lesen des Ergebnis der Aktion:
ActionInstallation: Lese InputReg ResultInstallation.
ActionOCR: Lese InputReg ResultOCR und folgende Register auf einmal.

Modbus Register Adressen

Modbus definiert Register Adressen, die ab 1 gezählt werden (1-basiert). Manche Modbus Master Implementierungen zählen jedoch ab 0 (0-basiert). Bitte prüfen Sie, welche Implementierung Sie verwenden und benutzen Sie die entsprechende Spalte in den folgenden Tabellen. Unabhängig von der Adressierung ist die physikalische Übertragung immer 0-basiert.

Um einen Modbus Master zu testen, können die folgenden Register verwendet werden:

- Input-Register: Test1¹, Test0
- Holding Register: TestReadOnly1, TestReadOnly0, TestReadWrite1 und TestReadWrite0.

Das Lesen und Schreiben dieser Register zu Testzwecken, hat keine Auswirkung auf EnergyCam.

Unterstützte Input-Register

Da Modbus nur mit 16 Bit breiten Einzelzugriffen arbeitet, definiert EnergyCam einige aufeinanderfolgende Register als Block, die auf einmal zu lesen sind.

Modbus-Adresse 0-basiert	Modbus-Adresse 1-basiert	Block Name	Register Name [16 Bits]	Beschreibung
0x0000	0x0001	ProtokollVersion	ProtokollVersion	EnergyCam Modbus-Protokoll Version: z.B: 5
0x0001	0x0002	Manufacturer-Identification	Manufacturer-Identification	Drei-Buchstaben-Code ² für "FAST FORWARD AG" in M-Bus Format: "FFD" = 0x18C4
0x0002	0x0003	DeviceID	DeviceID3	DeviceID [63..48]
0x0003	0x0004		DeviceID2	DeviceID [47..32]
0x0004	0x0005		DeviceID1	DeviceID [31..16]
0x0005	0x0006		DeviceID0	DeviceID [15..0], Beispiel 0x4F92F42C109AB502
0x0006	0x0007	AppRevision	AppRevisionMajor	z.B. 2, Revision der Applikation
0x0007	0x0008		AppRevisionMinor	z.B. 0, die gesamte Revision ist damit z.B. "2.0"
0x0008	0x0009	AppType	AppType	Typ der Applikation, 0x0100 für Sensor
0x0009	0x000A	Time	Time1	Zeit [31..16], UNIX Epoche Zeit in Sekunden (UTC), 0 bedeutet 1.1.1970 00:00
0x000A	0x000B		Time0	Zeit [15..0]
0x000B	0x000C	AppBuildnumber	AppBuildnumber1	Build-Nummer der Applikation[31..16]
0x000C	0x000D		AppBuildnumber0	Build-Nummer der Applikation[15..0]
0x000D	0x000E	EPLDRevision	EPLDRevision	EPLDRevision, z.B. 3
0x000E	0x000F	PCBRevision	PCBRevision	PCBRevision
0x000F	0x0010	OMSConfig	OMSConfig	Meter und wireless M-Bus Konfiguration [7:0] Geräte Typ (2: Strom; 3: Gas; 7: Wasser) [8:12] reserviert [13] wM-Bus aktiviert [14] wM-Bus verschlüsselt [15] wM-Bus Installation Modus (0: auto; 1: manuell)
0x0010	0x0011	AppFirmwareType	AppFirmwareType	FirmwareType der Applikation (wie wM-Bus Stack Version T2, S2)
0x0011	0x0012	MBusIdentNumber	MBusIdentNumber1	Identnummer[31:16] in M-Bus Format: 8 BCD Zahlen, abgeleitet

1 Die Testregister sind ab ProtokollVersion 5 verfügbar

2 Siehe <http://www.m-bus.com/files/MBDOC48.PDF>

Modbus-Adresse 0-basiert	Modbus-Adresse 1-basiert	Block Name	Register Name [16 Bits]	Beschreibung
				von DeviceID, Beispiel 0x12345678
0x0012	0x0013		MBusIdentNumber0	Identnummer[15:0]
0x0013	0x0014	BLBuildNumber	BLBuildNumber1	Build-Nummer des Bootloaders [31..16]
0x0014	0x0015		BLBuildNumber0	Build-Nummer des Bootloaders [15..0]
0x0015	0x0016	BLFirmwareType	BLFirmwareType	Bootloader FirmwareType
0x0016	0x0017	Test1 ³	Test1	Test Register für Modbus Master Tests: Enthält 0xABCD = 43981
0x0017	0x0018	Test0 ³	Test0	Test Register für Modbus Master Tests: Enthält 0x1234 = 4660 Für 32 Bit Zugriffe (d.h. unsigned DWORD) sollten zwei 16 Bit Register ab Test1 gelesen werden. Das Ergebnis sollte 0xABCD1234 = 2882343476 sein. Im Falle von 0x1234ABCD = 305441741 sollten die beiden 16 Bit Wörter vertauscht werden („word swap“)
Lücke	**Lücke**			
0x0001F	0x0020	StatusEnergyCam	StatusEnergyCam	Aktueller Status von EnergyCam: 0: Ungültig 1: PowerDown (war eigentlich gerade in Power-Down) 2: ActionOngoing 3: ActionCompletedSuccessfully 4: ActionCompletedWithError
0x0020	0x0021	ResultInstallation	ResultInstallation	Installationsergebnis: 0x0000: Ungültig 0xIIFF: OKDIGITSFOUND (I = Integer Teil, F = Nachkommaanteil (Fraction)) 0xFFFD: ONGOING 0xFFFE: NOTDONE 0xFFFF: ERROR
0x0021	0x0022	ResultOCR	ResultOCRValid	Ergebnis der letzten OCR: 0x0000: Ungültig 0x0001: OK (Alle Zahlen konnten interpretiert werden) 0x0003: OK (Wiederholung des letzten Werts durch Fehlerkorrektur) 0xFFFD: ONGOING 0xFFFE: NOTDONE 0xFFFF: ERROR (Nicht alle Zahlen konnten interpretiert werden)
0x0022	0x0023	ResultOCRIntChar	ResultOCRIntChar7	Darf nur gelesen werden wenn ResultOCRValid = OK. ResultOCRIntChar[7..0] beinhaltet OCR Integer Ergebnis als ASCII String, rechtsbündig, links aufgefüllt mit '-', falls weniger als 8 Zeichen auf Zähler gefunden wurden. Nur Integer, i.e. die ganz rechts liegende Zahl ist in in OCRResult0 char
0x0023	0x0024		ResultOCRIntChar6	char
0x0024	0x0025		ResultOCRIntChar5	char
0x0025	0x0026		ResultOCRIntChar4	char

3 Die Testregister sind ab ProtokollVersion 5 verfügbar

Modbus-Adresse 0-basiert	Modbus-Adresse 1-basiert	Block Name	Register Name [16 Bits]	Beschreibung
0x0026	0x0027		ResultOCRIntChar3	char
0x0027	0x0028		ResultOCRIntChar2	char
0x0028	0x0029		ResultOCRIntChar1	char
0x0029	0x002A		ResultOCRIntChar0	char
0x002A	0x002B	ResultOCR-FracChar	ResultOCRFracChar3	ResultOCRFracChar[3..0] beinhaltet OCR Nachkomma Ergebnis als ASCII String, links bündig. Rechts aufgefüllt mit '-', falls weniger als 4 Zeichen auf Zähler gefunden wurden. Beispiel: "12345678.1234" oder "---12345.1---" char
0x002B	0x002C		ResultOCRFracChar2	char
0x002C	0x002D		ResultOCRFracChar1	char
0x002D	0x002E		ResultOCRFracChar0	char
Lücke	**Lücke**			
0x003C	0x003D		OCRInfoRed	Horizontale Position der erkannten roten Markierung [11:0] Horizontale Position [15] Über Kopf installiert
Lücke	**Lücke**			
0x0043	0x0044	ResultOCRInt	ResultOCRInt1	Darf nur gelesen werden wenn ResultOCRValid = OK. Beinhaltet OCR Ergebnis als Integer. ResultOCRInt[31..16]
0x0044	0x0045		ResultOCRInt0	ResultOCRInt[15..0]
0x0045	0x0046	ResultOCRFrac	ResultOCRFrac	Darf nur gelesen werden wenn ResultOCRValid = OK. Beinhaltet ein einstelliges OCR Nachkomma Ergebnis als Integer. Für mehr Nachkommastellen ResultOCR64 verwenden. ResultOCRFrac[15..0]
Lücke	**Lücke**			
0x004E	0x004F	ResultOCR64	ResultOCR64_3	ResultOCR64 beinhaltet das OCR Ergebnis (Vor- und Nachkommastellen) als 64 Bit Integer Wert. Dazu ist der Integer-Wert um Faktor 1000 skaliert, d.h. 12345.1 wird als 12345100 dargestellt.
0x004F	0x0050		ResultOCR64_2	
0x0050	0x0051		ResultOCR64_1	
0x0051	0x0052		ResultOCR64_0	
Lücke	**Lücke**			
0x7FFF	0x8000	UpdateCRCOK		Berechnet CRC des aktuell gespeicherten Update Image und liefert 0 (CRC falsch) oder 1 (CRC OK). Wenn OK wird Firmware-Update Prozess gestartet (führt zum Neustart von EnergyCam).

Tabelle 2: Unterstützte Input-Register

Beispiele

Lese AppRevision

Es folgt ein Beispiel für das Lesen von mehreren Input-Registern ab Adresse 0x0006 (0-basiert, "AppRevision"), welches die Revision zurückgibt, also z.B.: "2.0":

0x0006 → 2
0x0007 → 0

Master-Tx: 01 04 00 06 00 02 91 CA

Wert [hex]	Beschreibung
01	EnergyCams Slave Adresse
04	Code für Lesen von aufeinanderfolgenden Input-Registern
00	Adresse des ersten Input-Registers (höherwertiges Byte)
06	Adresse des ersten Input-Registers (niederwertiges Byte)
00	Anzahl der Register die zu lesen sind (höherwertiges Byte)
02	Anzahl der Register die zu lesen sind (niederwertiges Byte)
91	CRC (höherwertiges Byte)
CA	CRC (niederwertiges Byte)

Tabelle 3: Beispiel für das Lesen von mehreren Input-Registern "AppRevision"

Im Erfolgsfall antwortet EnergyCam wie folgt:

Master-Rx: 01 04 04 00 02 00 00 5A 44

Wert [hex]	Beschreibung
01	EnergyCams Slave Adresse
04	Code für Lesen von aufeinanderfolgenden Input-Registern
04	Anzahl der Bytes der folgenden Daten
00	Daten für erstes Register (höherwertiges Byte)
02	Daten für erstes Register (niederwertiges Byte)
00	Daten für zweites Register (höherwertiges Byte)
00	Daten für zweites Register (niederwertiges Byte)
5A	CRC (höherwertiges Byte)
44	CRC (niederwertiges Byte)

Tabelle 4: Beispiel Antwort für erfolgreiches Lesen von mehreren Input-Registern "AppRevision"

Lese ResultOCR

Es folgt ein Beispiel für das Lesen von drei Input-Registern ab Adresse 0x0043 (0-basiert, "ResultOCRInt" und "ResultOCRFrac" werden zusammen gelesen). Gelesener Wert ist:

0x0043 → 0x0001 (= 1_{10} , höherwertiges von ResultOCRInt)
0x0044 → 0x0D66 (= 3430_{10} , höherwertiges von ResultOCRInt)
0x0045 → 0x0001 (= 1_{10} , ResultOCRFrac)

Der OCR Wert wird berechnet durch: $1 \cdot 65536 + 3430 + 1 \cdot 10^{-1} = 68966.1$

Master-Tx: 01 04 00 43 00 03 41 DF

Wert [hex]	Beschreibung
01	EnergyCams Slave Adresse
04	Code für Lesen von aufeinanderfolgenden Input-Registern
00	Adresse des ersten Input-Registers (höherwertiges Byte)
43	Adresse des ersten Input-Registers (niederwertiges Byte)
00	Anzahl der Register die zu lesen sind (höherwertiges Byte)
03	Anzahl der Register die zu lesen sind (niederwertiges Byte)
41	CRC (höherwertiges Byte)
DF	CRC (niederwertiges Byte)

Tabelle 5: Beispiel für das Lesen von mehreren Input-Registern ResultOCR

Im Erfolgsfall antwortet EnergyCam:

Master-Rx: 01 04 06 00 01 0D 66 00 01 7E 20

Wert [hex]	Beschreibung
01	EnergyCams Slave Adresse
04	Code für Lesen von aufeinanderfolgenden Input-Registern
06	Anzahl der Bytes der folgenden Daten
00	Daten für erstes Register (höherwertiges Byte)
01	Daten für erstes Register (niederwertiges Byte)
0D	Daten für zweites Register (höherwertiges Byte)
66	Daten für zweites Register (niederwertiges Byte)
00	Daten für drittes Register (höherwertiges Byte)
01	Daten für drittes Register (niederwertiges Byte)
7E	CRC (höherwertiges Byte)
20	CRC (niederwertiges Byte)

Tabelle 6: Beispiel Antwort für erfolgreiches Lesen von mehreren Input-Registern ResultOCR

Unterstützte Holding-Register

Obwohl Modbus Holding-Register als read-write definiert, sind die meisten von EnergyCam write-only. Das Lesen von write-only Holding-Registern führt zu einer Modbus-Fehler Antwort (Exception). Lesbare Holding-Register sind mit read-only oder read-write markiert.

Modbus-Adresse 0-basiert	Modbus-Adresse 1-basiert	Block Name	Block Zugriff	Register Name [16 Bits]	Beschreibung
0x0000	0x0001	SetTime	write-only	SetTime1	SetTime[31..16], Zeit seit UNIX Epoche in Sekunden (UTC), 0 bedeutet 1.1.1970 00:00:00, 1360751350 bedeutet 13. Feb 2013 10:29:10
			write-only	SetTime0	SetTime[15..0]
0x0002	0x0003	SetOMSCfg	write-only	SetOMSCfg	Zähler und wireless M-Bus Konfiguration [7:0] Device Typ (2: Strom; 3: Gas; 7: Wasser) [8:12] Reserviert, schreibe 0 [13] wM-Bus aktiviert ⁴ [14] wM-Bus verschlüsselt

Modbus-Adresse 0-basiert	Modbus-Adresse 1-basiert	Block Name	Block Zugriff	Register Name [16 Bits]	Beschreibung
					[15] wM-Bus Installation Modus ⁴ (0: auto; 1: manuell) Es werden nur Device Typen 2, 3 and 7 unterstützt.
0x0003	0x0004	Explfc	read-write	Explfc1	Expansion Interface Serielle Konfiguration [3:0] Baud-Rate, kodiert als: 0 115200, 1 600 2 1200, 3 2400 4 4800, 5 9600 6 14400, 7 19200 8 38400, 9 57600 10 115200 [5:4] Reserviert, schreibe 0 [7:6] Parität 0 Gerade 1 Keine 2 Ungerade [11:8] Reserviert, schreibe 0 [13:12] Stopp-Bits 0 Eins 1 Zwei [15:14] Reserviert, schreibe 0
				Explfc0	Expansion Interface Protokoll [7:0] Slave Adresse (Primary Address) Modbus 1..247 M-Bus 0..250 (0 reagiert auf alle Adressen) [8] AutoPowerDown (wenn aktiv Wake-Up nötig) [11:9] Reserviert, schreibe 0 [15:12] Protokoll, kodiert als: 0 Modbus 2 M-Bus Explfc1 und Explfc0 müssen als einzelner Modbus-Zugriff geschrieben werden. Es wird eine Modbus-Antwort mit der aktuellen Konfiguration zurückgeschickt und die Änderungen sofort danach vollzogen.
Lücke	**Lücke**				
0x0007	0x0008	TestReadOnly1 ⁵	read-only	TestReadOnly1	Test Register für Modbus Master Tests: Enthält 0xDEAD = 57005
0x0008	0x0009	TestReadOnly0 ⁵	read-only	TestReadOnly0	Test Register für Modbus Master Tests: Enthält 0xBEEF = 48879. Für 32 Bit Zugriffe (d.h. unsigned DWORD) sollten zwei 16 Bit Register ab TestReadOnly1 gelesen werden. Das Ergebnis sollte

4 Änderungen dieser Werte werden bei der nächsten OCR Installation verwendet

5 Die Testregister sind ab ProtokollVersion 5 verfügbar

Modbus-Adresse 0-basiert	Modbus-Adresse 1-basiert	Block Name	Block Zugriff	Register Name [16 Bits]	Beschreibung
					0xDEADBEEF = 3735928559 sein. Im Falle von 0xBEEFDEAD = 3203391149 sollten die beiden 16 Bit Wörter vertauscht werden („word swap“)
0x0009	0x000A	TestReadWrite1 ⁵	read-write	TestReadWrite1	Test Register für Modbus Master Tests: Enthält 0xFA51 = 64081. Kann auch geschrieben werden, nach Neustart geht Wert verloren.
0x000A	0x000B	TestReadWrite0 ⁵	read-write	TestReadWrite0	Test Register für Modbus Master Tests: Enthält 0xFFDD=65501. Kann auch geschrieben werden, nach Neustart geht Wert verloren. Für 32 Bit Zugriffe (d.h. unsigned DWORD) sollten zwei 16 Bit Register ab TestReadWrite1 gelesen werden. Das Ergebnis sollte 0xFA51FFDD = 4199677917 sein. Im Falle von 0xFFDDFA51 = 4292737617 sollten die beiden 16 Bit Wörter vertauscht werden („word swap“)
Lücke	**Lücke**				
0x001F	0x0020	ActionOCR-Installation	write-only	ActionOCR-InstallationTO	Timeout in [s]
0x0020	0x0021		write-only	ActionOCR-Installation	0: Kommando wird ignoriert 1: Installation wird gestartet, Aktion ist beendet, wenn EnergyCamStatus = ActionCompleted
0x0021	0x0022	ActionOCR	write-only	ActionOCR	0: Kommando wird ignoriert 1: Foto wird gemacht und OCR gestartet, Aktion ist beendet, wenn EnergyCamStatus = ActionCompleted
Lücke	**Lücke**				
0x0024	0x0025	ActionPowerDown	write-only	ActionPowerDown	0: Kommando wird ignoriert 1: Solange keine Aktion läuft wird Stromsparmmodus sofort aktiviert, Ende der Aktion kann nicht kontrolliert werden, da dieses ein erneutes Wake-up zur Folge hätte. Es sollte vorher kontrolliert werden ob StatusEnergyCam != ActionOngoing, sonst wird ActionPowerDown nicht durchgeführt und führt zu einer Modbus Exception.
Lücke	**Lücke**				
0x0034	0x0035	OCRConfig	read-write	OCRConfig	OCR Konfiguration (es ist ein read-modify-write durchzuführen, um die reservierten Bitfelder zu erhalten) [0] OCR Option: Lese Nachkommastellen ⁴ [5:1] Reserviert: Aktueller Wert muss beibehalten werden [10:6] OCR Ablese Timer: Wert in Minuten ⁴ [1..15] [15:11] OCR maximales Inkrement pro Ablesung
Lücke	**Lücke**				
0x0038	0x0039	OCRIncrConfig	read-write	OCRIncrConfig	OCR maximales Inkrement pro Stunde [11:0] Inkrement pro Stunde

Modbus-Adresse 0-basiert	Modbus-Adresse 1-basiert	Block Name	Block Zugriff	Register Name [16 Bits]	Beschreibung
					[15:12] wird immer als 0x00 gelesen. Beim Schreiben muss [15:12] = 0xA sein.
0x0039	0x003A	OCRReadIncrConfig	read-write	OCRReadIncrConfig	OCR maximales Inkrement pro Ablesung (wird ausgewertet wenn OCRConfig [15:11] = 0x1F)
0x003A	0x003B	OCRReadReadTimer	read-write	OCRReadTimerConfig	OCR Ablese Timer: Wert in Minuten (wird ausgewertet wenn OCRConfig [10:6] = 0x1F)
Lücke	**Lücke**				
0x003C	0x003D	OCRConfigMeter	read-write	OCRConfigMeter	OCR Konfiguration spezielle Zähler [7:0] reduzierte Ablesebreite 0: 100% 1: 90% ... [15:8] spezielle Zähler 0: Standard 1: Schwarz auf Weiß mit schwarzen Lücken ...
Lücke	**Lücke**				
0x7FFF	0x8000	UpdateChunk	write-only	UpdateChunkStartAddr1	Update Chunk. Ein Chunk muss in einem einzelnen Frame geschrieben werden mit bis zu 122 Words (je 16 bit). Byte Adresse (32 bit) der ersten Binär-Daten gefolgt von Data relativ zum ersten Byte des Update-Images. Das allererste Byte des Update-Images hat die Adresse 0x00000000. Anzahl der ChunkData in Bytes muss ein Vielfaches von 4 sein, mit Ausnahme des allerletzten Chunks. Maximum ist 120 words = 240 Bytes (240 mod 4=0)
				UpdateChunkStartAddr0	
				UpdateChunkData0	
				UpdateChunkData...	
0x8078	0x8079			UpdateChunkData119	

Tabelle 7: Unterstützte Holding-Register

Beispiele

Get OCRConfig

Es folgt ein Beispiel für das Lesen von einem Holding-Register ab Adresse 0x0034 (0-basiert, "OCRConfig") welches die OCR Konfiguration zurückgibt (im Beispiel „Lese Nachkommastelle“, „15 Minuten Ablese Timer“:

0x0034 → 0x43C9 → 01 00 00 11 11 00 10 01 → **Lese Nachkommastelle**, **15 Minuten Ablese Timer**

Master-Tx: 01 03 00 34 00 01 C5 C4

Wert [hex]	Beschreibung
01	EnergyCams Slave Adresse
03	Code für read multiple Holding-Registers
00	Adresse des ersten zu lesenden Holding-Register (höherwertiges Byte)
34	Adresse des ersten zu lesenden Holding-Register (niederwertiges Byte)
00	Anzahl der zu lesenden Register (höherwertiges Byte)
01	Anzahl der zu lesenden Register (niederwertiges Byte)
C5	CRC (höherwertiges Byte)
C4	CRC (niederwertiges Byte)

Tabelle 8: Beispiel zum Lesen von mehreren Holding-Registern

Im Erfolgsfall antwortet EnergyCam wie folgt:

Master-Rx: 01 03 02 43 C9 49 08 22

Wert [hex]	Beschreibung
01	EnergyCams Slave Adresse
03	Code für read multiple Holding-Registers
02	Anzahl der Bytes der folgenden Daten
43	Daten für erstes Register (höherwertiges Byte)
C9	Daten für erstes Register (niederwertiges Byte)
08	CRC (höherwertiges Byte)
22	CRC (niederwertiges Byte)

Tabelle 9: Beispielantwort für erfolgreiches Lesen von mehreren Holding-Registern

Set ActionPowerDown

Es folgt ein Beispiel für das Schreiben eines einzelnen Holding-Registers ab Adresse 0x0024 (0-basiert, "ActionPowerDown"). Diese Aktion versetzt EnergyCam in den Stromsparmodus.

0x0024 ← 0x01

Master-Tx: 01 06 00 24 00 01 08 01

Wert [hex]	Beschreibung
01	EnergyCams Slave Adresse
06	Code für das Schreiben eines einzelnen Holding-Registers
00	Adresse des Holding-Registers (höherwertiges Byte)
24	Adresse des Holding-Registers (niederwertiges Byte)
00	Daten (höherwertiges Byte)
01	Daten (niederwertiges Byte)
08	CRC (höherwertiges Byte)
01	CRC (niederwertiges Byte)

Tabelle 10: Beispiel für das Schreiben auf ein Holding-Register

Im Erfolgsfall antwortet EnergyCam wie folgt:

Master-Rx: 01 06 00 24 00 01 08 01

Wert [hex]	Beschreibung
01	EnergyCams Slave Adresse
06	Code für das Schreiben eines einzelnen Holding-Registers
00	Adresse des Holding-Registers (höherwertiges Byte)
24	Adresse des Holding-Registers (niederwertiges Byte)
00	Daten (höherwertiges Byte)
01	Daten (niederwertiges Byte)
08	CRC (höherwertiges Byte)
01	CRC (niederwertiges Byte)

Tabelle 11: Beispielantwort für erfolgreiches Schreiben eines Holding-Registers

Set ActionOCRInstallation

Es folgt ein Beispiel für das Schreiben mehrerer Holding-Register ab Adresse 0x001F (0-basiert, "ActionOCRInstallation"). Diese Aktion startet die OCR Installation (Timeout 100 = 0x64 seconds).

0x001F ← 0x64

0x0020 ← 0x01

Master-Tx: 01 10 00 1F 00 02 04 00 64 00 01 32 FC

Wert [hex]	Beschreibung
01	EnergyCams Slave Adresse
10	Code für das Schreiben mehrerer Holding-Register
00	Adresse des ersten Holding-Registers (höherwertiges Byte)
1F	Adresse des ersten Holding-Registers (niederwertiges Byte)
00	Anzahl der Register, die zu schreiben sind (höherwertiges Byte)
02	Anzahl der Register, die zu schreiben sind (niederwertiges Byte)
04	Anzahl der Bytes der folgenden Daten
00	Daten für erstes Register (höherwertiges Byte)
64	Daten für erstes Register (niederwertiges Byte)
00	Daten für zweites Register (höherwertiges Byte)
01	Daten für zweites Register (niederwertiges Byte)
32	CRC (höherwertiges Byte)
FC	CRC (niederwertiges Byte)

Tabelle 12: Beispiel für das Schreiben mehrerer Holding-Register

Im Erfolgsfall antwortet EnergyCam wie folgt:

Master-Rx: 01 10 00 1F 00 02 70 0E

Wert [hex]	Beschreibung
01	EnergyCams Slave Adresse
10	Code für das Schreiben mehrerer Holding-Register
00	Adresse des ersten Holding-Register (höherwertiges Byte)
1F	Adresse des ersten Holding-Register (niederwertiges Byte)
00	Anzahl der Register, die zu schreiben sind (höherwertiges Byte)
02	Anzahl der Register, die zu schreiben sind (niederwertiges Byte)
70	CRC (höherwertiges Byte)
0E	CRC (niederwertiges Byte)

Tabelle 13: Beispielantwort für erfolgreiches Schreiben mehrerer Holding-Register

Fehlerfälle

Hier ein Beispiel einer Antwort von EnergyCam, wenn auf eine unbekannte Adresse zugegriffen wird. Es folgt der Versuch auf zwei nicht existierende Holding-Register, die ab Adresse 0x00C8 (0-basiert) beginnen, zu schreiben.

0x00C8 ← 1
0x00C9 ← 2

Master-Tx: 01 10 00 C8 00 02 04 00 01 00 02 2E 58

Master-Rx: 01 90 02 CD C1

Wert [hex]	Beschreibung
01	EnergyCams Slave Adresse
90	Fehler-Code [7] Immer 1 [6..0] Funktionsnummer des fehlgeschlagenen Zugriffs (hier 0x10 für Schreiben mehrerer Holding-Register)
02	Ausnahme-Code für ungültige Datenadresse ("ILLEGAL DATA ADDRESS") bezugnehmend auf offizielle „Modbus Exception Codes“ 1 Nicht erlaubte Funktion 2 Nicht erlaubte Register-Adresse 3 Nicht erlaubtes Datum 4 Fehler (u.a. nicht erlaubtes Datum, das geschrieben werden sollte)
CD	CRC (höherwertiges Byte)
C1	CRC (niederwertiges Byte)

Tabelle 14: Beispiel Antwort für Schreiben auf eine unbekannte Holding-Register Adresse

Automatisches Verhalten

Üblicherweise startet EnergyCam keine Aktionen autonom, es wird nur auf Kommandos, die vom Master gesendet werden, reagiert.

Ausnahmen sind:

- Die OCR startet automatisch alle $t_{OCRDefault}$ Minuten solange in diesem Intervall diese Aktion nicht schon durch den Master angefordert wurde. Dieses wird für interne Fehlerkorrekturen benötigt.

- Der Stromsparmodes wird automatisch nach t_{modbusTO} aktiviert. Diese Periode wird nach einem Wake up (wenn EnergyCam wirklich im Stromsparmodes war) neu gestartet oder wenn ein gültiger Modbus Frame empfangen wurde (d.h. dass kein CRC-Fehler auftrat).

Symbol	Zeit	Beschreibung
t_{modbusTO}	5 Sekunden	Nach dieser Zeit ohne Modbus-Kommunikation schaltet EnergyCam automatisch in den Stromsparmodes
$t_{\text{OCRDefault}}$	15 Minuten	Nach dieser Zeit wird eine automatische OCR Ablesung durchgeführt.

Tabelle 15: Timings für automatisches Verhalten

Historie

Datum	Autor	Version	Änderungen
17. Apr 14	SPR	1.11	Erste deutsche Fassung
22. Apr 14	ARG	1.12	Verbesserung
28. Apr 14	SPR	1.13	Vereinheitlichung der Namen
28. Jul 14	SPR	1.16	Beispiel für Lesen von ResultOCR hinzugefügt
31. Jul 14	SPR	1.17	OCR Ablese Timer geändert von [2..15] zu [1..15]
15. Sep 14	SPR	1.18	ProtocolVersion von 2 auf 4 geändert
29. Okt 14	SPR	1.19	0-basierte Modbus Registeradressen hinzugefügt
30. Okt 14	SPR	1.20	Testregister hinzugefügt
20. Apr 15	SPR	1.21	Ausnahme Code hinzugefügt
27. Apr 15	SPR	1.22	Input Register ResultOCR64 hinzugefügt
30. Apr 15	CRG	1.23	Input Register OCRInfoRed, Holding Register OCRIncrConfig, OCRReadIncrConfig hinzugefügt
08. Jul 15	SPR	1.24	Fehlenden Fehler-Code 4 zu Tabelle 14 hinzugefügt
19. Nov 15	SPR	1.25	Einschränkung von ResultOCRfrac näher definiert. In ResultOCRIntChar unbenutzte Stellen sind mit '-' gefüllt.

Tabelle 16: Historie