



Profitcenter
Lötwerkzeuge und Inspektionssysteme

FUNKTIONSBESCHREIBUNG SERIELLES INTERFACE *i-CON*



Diese Unterlage ist unser Eigentum, jede Vervielfältigung, Verwertung oder Mitteilung an dritte Personen ist strafbar, verpflichtet zu Schadenersatz und wird gerichtlich verfolgt. Alle Rechte für den Fall einer Patenterteilung oder GM – Eintrag behalten wir uns vor. Die Unterlage ist auf Verlangen jederzeit zurückzugeben.

Inhaltsverzeichnis

1	FUNKTIONSÜBERSICHT	3
1.1	ÜBERTRAGUNGSFORMAT	4
1.1.1	Datenformat	4
1.1.2	Schnittstellenparameter	4
2	KOMMUNIKATIONS-PROTOKOLL	5
2.1.1	Geräteadresse	5
2.1.2	Funktionscode	6
2.1.3	Endekennung	6
2.1.4	Datenadresse	7
2.1.5	Datenlänge	7
2.1.6	Datenblock	7
2.1.7	Checksumme	7
2.2	BEISPIELE ZUR DATENÜBERTRAGUNG	8
2.2.1	Register lesen	8
2.2.2	Register schreiben	8
3	DATENBEREICHE	9
3.1	AKTUELLE DATEN	9
3.2	STANDBYFUNKTION	9
3.2.1	i-Con	10
3.2.2	i-Con2	10
3.3	TOOLUMSCHALTUNG	10
3.4	AKTUELLE TOOLPARAMETER	10
3.5	FEHLERBEARBEITUNG	11
3.6	TOOL - PARAMETER	12
3.6.1	Offset Tabelle	13
3.6.2	Powerlevel	13
3.6.3	Standbyzeit	13
3.7	SYSTEMPARAMETER	14
3.7.1	Wechsel der Temperatureinheit	14
3.8	DIREKTER ZUGRIFF AUF DEN EEPROM SPEICHER	15
3.8.1	Parameter	15
3.9	SONSTIGE BEREICHE	15
3.9.1	Applikationskennung	15

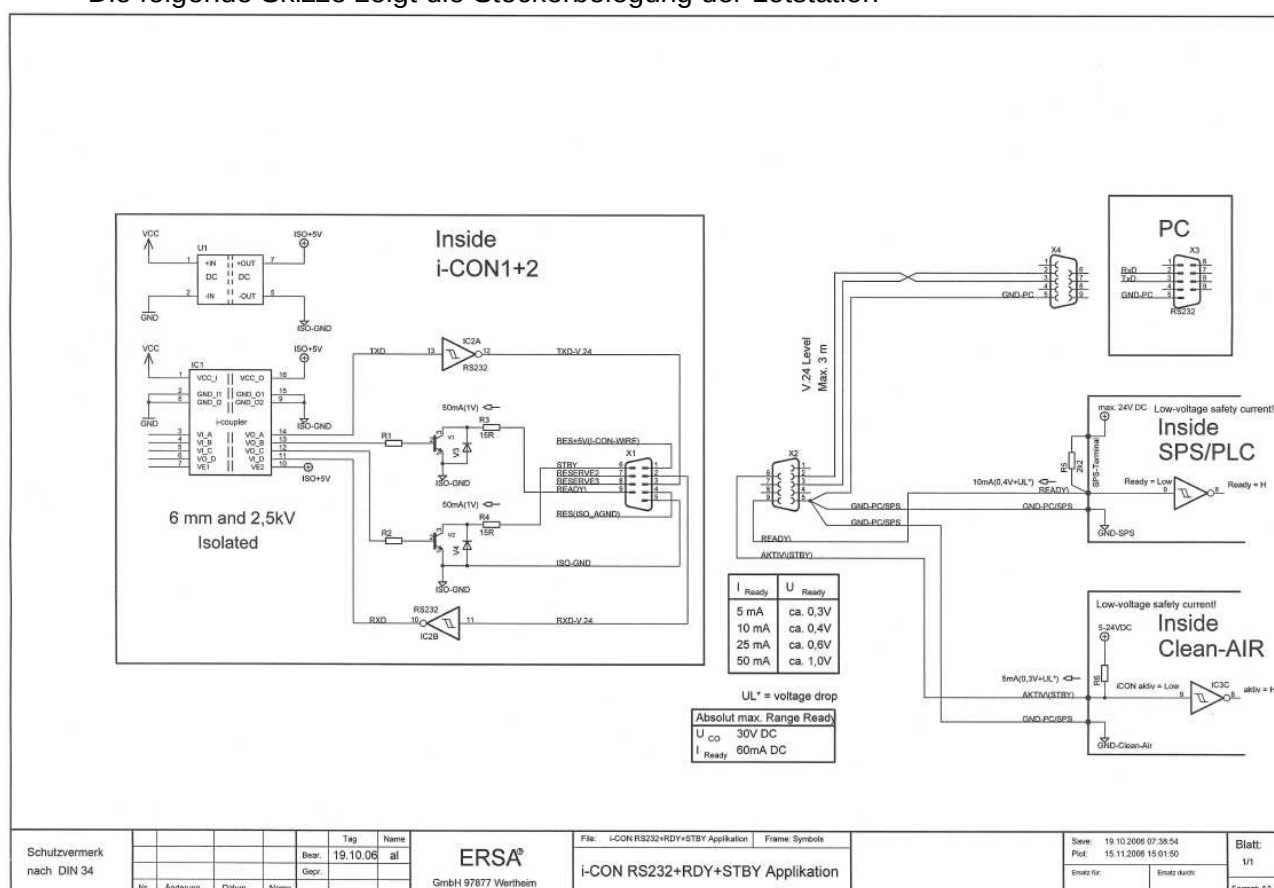
1 Funktionsübersicht

Diese Dokumentation beschreibt das serielle Interface der i-Con (i-Con2) Lötstation. Über eine RS232 Verbindung, die optional für die i-Con Lötstation verfügbar ist, kann mittels einem entsprechenden Kommunikations-Protokoll ein Datenaustausch erfolgen. Dabei ist die Lötstation immer als Slave zu betrachten.



Die Steckerbelegung des 9 poligen D-Sub Stecker der Lötstation entspricht nicht dem RS232 Standard! Eine Verbindung zwischen Master und Slave (Lötstation) darf nur über ein speziell konfektioniertes Kabel hergestellt werden. Herkömmliche RS232 Leitungen können zu Zerstörungen beim Master führen.

Die folgende Skizze zeigt die Steckerbelegung der Lötstation



1.1 Übertragungsformat

Das Übertragungsformat der Lötstation ist fest eingestellt und kann nicht verändert werden. Die Datenübergabe erfolgt nach dem Master/Slave Prinzip ohne zusätzliche Steuersignale. Das heißt, nur auf Anforderung des Masters antwortet der Slave.

1.1.1 Datenformat



Alle Daten sind im sog. Intel-Format zu interpretieren. Das bedeutet, dass die Adresse immer auf das low-Byte und dann auf das high-Byte zeigt.

1000d → E803h

Bestimmte Daten werden als sog. "Real"-Werte abgebildet. Diese Daten sind vom Typ "Float" mit einer Länge von 4 Datenbyte nach IEEE 754.

Beispiel:

1.334 → 0x3FAAC083 → 83C0AA3F (Intel-Format)

Unter <http://babbage.cs.qc.edu/IEEE-754> besteht die Möglichkeit entsprechende Beispiele umwandeln und anzeigen zu lassen.

1.1.2 Schnittstellenparameter

Datenbits	8
Paritätsprüfung	NO
Stoppbit	1
Geschwindigkeit	57600 bd

Die Schnittstelle des Masters muss als COMx: 57600,8,n,1 initialisiert werden.

2 Kommunikations-Protokoll

Die Kommunikation erfolgt über ein spezielles Protokoll, das dem MODBUS Protokoll ähnlich ist. Vom Master wird ein Anforderungstelegramm abgesetzt, das vom Slave ausgewertet und mit den entsprechenden Daten sofort beantwortet wird.

Das Telegramm ist so aufgebaut, dass der Slave anhand eines Funktionscode und der Datenadresse ermittelt ob Daten beschrieben oder gelesen werden sollen.

Die Übertragungstelegramme vom Master (Anforderungstelegramm) und die entsprechende Antwort vom Slave (Antworttelegramm) sind gleich aufgebaut. Im Antworttelegramm auf den Schreibbefehl wird nur die Datenadresse (Byte 3,4) ausgegeben. Synchronisiert wird das Ende eines Telegramms über die Byteanzahl, die als Byte 2 übertragen wird.

Aufbau eines Telegramms:

Byte 0	1	2	3	4	5	6	7	8	9
ID	FC	LEN	Adresse		Anzahl	Daten		Checksumme	

ID	Geräteadresse derzeit nur 01 möglich
FC	Funktionscode Art/Typ der Datenübertragung
LEN	Länge (Byteanzahl) bestimmt die Anzahl der Bytes im Übertragungstelegramm
Adresse	Adresse der Daten "symbolisiert" die Startadresse ab der Daten (byte) beschrieben/gelesen werden.
Anzahl	Anzahl der folgenden Datenbytes
Daten	Datenblock
Checksumme	Prüfsumme zur Überprüfung der empfangenen Daten.

2.1.1 Geräteadresse

Über die Geräteadresse erfolgt die Auswahl zum entsprechenden Controller-Modul (Lötstation).
Derzeit ist nur die Geräteadresse "01" erlaubt!

2.1.2 Funktionscode

Das Byte bestimmt welche Art von Funktion an den Datenaustausch geknüpft ist. Derzeit wird nur der Funktionscode zum Lesen und Schreiben von sog. Registerdaten unterstützt.

FC Bit -							
7	6	5	4	3	2	1	0
Error	write	read		Funktionscode			

Code (HEX)	R/W	Bedeutung
20...20	R	Nicht definiert
2F	R	read registers
40...4E	W	Nicht definiert
4F	W	write registers

Bei der Antwort wird der FC vom Slave mit übertragen. Das gesetzte Bit 7 signalisiert dem Master, dass der Befehl nicht ausgeführt werden konnte. Im Adressbyte des Antwortstring wird in Byte 3 der entsprechende Error-Code übergeben.

Error-Code	Bedeutung
1	Funktionscode nicht definiert
2	Adresse nicht bekannt
3	Datenwert nicht erlaubt
4	CRC Fehler

2.1.3 Endekennung

Im Übertragungstelegramm wird die Länge (Anzahl) des Telegramms mit übertragen. Der Schnittstellentreiber synchronisiert anhand des Bytes, nach wie viel Zeichen das Telegramm vollständig übertragen wurde und die Auswertung (Interpreter) ausgelöst werden kann.

Das Längenbyte bestimmt die Anzahl der Bytes, die dem Byte 3 bis zum letzten Byte der Checksumme folgen. Die Gesamtlänge eines Telegramms ergibt sich somit aus dem Längenbyte + 3 Byte (ID, Funktionscode, LEN).

Pausen (>250ms) zwischen den einzelnen Bytes führen dazu, dass das Telegramm verworfen wird.

2.1.4 Datenadresse

Die Datenadresse ist eine symbolische Adresse, die von der Applikation (Lötstation) definiert wurde und für einen entsprechenden Inhalt steht.

Die Adresse zeigt immer auf das erste Datenbyte, dem dann die "ANZAHL" Datenbytes folgen. Anhand der Datentabelle ist festgelegt wie die Daten zu interpretieren sind (Bedeutung und Format).

Der Adressebereich zur symbolischen Datenadresse reicht von 0x0000-0x7FFF, wobei nur die aufgeführten Datenadressen definiert sind. Adressen über den Bereich zeigen auf physikalische Bereiche (RAM und EEPROM) und können nur eingeschränkt verwendet werden.

Beispiel:

Adresse	Name	Datentyp	Skalierung
0x0100	Isttemperatur 1	Integer	°C/°F
0x0102	Status	Byte	
0x0103			

2.1.5 Datenlänge

Die Datenlänge bestimmt die Anzahl der folgenden Datenbytes, die beim Lesen angefordert und beim Schreiben übertragen werden.

Maximale Anzahl von Datenbytes: 16

2.1.6 Datenblock

Im Datenblock befindet sich die Anzahl der Datenbytes, die über die Datenlänge festgelegt sind. Der Datenblock sollte auf eine maximale Anzahl von Datenbytes beschränkt werden.

2.1.7 Checksumme

Die Checksumme dient zur Überwachung der Datensicherheit bei der Übertragung. Der Algorithmus zur Ermittlung der Checksumme lehnt sich an das Standardverfahren "CRC16". Das folgende Beispiel zeigt im *C-Kontext* wie die Summe gebildet wird:

```
int crc16(char *ptr, int count)
{
    int crc;
    char i;

    crc = 0;
    while (--count >= 0)
    {
        crc = crc ^ (int) *ptr++ << 8;
        i = 8;
        do
        {
            if (crc & 0x8000)
                crc = crc << 1 ^ 0x1021;
            else
                crc = crc << 1;
        } while(--i);
    }
    return (crc);
}
```

Die Prüfsumme wird über Byte 0 (ID) bis zum letzten Datenbyte ermittelt und wird als "unsigned integer" am Ende des Telegramms übertragen.

Beispiel: [012F050010045FD7]

ERSA GmbH

A **Kurtz** Company

2.2 Beispiele zur Datenübertragung

In den folgenden Beispielen sind alle aufgeführten Adressen und Daten in hexadezimaler Form dargestellt.

2.2.1 Register lesen

Über den folgenden String werden entsprechend der Tabelle (symbolische Datenadresse) die Ist- (0x0900) und Solltemperatur (0x0902), sowie der Status und der Tool index vom Slave (ID=1) gelesen.

Master:

ID	Code	LEN	Adresse	Anzahl	Checksum
01	2F	05	00 09	06	F6 4E

Slave:

					Adressindex							
ID	Code	LEN	Adresse	Anzahl	0	1	2	3	4	5	Checksum	
01	2F	0B	00 09	06	FB	00	68	01	11	0B	22	5C
					250		360		17	11		

Das Beispiel zeigt die Anforderung der aktuellen Daten der iCon.

Registeradr. 0x900: akt. Istwert 250°C
 0x902: Sollwert 360°C
 0x904: Status bit0=1 Standby
 bit4=1 aktives Tool (nur wenn iCon2)
 0x905: Toolindex 11=i-Tool erkannt

2.2.2 Register schreiben

Im Beispiel wird ein Parameter mit der Adresse 0x6023 und dem Wert 50 beschrieben.

Master:

ID	Code	LEN	Adresse	Anzahl	Daten	Checksum	
01	4F	06	23 60	01	32	2E	C8
					50		

Slave

ID	Code	LEN	Adresse	Checksum	
01	4F	04	23 60	F0	F3

3 Datenbereiche

In den nachfolgend beschriebenen Datenbereichen sind die Daten definiert, die vom Master aus der Lötstation gelesen und beschrieben werden können.



Werden vom Master Daten der Lötstation verändert, muss darauf geachtet werden, dass die Daten die zur Lötstation geschickt innerhalb der erlaubten Grenzen liegen. Der Master hat die Verantwortung über die Daten, die gesendet werden. Gültigkeitsprüfungen (Min/Max-Begrenzungen) müssen vom Master realisiert werden.

Die Datenbereich sind immer für beide Lötstationen (i-Con/i-Con2) beschrieben. Ist nur die i-Con angeschlossen, sind die Bereich für die i-Con2 nicht gültig.

Sofern nicht explizit vermerkt, sind Temperaturwerte in den Datenbereichen in der eingestellten Temperatureinheit (Systemparameter) zugeordnet.

3.1 Aktuelle Daten

In diesem Datenbereich sind die wichtigsten aktuellen Daten der Station definiert. Dabei wird der aktuelle Istwert, Sollwert und der Status angezeigt.

Linke Seite (Tool1) 0x900
Rechte Seite (Tool2) 0x920 (nur i-Con2)

Offset	Daten	Format
0x00	Aktueller Istwert	INT
0x02	Sollwert	INT
0x04	Status b0 Standby b1 Temperatur innerhalb des Prozessfenster (ready) b4 Tool ist ausgewählt (iCon2) b7 Fehler aktiv	Bitfeld[8]
0x05	Toolindex 0 kein Tool erkannt/angeschlossen 1 Chip tool 2 Micro tool 3 Tech tool 4 X-Tool 6 Power tool 11 I-Tool 12 I-Set tool	Char

Wird im Status ein Fehler erkannt, muss der Anwender (Master) den Fehlerstatus auslesen, um den Zustand näher zu definieren.

3.2 Standbyfunktion

Ist die Funktion auf der Station eingestellt, wird das angeschlossene Tool nach einer definierten Zeit auf die Standbytemperatur geregelt. Je nach Typ wird das Tool wieder aktiviert (i-Tool Bewegung) und auf die Solltemperatur geregelt.

Über das serielle Interface hat der Master die Möglichkeit direkt das Tool in Standby zu schalten oder aus diesem Zustand zu holen.

Die Funktionsweise zwischen i-Con und i-Con2 unterscheiden sich dabei und werden nachfolgend getrennt beschrieben.

3.2.1 i-Con

Über die Registeradresse 0x100B kann die Standbyfunktion direkt beschrieben werden:

- Standby einschalten 0x100B=0x80
- Standby abschalten 0x100B=0x00

3.2.2 i-Con2

Um die Funktion bei der i-Con2 zu nutzen, muss der Master die Aktivität des jeweiligen Tools (linke/rechte Seite) beachten. Ist Tool2 (rechte Seite) aktiv, ist Tool1 (linke Seite) in Standby. Soll nun Tool1 aktiviert werden, muss dies über die Toolumschaltung erfolgen, mit der Tool1 aktiviert und Tool2 in Standby geschaltet wird.

Somit darf die direkte Umschaltung der Standbyfunktion nur zum aktiven Tool erfolgen.

- | | Tool1 | Tool2 | |
|-----------------------|--------|--------|-------|
| • Standby einschalten | 0x100B | 0x200B | =0x80 |
| • Standby abschalten | 0x100B | 0x200B | =0x00 |

3.3 Toolumschaltung

In aller Regel erfolgt die Toolumschaltung (nur bei i-Con2) durch den Bediener (i-Tool Bewegung, Tastendruck). Über das serielle Interface besteht auch die Möglichkeit zwischen den beiden Tools zu wechseln. Der Master kann über die Registeradresse 0x0DFF oder den Status (siehe Aktuelle Daten) erkennen, welche Seite aktiv ist.

0x0DFF	0x00 (Tool1)	0x80 (Tool2)
	0x904.4=1	0x924.4=1

Mit einem beliebigen Schreibbefehl kann immer zwischen den beiden Buchsen gewechselt werden.

3.4 Aktuelle Toolparameter

Der folgende Datenbereich zeigt die aktuellen Parameter des angeschlossenen Tools. Dieser Datenbereich darf nicht direkt vom Master geändert werden.

Basisadresse Tool1 0x1000 (i-Con) Tool2 0x2000 (nur i-Con2)

Offset	Daten	Format
0x0020	Solltemperatur	UINT
0x0022	Kalibrierung	SCHAR
0x0023	Offset	CHAR
0x0024	Energie (Powerlevel)	CHAR
0x0025	Standbyzeit	CHAR
0x0026	Standbytemperatur	UINT

Bei den Parametern unbedingt die Hinweise im Kapitel Tool - Parameter (Seite 12) beachten.

3.5 Fehlerbearbeitung

Die Station überwacht die unterschiedlichen Betriebszustände auf Fehlerzustände. Wird ein Fehler erkannt, wird der Fehler angezeigt und im Datenbereich abgebildet. Dabei ist bei der Visualisierung im wesentlichen das Error Interface und der Fehlercode relevant.

Fehler können über das entsprechende Steuerbit (100.0=1) quittiert werden, sofern die Fehlerursache nicht mehr besteht.

Adresse	Daten	Format
0x0100	Error-Interface b0 Fehler quittieren (ACK) b4 b1 b5 b2 Fehler an Tool1(=0) / Tool2(=1) b6 Toggle-Flag (interne Nutzung) b3 Stationsfehler b7 Error aktiv	Bitfeld[8]
0x0101	Akt. Tabellen Index	Char
0x0102	Fehlertextnummer	Char
0x0103	Fehlercode	Char
0x0104	CRC Fehler System-Daten 1 Systemparameter 2 Kalibrierdaten	Char
0x0105	CRC Fehler Tool-Daten 1 Chip tool 5 Power tool 2 Micro tool 6 reserviert 3 Tech tool 7 iTool 4 X-Tool	char

Fehlertabelle

Code	LCD Text	Ursache	Maßnahmen
2	KALIBRIERUNG FEHLERHAFT	Kalibrierdaten zerstört	Station muss neu Werkskalibriert werden
4	RT-KOMPENSATION	Messwert des KTY Sensor (Klemmstellentemperatur) nicht ok	Kolben überprüfen und Station überprüfen lassen (Vertretung oder Werkseitig)
6	UNDEF. TOOL	Angeschlossenes Tool konnte nicht korrekt erkannt werden	Kolben überprüfen und Station überprüfen lassen (Vertretung oder Werkseitig)
7	THERMOELEMENT HE GESTECKT / THERMOELEMENT?	Fehler Temperatursensor	Kolben überprüfen und Station überprüfen (siehe Betriebsanleitungen) Ggf. Heizelement nicht korrekt im iTool gesteckt
8	HEIZKÖRPER	Heizkörper defekt	Kolben überprüfen und Station überprüfen (siehe Betriebsanleitungen)
9	DATENVERLUST	Parameter / Konfigurationsdaten zerstört	Station über Default-Einstellungen rücksetzen
10	ICON SYSTEMFEHLER	Hardwarefehler auf der Regelkarte	Regelkarte austauschen
98	Netzfrequenz ZU GROSS/KLEIN	Netzfrequenz außerhalb des Frequenzbereiches ($\pm 10\%$)	
99	ICON ÜBERLASTSCHUTZ	Station wurde überlastet	Icon1: Station austauschen Icon2: abkühlen

3.6 Tool - Parameter

Die Station speichert die Parameter toolspezifisch ab. Das heißt, jedes Tool hat einen eigenen Datensatz, der von der Station geladen wird, sobald das Tool erkannt wird. Bei der i-Con2 wird noch zusätzlich die Buchsen-seite unterschieden.

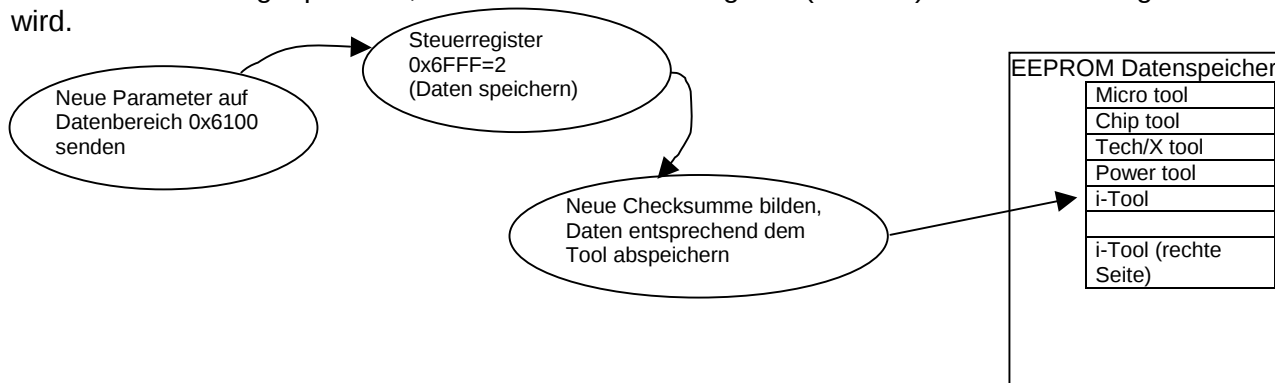
Tech tool und X-Tool werden im Datenbereich nicht unterschieden werden.

Über den nachfolgenden Datenbereich können die Parameterdaten des angeschlossenen Tools bearbeitet werden. Dabei muss der Master die entsprechenden Daten senden und anschließend über ein spezielles Register den Datensatz speichern.

ICon / ICon2 linke Seite	ICon2 rechte Seite	Daten	Format
0x6100	0x6120	Solltemperatur	150-450°C / 300-842°F
0x6102	0x6122	Kalibrierung	± 50°C / ± 120°F
0x6103	0x6123	Offset (siehe Offset Tabelle)	CHAR (1-25)
0x6104	0x6124	Energie (Powerlevel) (siehe Powerlevel)	CHAR (1-3)
0x6105	0x6125	Standbyzeit (siehe Standbyzeit)	CHAR (0-60)
0x6106	0x6126	Standbytemperatur	150-300°C / 300-575°F

Sind die Daten vom Master übertragen, muss der Datensatz auf der Lötstation noch im EEPROM gespeichert werden. Das Speichern (mit Bildung der neuen Checksumme) erfolgt über ein Speicherbefehl auf das Steuerregister 0x6FFF.

Die Daten werden gespeichert, wenn auf das Steuerregister (0x6FFF) der Code 0x02 geschrieben wird.



Sollen Parameter direkt geändert werden, muss der Master direkt die Daten auf dem EEPROM-Speicher ablegen. Dabei muss auch die Checksumme im Datensatz vom Master erzeugt werden. (siehe Direkter Zugriff auf den EEPROM Speicher / Seite 15)

3.6.1 Offset Tabelle

Über die Offset Tabelle können bestimmte Eigensschaften zum Tool definiert werden. So kann über den Offset beim Chip tool der Spitzentyp bestimmt werden.

Da Tech tool und X-Tool über den gleichen ID Widerstand erkannt werden, erfolgt die Unterscheidung über den Offset .

Die entsprechende Definition zum Offset ist der Bedienungsanleitung zu entnehmen.

3.6.2 Powerlevel

Über den Powerlevel wird das Nachheizverhalten und somit das Regelverhalten des Tools bestimmt. Wie die Tabelle zeigt, können aufgrund der Regelcharakteristik nur bestimmte Tools über den Parameter beeinflusst werden.

Tool	Low	Medium	High
ITool	1	2	3
Tech Tool	1	2	3
Power tool	1	2	3
X-Tool	-	-	X
Chip Tool	-	-	X
Micro tool	-	-	X

3.6.3 Standbyzeit

Die Standbyzeit kann beim i-Tool auch kleiner 1 Minute definiert werden. An der Station wird die Standbyzeit von 0, 20s, 30s, 40s, 50s und 1-60Minuten eingestellt. Alle anderen Tools haben einen Einstellbereich von 0-60 Minuten eingestellt.

Da nur ein Datenbyte für den Parameter vorhanden ist, wird über das höchste Bit (MSB bit7) angezeigt, ob der Zahlenwert (bit0..bit6) in Sekunden oder Minuten zu interpretieren ist.

Beispiel:	Zeitwert	Datenbyte
	0 (kein Standby)	0x00/0x80
	20s	0x94 (0x80 0x14)
	30s	0x9E
	40s	0xA8
	50s	0xB2
	1Min	0x01
	10Min	0x0A

Zeiten kleiner 20s sind für die Funktion des i-Tool nicht zulässig!

3.7 Systemparameter

Wie bei den Parametern, werden die Daten über den beschriebenen Datenbereich ausgetauscht. Werden vom Master Daten geändert, müssen die Änderungen über das Steuerregister 0x6FFF (Code 1) gespeichert werden.

Adresse	Daten	Format
0x6020	Optionen B0: Temperatureinheit 0→°C / 1→°F B1..B4: Sprachdialog iCon 0→ Deutsch / 1→ Englisch / 2→ Spanisch / 3→ Französisch / 4→ Russisch 5→ Italienisch / 6→ Portugiesisch B5..B6: nicht definiert (0) B7: Prozessalarm 0→Buzzer off / 1→ Buzzer on	Bitfeld
0x6021	Passwort	0-9999
0x6023	Temperaturfenster low (iCon / iCon2-linke Seite)	0-150K
0x6025	Temperaturfenster high (iCon / iCon2-linke Seite)	0-150K
0x6027	LCD Kontrast (10-50)	0-255
0x6028	Temperaturfenster low (iCon2-rechte Seite)	0-150K
0x602A	Temperaturfenster high (iCon2-rechte Seite)	0-150K

Soll die Temperatureinheit gewechselt werden, kann dies über ein spezielles Steuerregister erfolgen, damit alle Temperatureinstellungen entsprechend umgerechnet werden.

3.7.1 Wechsel der Temperatureinheit

Wird das Steuerregister 0x6FFF mit dem Code 9 beschrieben, erfolgt eine Umschaltung der Temperatureinheit (°C→°F→°C...). Dabei werden automatisch alle Temperaturparameter umgerechnet und abgespeichert.

Nach einer Umschaltung sollte die Lötstation ausreichend Zeit (ca. 3-4s) haben, alle Datensätze zu speichern.

3.8 Direkter Zugriff auf den EEPROM Speicher

Für bestimmte Anwendung kann es erforderlich sein, direkt Daten in den EEPROM Speicher zu schreiben. Dabei muss der Master beachten, dass der Slave keine Gültigkeitsprüfung beim Empfang der Daten macht. Datenbereiche (Parameter) werden über eine Checksumme überwacht, ändert der Master diese Datenbereiche, muss vom Master auch die Checksumme neu gebildet werden (CRC16).

3.8.1 Parameter

In der nachfolgenden Tabelle ist der Offset innerhalb eines Datenbereichs beschrieben. Zu jedem Tool gibt es Basisadresse, die zusammen mit dem Offset die Zieladresse ergibt.

Basisadresse:

Chip tool	0xD110
Micro tool	0xD120
Tech/X-Tool	0xD130
(reserviert)	0xD140
Power tool	0xD150
i-Tool	0xD160 (iCon/iCon2 linke Seite)
i-Tool	0xD210 (iCon2/rechte Seite)

Offset	Daten	Format
0x00	Solltemperatur	UINT
0x02	Kalibrierung	SCHAR
0x03	Offset	CHAR
0x04	Energie (Powerlevel)	CHAR
0x05	Standbyzeit	CHAR
0x06	Standbytemperatur	UINT
0x08	CRC	UINT

3.9 Sonstige Bereiche

3.9.1 Applikationskennung

Anhand der Kennung kann der Master ermitteln, mit welchem Applikationstyp eine Verbindung besteht. Mit der Versionskennung wird der Softwarestand des Slave eindeutig beschrieben.

Adresse	Daten	Format
0x0010	Applikationskennung ICon = 101 ICon2 = 102	UINT
0x0012	Versionskennung Version = 212 (Version 2.12)	UINT