

---

# HC08 Welcome Kit

Hardware-  
Version 2.01

## Benutzerhandbuch

26. Juni 2003

---

Copyright (C)2000-2003 by  
MCT Elektronikladen GbR  
Hohe Str. 9-13 D-04107 Leipzig  
Telefon: +49-(0)341-2118354  
Fax: +49-(0)341-2118355  
Email: mct@elektronikladen.de  
Web: <http://www.elektronikladen.de/mct>

Dieses Handbuch wurde sorgfältig erstellt und geprüft. Trotzdem können Fehler und Irrtümer nicht ausgeschlossen werden. MCT übernimmt keinerlei juristische Verantwortung für die uneingeschränkte Richtigkeit und Anwendbarkeit des Handbuchs und des beschriebenen Produktes. Die Eignung des Produktes für einen spezifischen Verwendungszweck wird nicht zugesichert. Die Haftung des Herstellers ist in jedem Fall auf den Kaufpreis des Produktes beschränkt. Eine Haftung für eventuelle Mangelfolgeschäden wird ausgeschlossen.

Produkt- und Preisänderungen bleiben, auch ohne vorherige Ankündigung, vorbehalten.

Die in diesem Handbuch erwähnten Software- und Hardwarebezeichnungen sind in den meisten Fällen auch eingetragene Warenzeichen und unterliegen als solche den gesetzlichen Bestimmungen. Es kann aus dem Fehlen einer besonderen Kennzeichnung nicht darauf geschlossen werden, daß die Bezeichnung ein freier Warenname ist. Gleiches gilt für Rechte aus Patenten und Gebrauchsmustern.

---

# Inhalt

1. Überblick	3
Technische Daten	3
Lieferumfang	4
2. Schnellstart	5
3. Bestückungsplan	7
4. Jumper und Lötbrücken	9
Jumper	9
Lötbrücken	9
5. Anschlüsse	11
6. Schaltungsbeschreibung	14
Schaltplan	14
Controllerkern und Input/Output	14
Takterzeugung	15
Monitor Mode Interface	15
User RS232 Port	17
Stromversorgung	17
7. Anwendungshinweise	18
Startup-Code	18
Initialisierung der PLL	18
Zusatzinformationen im Web	19
Zusätzliche HC08 Entwicklungssoftware	19
8. Flash Loader FL08	20
Überblick	20
Serielle Kommunikation	20
FL08 Kommandos	21
Autostart Funktion	21
Startup Verhalten	22

Redirected Interrupt Vectors .....	22
9. Memory Map .....	24
Anhang .....	25
S-Record Format .....	25
EMV Hinweise .....	27

# 1. Überblick

Das HC08 Welcome Kit erleichtert die Evaluierung des Motorola Mikrocontrollerbausteins MC68HC908GP32 und ist eine schnell verfügbare, kostengünstige Ausgangsbasis für die Realisierung von Applikationen mittlerer Komplexität.

Die kompakte Bauweise des Evaluation Boards ermöglicht den Einsatz der HC08 Welcome Kit Hardware auch für Kleinserienanwendungen.

Die für den HC08 erhältliche umfassende Softwareunterstützung (Debugging Tools, Assembler, C-Compiler) erleichtert die Entwicklung von Embedded Systemen.

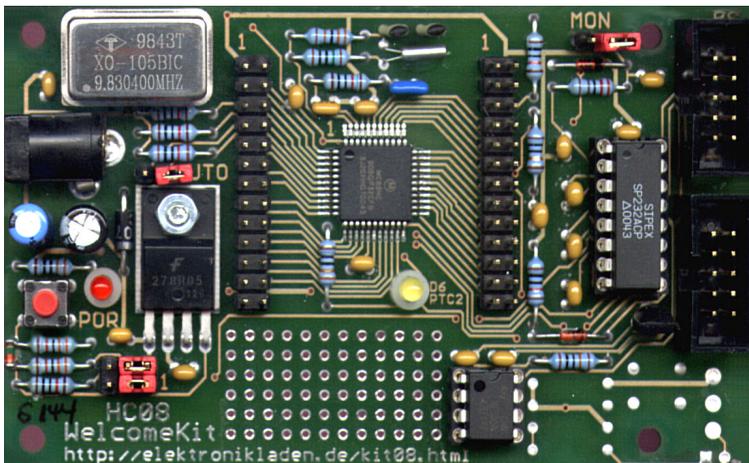
## Technische Daten

- Motorola 8 Bit Mikrocontroller MC68HC908GP32 im QFP44 Gehäuse mit bis zu 33 I/O-Pins
- Quarzoszillator 9,8304 MHz (gesockelt), Bustakt bis 8 MHz
- Alternativ: PLL-Takterzeugung mit 32 kHz Uhrenquarz
- ca. 32 KB Flash Speicher, 512 Byte RAM
- SPI - synchrone serielle Schnittstelle
- 4x 16-Bit Timer (Input Capture/Output Compare/PWM)
- 8-Kanal 8-Bit A/D-Wandler
- alle I/O-Anschlüsse sind auf seitliche Steckverbinder herausgeführt
- Monitor-Mode-Interface zur Programmierung im eingebauten Zustand (ISP)
- einfaches Debugging durch Remote Power-Cycle und Remote Break Feature
- Serielle Schnittstelle inkl. RS232-Treiber zum Anschluß an PC oder serielles LCD
- Indikator LED
- 5V On-Board Spannungsregler und Anschluß für Steckernetzteil

- Reset-Taste (Power-On Reset)
- Lochrasterfeld 6x12
- kompakte Abmessungen: 100 mm x 60 mm

### Lieferumfang

- Evaluation Board mit HC908GP32
- RS232 Anschlußkabel (Sub-D9)
- steckbarer Quarzosillator 9,8304 MHz
- Integrierter High-Speed Flash-Loader FL08
- Hardwarehandbuch, Schaltplan



*HC08 Welcome Kit V2*

## 2. Schnellstart

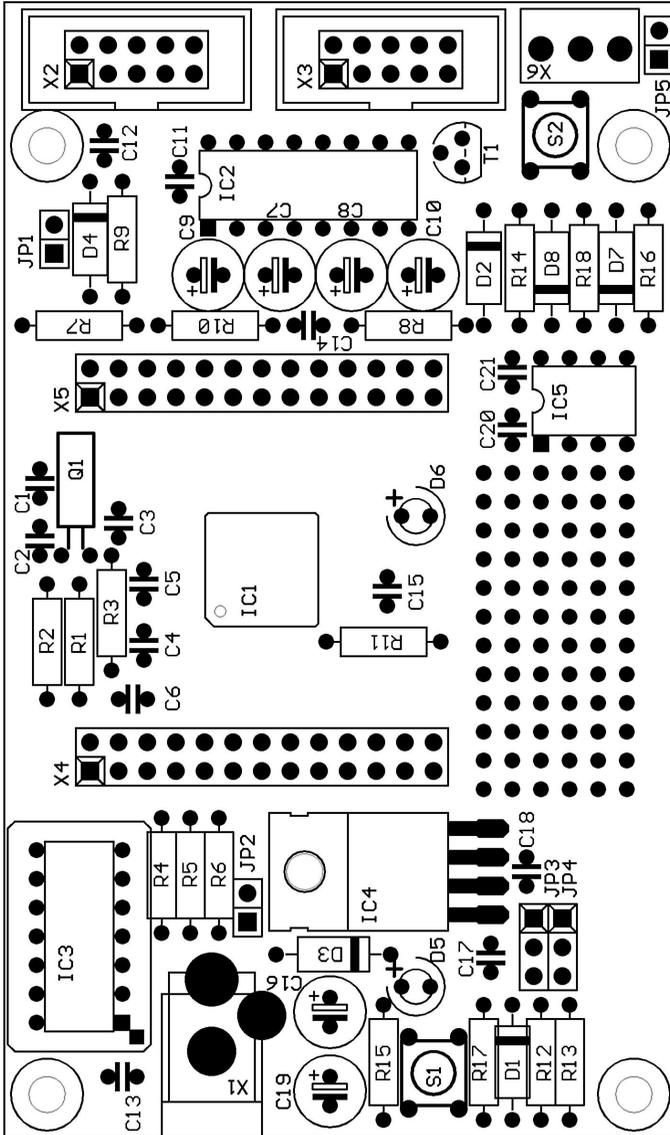
Kein Mensch liest gern dicke Handbücher. Daher hier die wichtigsten Hinweise in Kürze. Wenn Sie sich jedoch über ein Detail einmal nicht sicher sind, dann informieren Sie sich am Besten in den nachfolgenden Kapiteln. Und so können Sie beginnen:

- Kontrollieren Sie die Baugruppe zuerst auf eventuelle Transportschäden.
- Überprüfen Sie die Stellung der Jumper: JP1 (MON) und JP2 (AUTO) müssen offen sein, JP3 (VENA) und JP4 (RRST) in Stellung 1-2
- Überprüfen Sie die Lötbrücken: BR1 und BR2 sind "ab Werk" in Stellung 1-2, BR3 ist offen und BR4 sowie BR5 sind geschlossen
- Verbinden Sie das Controller Modul via RS232 mit Ihrem PC. Die Verbindung zwischen HC08 Welcome Kit (Anschluß X3) und PC erfolgt über das beiliegende 10-polige Flachbandkabel.
- Starten Sie auf dem PC ein Terminalprogramm. Ein einfaches Programm wie OC-Console (kostenlos auf unserer Website!) reicht aus.
- Stellen Sie die Übertragungsgeschwindigkeit auf 115200 Baud ein. Schalten Sie alle Übertragungsprotokolle (Hard- bzw. Softwarehandshake) aus. Aktivieren Sie die \*-Quittungsfunktion (das Terminal muß vor Senden der nächsten Zeile auf das Quittungszeichen "\*" warten)
- Schließen Sie über X1 eine Versorgungsspannung von ca. 9V= an (Plus am Innenkontakt, GND außen).
- Daraufhin startet der Flash Loader FL08, eine Software zum schnellen Download und Start von Programmen im Motorola S-Record- (S19-) Format.
- Löschen Sie ggf. zunächst den Flash-Speicher mit dem Kommando "X"
- Starten Sie den Download mit dem Kommando "L"
- Senden Sie die S-Record Datei (OC-Console: "Transfer")

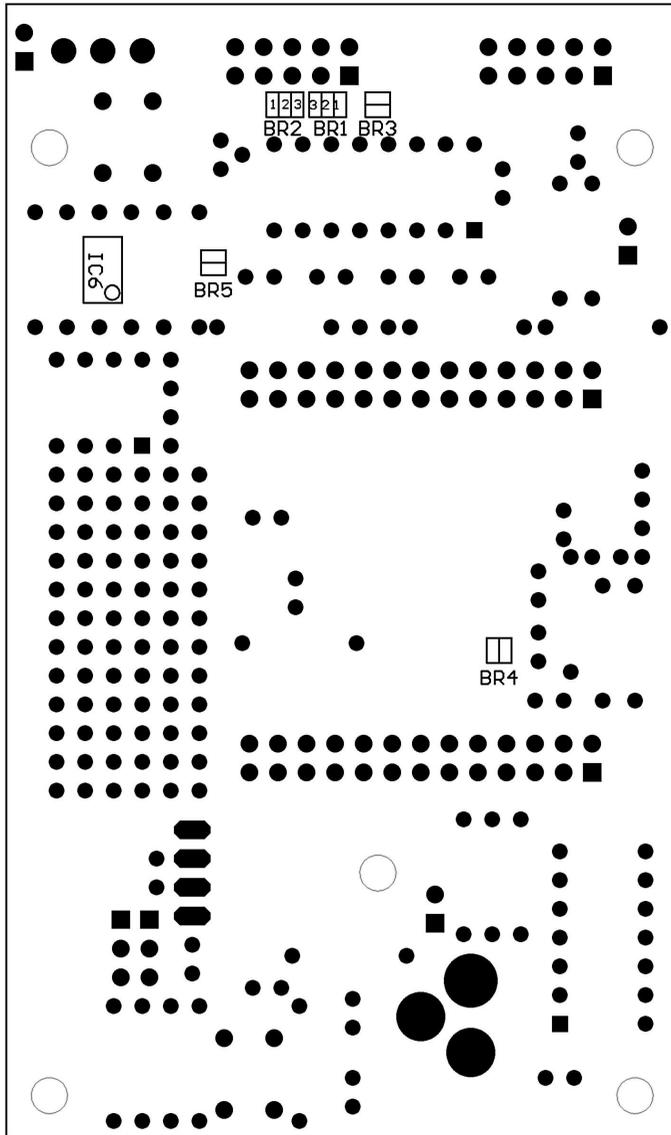
- Das Programm wird nun geladen, dieser Vorgang ist zu erkennen an der wachsenden Anzahl Quittungszeichen. Schließlich wird das geladene Programm gestartet (wenn es eine entsprechende S9-Zeile enthält).
- Ein Neustart des Programms (nach Reset) kann entweder mit dem Flash Loader Kommando "G" veranlasst werden, oder durch Stecken des AUTO-Jumpers JP2. In beiden Fällen muß das Programm an der Adresse \$8000 beginnen.

Wir wünschen Ihnen viel Erfolg mit dem HC08 Welcome Kit!

### 3. Bestückungsplan



Lageplan Bestückungsseite



*Lötbrücken auf der Platinenrückseite*

---

## 4. Jumper und Lötbrücken

### Jumper

Die Lage der Jumper kann dem Bestückungsplan (siehe oben) entnommen werden:

- JP1: MON (Monitor Mode)**
- offen\* User Mode (Anwenderprogramm bzw. FL08 Flash Loader)
  - geschl. Monitor Mode (Programmierung/Debugging via MON08-Schnittstelle)
- JP2: AUTO (Autostart)**
- offen\* Flash Loader FL08 startet normal
  - geschl. Flash Loader springt sofort zu Adresse \$8000 (Autostart Anwenderprogramm)
- JP3: VENA (Voltage Enable)**
- 1-2\* Standardeinstellung zur Steuerung IC4
  - 2-3 reserviert für LIN-Option
- JP4: RRSST (Remote Reset)**
- 1-2\* Remote Reset via /DTR Leitung der ser. Schnittstelle
  - 2-3 Remote Reset deaktiviert

### Lötbrücken

Die folgenden Lötbrücken befinden sich auf der Unterseite der Platine (vergl. Lageplan auf vorhergehender Seite):

- BR1, BR2: RS232 TxD/RxD Select**
- 1-2\* RS232 als "Device" konfiguriert (für Verbindung zu einem PC)
  - 2-3 RS232 als "Host" konfiguriert (für Verbindung zu serielltem LCD o.ä.)

**BR3: LCD Power Supply**

- offen\* VCC nicht am RS232 Steckverbinder verfügbar  
(normale Belegung)
- geschl. VCC am RS232 Port verfügbar  
(Pin9 des Sub-D Steckers)

**BR4: OSC**

- offen Takterzeugung durch Oszillatorbaustein IC3  
(erforderlich für Monitor Mode)
- geschl.\* Takterzeugung durch 32 kHz Quarz und PLL  
(gesockelten IC3 in diesem Fall entfernen!!)

**BR5: RXD**

- offen reserviert für LIN-Option - RxD der MCU offen
- geschl.\* RxD-Pin des Controllers ist mit dem RS232-  
Transceiver IC2 verbunden

\* = Standardeinstellung

## 5. Anschlüsse

### Power Supply Connector X1

- + Innenkontakt
- Außenkontakt

### Monitor Mode Connector X2

n.c.	<b>1</b>	<b>2</b>	n.c.
PC_RxD	<b>3</b>	<b>4</b>	n.c.
PC_TxD	<b>5</b>	<b>6</b>	n.c.
n.c.	<b>7</b>	<b>8</b>	n.c.
GND	<b>9</b>	<b>10</b>	n.c.

**X2**

### User RS232 Connector X3

n.c.	<b>1</b>	<b>2</b>	n.c.
Rx (Tx)	<b>3</b>	<b>4</b>	n.c.
Tx (Rx)	<b>5</b>	<b>6</b>	n.c.
n.c.	<b>7</b>	<b>8</b>	(VCC)
GND	<b>9</b>	<b>10</b>	n.c.

**X3**

## Expansion Connector X4

VCC	<b>1</b>	<b>2</b>	VCC
GND	<b>3</b>	<b>4</b>	/RST
PTC0	<b>5</b>	<b>6</b>	PTC1
PTC2	<b>7</b>	<b>8</b>	PTC3
PTC4	<b>9</b>	<b>10</b>	PTC5
PTC6	<b>11</b>	<b>12</b>	n.c.
TXD	<b>13</b>	<b>14</b>	RXD
GND	<b>15</b>	<b>16</b>	/IRQ
/SS	<b>17</b>	<b>18</b>	MISO
MOSI	<b>19</b>	<b>20</b>	SPSCK
T1CH0	<b>21</b>	<b>22</b>	T1CH1
T2CH0	<b>23</b>	<b>24</b>	T2CH1
GND	<b>25</b>	<b>26</b>	GND

**X4**

---

## Expansion Connector X5

VCC	<b>1</b>	<b>2</b>	VCC
GND	<b>3</b>	<b>4</b>	n.c.
n.c.	<b>5</b>	<b>6</b>	n.c.
PTA7	<b>7</b>	<b>8</b>	PTA6
PTA5	<b>9</b>	<b>10</b>	PTA4
PTA3	<b>11</b>	<b>12</b>	PTA2
PTA1	<b>13</b>	<b>14</b>	PTA0
GND	<b>15</b>	<b>16</b>	VDDAD
PTB7	<b>17</b>	<b>18</b>	PTB6
PTB5	<b>19</b>	<b>20</b>	PTB4
PTB3	<b>21</b>	<b>22</b>	PTB2
PTB1	<b>23</b>	<b>24</b>	PTB0
GND	<b>25</b>	<b>26</b>	GND

**X5**

## **6. Schaltungsbeschreibung**

### **Schaltplan**

Damit alle Details gut lesbar bleiben, liegt der Schaltplan im A4-Format separat bei.

### **Controllerkern und Input/Output**

Der Mikrocontroller HC908GP32 benötigt nur sehr wenige externe Bauelemente. Eine Vielzahl Peripheriefunktionen, wie SCI-Schnittstelle, Analog/Digital-Wandler und 32 KB Flashspeicher, sind bereits integriert. Der HC908GP32 wird in mehreren Gehäuseformen angeboten. In der Schaltung des HC08 Welcome Kit kommt die 44-polige QFP-Version zum Einsatz, welche im Vergleich zur DIP-Version vier zusätzliche I/O-Pins besitzt.

Als Betriebsspannung wurde für das HC08 Welcome Kit 5V gewählt. Der HC908GP32 kann alternativ auch bei 3,0V betrieben werden (siehe Datenblatt).

Das Resetsystem des HC08 weist deutliche Unterschiede auf im Vergleich zu anderen Motorola-Controllerfamilien (HC11, HC12). Der hier in der Schaltung vorgesehene C6 am Resetpin wäre z.B. bei einem HC11-System fehl am Platze. Statt dessen wäre für den HC11 ein Resetcontroller (LVI - Low Voltage Inhibit) dringend anzuraten, der hier entfallen kann, denn der HC908GP32 verfügt bereits über eine integrierte LVI-Schaltung.

Zur optischen Signalisierung ist am Port PTC2 eine Leuchtdiode angeschlossen. Dieser Portpin hat eine vergleichsweise hohe Treiberleistung (bis zu 25 mA), daher ist die Zwischenschaltung eines Verstärkers nicht erforderlich.

Die I/O-Pins des Controllers sind über die Steckverbinder X4 und X5 zugänglich. An diesen Steckverbindern lassen sich Peripherieschaltungen des Benutzers betreiben, unter Beachtung der Rahmenbedingungen z.B. hinsichtlich des Monitor Mode (s.u.).

## **Takterzeugung**

Der Quarzoszillatorbaustein IC3 liefert der MCU einen Takt von 9,8304 MHz. Dieser wird über den OSC1 Eingang des Controllers eingespeist, dabei muß die Lötbrücke BR4 offen sein. Dieser externe Takt wird benötigt, um die MCU im Monitor-Mode zu betreiben (Programmierung, Debugging). Auch bei der normalen Abarbeitung eines Anwenderprogramms (User-Mode) kann der von IC3 stammende externe Takt genutzt werden.

Alternativ kann die PLL-Schaltung des HC908GP32 als Taktquelle dienen. Hierzu bilden die Bauelemente Q1, C1, C2, R1 und R2 zusammen mit dem im Controller integrierten aktiven Teil einen Pierce-Oszillator. Dieser schwingt auf der Quarzfrequenz von 32768 Hz und es wird daraus über die PLL der interne Bustakt (bis zu 8 MHz) abgeleitet. Die Bauelemente R3, C3 und C4 bilden die PLL-Filterkombination. Für die PLL-Betriebsart muß der gesockelte Oszillatorbaustein IC3 entfernt und die Brücke BR4 geschlossen werden. Der bei Auslieferung des HC08 Welcome Kits in der MCU enthaltene Flash-Loader FL08 nutzt die PLL zur Erzeugung eines (Bus-) Taktes von 7,3728 MHz.

Zum Betrieb der PLL sind einige Initialisierungsschritte notwendig, die im Datenbuch des HC908GP32 ausführlich erläutert sind. Ein Initialisierungsbeispiel befindet sich im Abschnitt "Applikationshinweise" dieses Handbuchs.

## **Monitor Mode Interface**

Für die Zwecke der Flash-Programmierung und des Debugging verfügt jeder HC08 Controller über eine spezielle Betriebsart namens "Monitor Mode". Der Unterschied des Monitor Mode zum normalen User Mode besteht darin, daß statt des Anwenderprogramms eine im ROM abgelegte Firmware zur Ausführung gelangt. Diese Firmware prüft zunächst eine Reihe von I/O-Pins und legt die konkreten Betriebsparameter fest. Schließlich etabliert diese Firmware auf dem Port Pin PTA0 eine asynchrone serielle Schnittstellenfunktion. Diese arbeitet bidirektional (halbduplex) und entspricht den üblichen RS232-Festlegungen. Die Baudrate beträgt 9600 Baud. Voraussetzung ist, abgesehen

vom Quarztakt (9,8304 MHz), die Belegung einiger Portpins mit Logikpegeln, wie sie folgende Tabelle auflistet:

<b>Port</b>	<b>Pegel</b>
PTA0	H
PTA7	L
PTC0	H
PTC1	L
PTC3	H

*Monitor Mode - Pegel an den Ports A und C*

Die Monitor Mode Interfaceschaltung auf dem Evaluation Board erzeugt diese Pegel über fünf Pull-Up bzw. Pull-Down Widerstände.

Neben den genannten Voraussetzung für den Monitor Mode ist es weiterhin notwendig, am /IRQ-Pin des Controllers eine Spannung im Bereich von ca. 7 bis 10 Volt anzulegen. Diese wird in der Schaltung von der Ladungspumpe des RS232 Transceivers IC2 abgezweigt und über die Zenerdiode D4 auf 8,2V begrenzt. Jumper JP1 (MON) muß gesetzt werden, um diese erhöhte Spannung an /IRQ anzulegen.

Die Verbindung des Monitor Mode Interface zum PC erfolgt über den Steckverbinder X2. Daran angeschlossen ist ein Flachbandkabel mit Pfostenverbinder (2x5 Pins, Crimpverbindung) an der Deviceseite und einer Sub-D9 Buchse (ebenfalls gecrimpt) an der PC-Seite. Durch diese Kabelkonfiguration wird eine 1:1 Verbindung realisiert, wie die folgende Tabelle zeigt:

<b>X2 Pin</b>	<b>USB08 MONI</b>	<b>PC RS232</b>	<b>Sub-D9 Pin</b>
3	T1OUT	RxD	2
5	R1IN	TxD	3
9	GND	GND	5

*Steckerbelegung Monitor Mode Kabel*

## **User RS232 Port**

Im Gegensatz zum RS232-Kanal für das Monitor Mode Interface, weist der User RS232 Port getrennte Sende- und Empfangsleitungen auf. Der HC908GP32 verfügt über ein SCI-Hardwaremodul für die asynchrone serielle Kommunikation mit den Signalbezeichnungen TXD (Senden) und RXD (Empfangen).

X3 ist der Steckverbinder für den User RS232 Port. Beim Anschluß dieser Schnittstelle an einen PC ist eine Leitungsverbindung analog zum Monitor Mode Interface erforderlich (s.o.). Die Brücken BR1 und BR2 auf der Platinenunterseite sind dann in Stellung 1-2 zu verbinden (Default). Der PC fungiert als Host, das HC08 Welcome Kit bildet die Deviceseite.

Der umgekehrte Fall tritt z.B. ein, wenn ein serielles LC-Display am User RS232 Port betrieben werden soll. Hier ist das HC08 Welcome Kit der Host und das LCD-Modul tritt an die Deviceseite. Die hierzu erforderliche RxD/TxD-Leitungskreuzung wird realisiert durch Konfiguration der Brücken BR1/BR2 in Stellung 2-3. Gleichzeitig kann durch Schließen der Brücke BR3 das serielle LCD mit Betriebsspannung versorgt werden (Achtung: diese Belegung weicht von der RS232 Standardbelegung ab!).

Serielle, alphanumerische LC-Displays gibt es von einer Vielzahl Anbieter, verbreitet sind z.B. die Module des kanadischen Herstellers Matrix Orbital (<http://www.matrix-orbital.com>).

## **Stromversorgung**

Die Stromversorgung erfolgt durch den auf der Platine befindlichen Längsregler IC4. An der Buchse X1 wird eine Gleichspannung von 8 bis 12 Volt eingespeist.

IC4 ist ein mit 1 Ampere klassifizierter Spannungsregler-IC. Normalerweise ist er - auch ohne spezielle Kühlmaßnahmen - mehr als ausreichend dimensioniert. Die Stromaufnahme des Boards ist, selbst im worst-case Fall, deutlich geringer als 100mA.

## 7. Anwendungshinweise

In diesem Abschnitt sollen einige Besonderheiten der Programmierung des HC908GP32 genannt werden.

Dieses Hardwarehandbuch kann nur einige *spezifische* Hinweise geben. Die Behandlung *allgemeiner* Techniken zur Programmierung des Controllers in Assembler bzw. Hochsprachen würden Umfang und Ziel dieses Handbuchs sprengen. Die meisten Antworten finden Sie beim (leider unerläßlichen) Studium der Datenblätter und Referenzhandbüchern der Halbleiterhersteller.

### Startup-Code

Jede Controllerfirmware beginnt mit einer Reihe von Anweisungen zur Initialisierung der Hardware. Im Fall des HC908GP32 beschränken sich die *unbedingt notwendigen* Initialisierungen auf wenige Aspekte, und zwar:

- Setzen des Stackpointers
- Einstellen des Basistaktes für das SCI (falls genutzt)
- Abschaltung (bzw. ggf. die geeignete Initialisierung) des Watchdog

Hierzu ein einfaches Assemblerbeispiel:

```
entry   ldhx  #$00ff           ; Program Entry Point
        txs                    ; Init Stack Pointer
        mov  #$01,CONFIG2     ; SCI uses Bus Clock
        mov  #$01,CONFIG1     ; Disable Watchdog
```

### Initialisierung der PLL

Falls ein Anwenderprogramm die Takterzeugung über die PLL abwickeln soll, muß das PLL Modul entsprechend der gewünschten Busfrequenz initialisiert werden.

Das folgende Programmlisting zeigt ein Initialisierungsbeispiel für einen Bus Clock von 7,3728 MHz (wie von FL08 verwendet):

PCTL	EQU	\$36
PBWC	EQU	\$37
PMSH	EQU	\$38
PMSL	EQU	\$39
PMRS	EQU	\$3A
PMDS	EQU	\$3B

BCS	EQU	4
PLLON	EQU	5
LOCK	EQU	6
AUTO	EQU	7

```

INITPLL BCLR    BCS,PCTL    ;Select external Clock
        BCLR    PLLON,PCTL  ;Turn PLL off
        MOV     #$02,PCTL    ;P=0, E=2
        MOV     #$01,PMDS    ;R=1
        MOV     #$03,PMSH    ;N=900
        MOV     #$84,PMSL
        MOV     #$C0,PMRS    ;L=192
        LDA     PCTL         ;clear PLLF
        BSET   AUTO,PBWC    ;Automatic bandwidth control
        BSET   PLLON,PCTL   ;Turn PLL on
W4LOCK  BRCLR   LOCK,PBWC,W4LOCK ;Wait for lock
        BSET   BCS,PCTL     ;Select PLL Clock

```

## Zusatzinformationen im Web

Sobald zusätzliche Informationen zu Hard- und Software des HC08 Welcome Kit vorliegen, veröffentlichen wir diese auf unserer Website:

<http://www.elektronikladen.de/kit08.html>

## Zusätzliche HC08 Entwicklungs-Software

Motorola stellt für die HC08 Controller eine Reihe kostenloser Entwicklungstools zur Verfügung (Assembler, Programmer, Debugger etc.). Informationen hierzu:

<http://www.motorola.com/mcu/>

<http://www.pemicro.com/ics08/>

## 8. Flash Loader FL08

### Überblick

Der Flash Loader FL08 wurde implementiert, um das Laden von Anwenderprogrammen im S19-Format zu vereinfachen. Zwar verfügt der HC908GP32 über den Monitormode als spezielle Betriebsart für Download und Debugging, dieser ist aber an eine Reihe von Voraussetzungen geknüpft, wie z.B. der Einsatz eines externen Quarzoszillators.

Durch die Verwendung des FL08 ergeben sich folgende Vorteile:

- kein Betriebsartenwechsel zwischen Monitor- und Usermode erforderlich; die Umstellung von Jumpers und der Takterzeugung entfällt
- Nutzung der selben RS232 Schnittstelle für Download und Betrieb der Anwendung
- hohe Downloadgeschwindigkeit (115200 Baud)
- durch Setzen des AUTO Jumpers kann das geladene Anwenderprogramm nach Rest automatisch gestartet werden
- einfache Systeminitialisierung
- 7,3728 MHz Bustakt (wird mittels PLL aus einem 32 kHz Uhrenquarz generiert)
- alle Interruptvektoren stehen (umgeleitet in den internen RAM) weiterhin zur Verfügung

Nachteile ergeben sich durch den Ressourcenverbrauch (FL08 belegt einen Teil des Speichers der MCU) und die Limitierung der Debuggingmöglichkeiten (nur im Monitormode möglich).

### Serielle Kommunikation

FL08 kommuniziert über die RS232 Schnittstelle (SCI an X3) mit **115200 Baud**. Weitere Einstellungen: 8N1, kein Hardware- oder Softwarehandshake, kein Protokoll.

## **FL08 Kommandos**

Nach dem Start meldet sich FL08 mit einer Ausschrift über die serielle Schnittstelle und erwartet vom Benutzer eines der folgenden Kommandos:

### **X - Mass Erase**

Löscht den gesamten Flashspeicher (der geschützte FL08 Code bleibt dabei natürlich erhalten).

### **G - Go**

Startet das Anwenderprogramm durch Sprung auf Adresse \$8000.

### **L - Load**

Lädt eine Datei im Motorola S-Record Format (vergl. Anhang) in den Flash Speicher des Mikrocontrollers. Es werden S1-Records mit maximal 32 Datenbytes verarbeitet.

Wird während des Ladevorgangs auf eine bestimmte Flash Page zum erstenmal zugegriffen, wird diese Flash Page (128 Byte) zunächst gelöscht. Daher ist es nicht unbedingt erforderlich, vor dem Ladevorgang ein Mass Erase Kommando auszuführen.

Das sendende Terminal muß nach jeder übertragenen S-Record Zeile auf die Empfangsbestätigung (\*) warten!

Wird der Ladevorgang durch ein S9-Record beendet, welches eine Adresse ungleich Null enthält, wird das geladenen Programm sofort an dieser Adresse gestartet. Ist die Adresse nicht spezifiziert, kann das Programm manuell mit dem G Kommando (oder nach Reset automatisch durch die Auto-Start Funktion) aufgerufen werden. Es muss in diesem Fall stets auf Adresse \$8000 beginnen.

## **Autostart Funktion**

Der FL08 Flash Loader überprüft nach Reset, ob durch den AUTO Jumper (JP2) die Portpins PTC2 und PTC3 miteinander verbunden sind. Ist das der Fall, springt FL08 zur Adresse \$8000. Durch dieses Feature

wird es möglich, ein Anwenderprogramm automatisch zu starten, ohne den Resetvektor im geschützten Flash Boot Block ändern zu müssen.

## Startup Verhalten

Nach Reset werden durch FL08 einige Initialisierungsschritte ausgeführt, welche auch Auswirkungen auf ein Anwenderprogramm haben:

- CONFIG1 = \$01 (Watchdog disabled)
- CONFIG2 = \$01 (SCI verwendet Bustakt)
- PLL erzeugt 7,3728 MHz Bustakt
- SCI aktiv mit 115200 Baud (Polling Mode)
- Stack ab \$0200 abwärts

## Redirected Interrupt Vectors

Die Interruptvektoren des HC08 liegen am Ende des 64 KB umfassenden Adreßraumes, d.h. innerhalb des schreibgeschützten Flash Loader Codes. Um dennoch Interruptfunktionen in einem Anwenderprogramm zu ermöglichen, leitet der Flash Loader alle Interruptvektoren (außer den Resetvektor) auf Adressen im internen RAM um. Das Verfahren entspricht der Vorgehensweise des HC11 im Special Bootstrap Mode.

Das Anwenderprogramm setzt den benötigten Interruptvektor zur Laufzeit (vor der globalen Interruptfreigabe!), indem es einen Sprungbefehl in den RAM-Pseudovektor einträgt. Um z.B. den IRQ Interrupt nutzen zu können, muß ein Anwenderprogramm folgende Schritte ausführen:

```
lda  #$$CC           ; JMP Opcode
sta  $023A           ; IRQ Pseudo Vector
ldhx #isrFunc       ; Jump Address
sthx $023B          ; IRQ Pseudo Vector + 1
```

Für C-Programme läßt sich eine Codesequenz nach folgendem Muster verwenden:

```
// install IRQ pseudo vector in RAM
// (if running with FL08)
*((unsigned char *)0x023a) = 0xCC;    // JMP opcode
*((void (**)(void))0x023b) = isrFunc;
```

Die folgende Tabelle enthält alle Vektoradressen (originale und umgeleitete) für den HC908GP32:

HC08 Vector Address	Redirected Address	Interrupt Vector
\$FFDC	\$020D	Timebase
\$FFDE	\$0210	ADC
\$FFE0	\$0213	Keyboard
\$FFE2	\$0216	SCI Tx
\$FFE4	\$0219	SCI Rx
\$FFE6	\$021C	SCI Error
\$FFE8	\$021F	SPI Tx
\$FFEA	\$0222	SPI Rx
\$FFEC	\$0225	TIM2 Overflow
\$FFEE	\$0228	TIM2 Channel 1
\$FFF0	\$022B	TIM2 Channel 0
\$FFF2	\$022E	TIM1 Overflow
\$FFF4	\$0231	TIM1 Channel 1
\$FFF6	\$0234	TIM1 Channel 0
\$FFF8	\$0237	PLL
\$FFFA	\$023A	/IRQ
\$FFFC	\$023D	SWI
\$FFFE	(\$F000)	Reset

## 9. Memory Map

Von	Bis	Größe	Belegung
\$0000	\$003F	64 Byte	Steuerregister
\$0040	\$023F	512 Byte	RAM (FL08 Redirected Interrupt Vectors ab \$020D)
\$0240	\$7FFF		reserviert
\$8000	\$EFFF	28672 Byte	Flash Memory, 28 KB frei für Anwenderprogramm
\$F000	\$FDFF	3584 Byte	Flash Memory, 3,5 KB belegt durch Flash Loader FL08
\$FE00	\$FFDB		Steuerregister und Monitor ROM Code
\$FFDC	\$FFFF	36 Byte	Interrupt Vector Table (Flash)

*Memory Map des HC908GP32 (mit FL08)*

Details zur Memory Map des Controllers, insbesondere die Adressen von Steuerregistern und Interruptvektoren sind im Motorola Datenbuch **MC68HC908GP32 Technical Data** enthalten

# Anhang

## S-Record Format

Das von Motorola publizierte S-Record Format ist ein Dateiformat zur Definition von Objektdateien (Maschinencode, Executables) unter Verwendung einer textuellen (ASCII-) Notation, die es erlaubt, diese Objektdateien mit jedem beliebigen Texteditor zu betrachten oder zu ändern. Eine S-Record Datei besteht aus einer beliebigen Anzahl S-Records bzw. Zeilen. Eine jede Zeile hat die folgende logische Struktur:

ID	LEN	ADDR	DATA	CS	<EOL>
----	-----	------	------	----	-------

Das Feld ID gibt den S-Record Typ an. Relevant sind die Typen "S1", "S9" und gelegentlich "S0" (Kommentarrecord). Außer dem ID Feld bestehen alle weiteren Felder aus Paaren von Hexziffern, beispielsweise "A9", "55" oder "0F".

Das Feld LEN besteht aus einem derartigen Paar und bestimmt die Anzahl der folgenden Ziffernpaare (enthält die Ziffernpaare der Felder ADDR, DATA und CS).

ADDR ist die Anfangsadresse der Datenbytes dieser Zeile. Das Feld besteht aus zwei Byte (erst H-, dann L-Byte), d.h. aus zwei Ziffernpaaren.

DATA enthält die eigentlichen Codebytes, die das Maschinenprogramm bilden. DATA umfaßt (LEN - 3) Bytes bzw. Zeichenpaare.

Im Feld CS ist eine Prüfsumme enthalten. Sie wird gebildet aus den Werten der Zeichenpaare der Felder LEN, ADDR und DATA. CS ist das (niederwertigste Byte des) Einerkomplement der Summe aller vorgenannten Werte. EOL schließlich steht symbolisch für den durch CR, LF (\$0D, \$0A) gebildeten Zeilenvorschub.

Ein Beispiel soll die Handhabung verdeutlichen:

S1	13	2000	13A400262741010167CC10FF05C7A501	D1	<EOL>
----	----	------	----------------------------------	----	-------

Dieser S1-Record definiert \$13-3 = \$10 Bytes ab Adresse \$2000 des Zielsystems. Die Ziffernpaare des DATA Feldes ergeben eine Summe von \$04FB. Addiert man die \$13 aus dem LEN Feld sowie \$20 und \$00 aus dem ADDR Feld hinzu, ergibt sich ein Wert von \$052E. Das Einerkomplement des LSB (\$2E) ergibt \$D1. Dies ist der korrekte Wert für das Prüfsummenfeld.

Neben den S1-Records, welche die eigentlichen Daten enthalten, wird auch der S9-Typ verwendet. Dieser Typ beendet eine Serie von S1-Records. Abgesehen von dieser Terminierungs-Funktion kann in einem S9-Record die Startadresse des Programms vermerkt werden. Der Aufbau des S9-Records entspricht dem S1 Typ, wobei jedoch das Feld DAT leer bleibt. Das Feld ADDR spezifiziert die Startadresse des Programms. Ist hier \$0000 eingetragen, wird angenommen, daß die Adresse des ersten geladenen Codebyte gleichzeitig die Startadresse des Programms ist. Ein typischer S9-Record sieht wie folgt aus:

S9	3	B600		46	<EOL>
----	---	------	--	----	-------

## **EMV Hinweise**

Die Baugruppe entspricht den EMV-Vorschriften. Zur Stromversorgung ist sie an einer Batteriespannungsquelle mit 5,0 Volt (Einhaltung der Spannungsgrenzwerte beachten!) oder an ein Netzteil mit CE-Kennzeichnung anzuschließen. Der Einsatz einer Mikrocontrollerplatine geht stets einher mit einer mehr oder minder umfangreichen Modifikation der Baugruppe (spezielle Firmware, angeschlossene Peripheriebauteile). Der Hersteller kann den vom Kunden geplanten Einsatz der Baugruppe nicht vorhersehen und daher auch keine Vorhersagen über die EMV-Eigenschaften der modifizierten Baugruppe machen. Anwender ohne Zugriff auf ein EMV-Prüflabor sollten die folgenden Richtlinien beachten, die in der Regel eine einwandfreie Funktion der modifizierten Baugruppe gewährleisten:

Um sicherzustellen, daß die Baugruppe auch dann den EMV-Vorschriften entspricht, wenn Verbindungsleitungen zu anderen Geräten (z.B. Personalcomputer) angeschlossen werden oder die Baugruppe vom Kunden selbst mit weiteren Bauteilen nachgerüstet wird (z.B. Meßadapter oder Leistungsendstufen), empfehlen wir, die komplette Baugruppe in ein allseitig geschlossenes Metallgehäuse einzusetzen.

Wird ein LC-Display angeschlossen (ebenfalls auf CE-Kennzeichnung achten), so darf das Verbindungskabel nicht länger als 10 cm sein; hier ist auf jeden Fall ein Metallgehäuse vorzusehen. Wenn für die Programmentwicklung oder die spätere Anwendung die RS232 Schnittstelle benötigt wird, so ist ein max. 10cm langes Kabel zur Verbindung mit der Anschlußbuchse zu verwenden. Die geschirmte Anschlußbuchse ist fest mit dem Metallgehäuse zu verschrauben. Extern zur Verbindung verwendete Anschlußkabel müssen, ebenso wie der Hostrechner (PC), mit dem CE-Zertifizierungszeichen versehen sein.

Es wird darauf hingewiesen, daß der Anwender selbst dafür verantwortlich ist, daß eine veränderte, erweiterte, mit anderen als vom Hersteller gelieferten IC's bestückte oder mit Anschlußkabeln versehene Baugruppe den EMV-Vorschriften entspricht.





