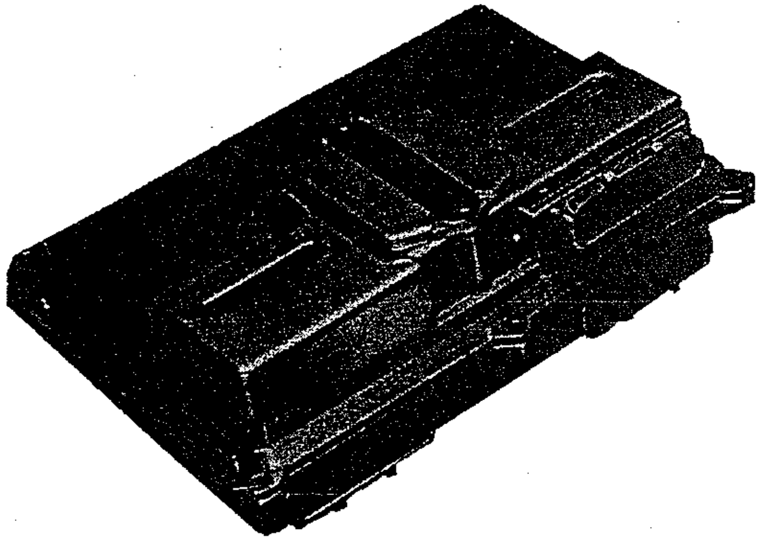
CC200 USER/APPLICATION MANUAL



MAGNE CHARGE™

Controller

Model CC200

Electric Vehicle
Engineering Support Services

Hughes Technical Services Company
3050 W. Lomita Blvd.
Torrance, CA. 90505
(310)517-5769

REV -

FEATURES

- MC68HC11 CPU
- Memory
  - 15616 Bytes SRAM
  - 640 Bytes EEPROM
  - 131,072 Bytes FLASH
  - 24,576 Bytes OTPROM
  - 16,384 Bytes SERIAL EEPROM
- Communications
  - SAE CLASS 2 : J1850
  - Distributed Energy Management Interface (SCI RS232 TTL-level)
- Two Hydrogen Gas Sensor Inputs
- 1 Frequency Measurement Input (PWM)
- 2 Pulse-Width Modulated Outputs
- 7 Isolated Discrete Inputs
- 9 Temperature Inputs
- 1 Pre-Charge Input
- 12-bit DAC Output (-5 to +5VDC)
- 8-bit Auxiliary Battery Sense
- Isolated 12-bit Pack Voltage Sense
- Isolated 12-bit Pack Current Sense
- 2 High-Side Switches (+12VDC)
- 7 Low-Side Switches (12VRTN)
- 1 SPDT Relay
- 1 DPDT Relay

APPLICATIONS

- Electric Vehicle Battery Charger Controller
- Electric Vehicle Energy Management System

DESCRIPTION

The CC200 is a 68HC11 microprocessor based electric vehicle battery charger controller and energy management system. It uses an embedded systems architecture to house battery charging and energy management algorithms in user configurable memory. During the charging process the CC200 continuously monitors the state of the batteries and controls the charger output level via the SAE Class 2 Communication Interface (Protocol J1850). The CC200 signals the charger to stop charging when it determines that the battery pack is completely charged, or if a fault is detected during the charging process. It has the capability of reading traction battery pack voltage with 12-bit resolution, reading charge/discharge current using fine (0-±100A) or coarse (0-±2500A) measurement channels, and monitoring, with a resolution of 8-bits, the auxiliary battery voltage. Additionally, the controller has the capability to control and monitor various vehicle subsystems, such as: safety interlocks, battery pack isolation(referenced to chassis), hydrogen detector inputs, environmental controls, state of charge indicators, fans, and pump controls.

The development software package, used in conjunction with the users laptop computer, provides the ability to customize the controller to the users needs, monitor controller diagnostics, and monitor charging status. Customizing is accomplished by an editing feature that allows various charging parameters such as: charge modes, temperature compensation values, voltage/current trip and set points, battery pack size, battery characteristics, and power-up delays to be modified and reprogrammed into the controller. Monitoring screens provide the user with real time information on the charging status such as: charge mode, intial SOC, amphours put back into battery pack, battery pack voltage levels, battery pack current levels(based on user supplied 1mohm shunt resistor), auxiliary battery level, and battery pack temperature. Diagnostic screens indicate the condition and status of the numerous fault conditions that can exist during a charging sequence such as: thermistor shorts and opens, battery pack overtemp, battery pack over or under voltage levels, pack voltage detection on vehicle chassis(isolation), charge port overtemp, conversion box overtemp, and communication conditions between controller and charger.

Initial programming of the CC200 is accomplished via the controllers serial communications interface (SCI), and depending on the controllers configuration, communication between the user's laptop and controller is either via the controller's SCI or the J1850 interface.

The CC200 also has the capability to communicate with Distributed Energy Managment Modules. These modules can be attached to individual batteries to monitor battery voltage and temperature and then provide this information to the charge controller.

Planned software upgrades will provide the following capabilities:
*Amp/hour tracking during drive mode
*Data logging/collection
*SOC gauge(peukert, temperature, aging, self-discharge, etc)
*Torque limits due to SOC, temperature, etc
*Regen limits due to SOC, temperature, etc
*Storing battery module historical data
*Range predictions
*Graphing data
*Battery chemistry files
*Non-venting/vent control
*Battery environmental controls
*Hydrogen sensing

ABSOLUTE MAXIMUM RATINGS[1]

Supply Voltage
(ΔVAUX)....................................16V

Pack Voltage Sense
(ΔVSENSE)...............................500V

Pack Current Sense[2]
(ΔISENSE)[3]..............................±100A
(ΔISENSE)[4]...........................±2500A

Discrete Input Voltage
(ΔV_{DISC})[5]......................................20V

Low-Side Switches[6]
Input Voltage (V_{LSS})........ (ΔVAUX)
Input Current (V_{LSS})[7]................0.5A

Relay Contact & Wiper Current
RELAY1 (I_{REL1})..............................5A
RELAY2 (I_{REL2})..............................5A

Frequency/Pulse Width Input[8]
Input High Voltage
(V_{PFW_IH})............................5.45V
Input Low Voltage
(V_{PFW_IL}).............................1.03V

Pre-Charge Input Voltage
(ΔV_{A8_SIG})............................(ΔVAUX)

Temperature Range
Operating (T_{OPR})........-40°C to 85°C
Storage (T_{STR})...........-55°C to 125°C

1. Unless other wise specified, absolute maximum ratings apply over operating temperature range.
2. Pack current sensed using a 1mΩ shunt resistor.
3. ΔISENSE measured using fine channel #0.
4. ΔISENSE measured using coarse channel #1.
5. Applicable for discrete signals 0 to 6.
6. Applicable for low-side switches 0 to 6. Voltage is referenced to VAux Rtn.
7. Per input continuous rating with all inputs on and equally conducting. With one input on, maximum current rating is 1.5A.
8. Voltage is referenced to VAux Rtn.

ELECTRICAL CHARACTERISTICS

$8V \leq \Delta VAUX \leq 16V$, $-40^{o}C \leq T_{OPR} \leq 85^{o}C$, $V_{ISO} \leq 500V$ (Unless otherwise specified)

SYMBOL	PARAMETER	CONDITIONS	CC200 MIN	CC200 TYP	CC200 MAX	UNITS
ΔVAUX	Input Power		8	12	16	VDC
I_{AUX}	Input Current	normal mode		300		mA
I_{AUX}	Input Current	sleep mode		55	60	mA
ΔVSENSE	Pack Voltage Sense		0		500	VDC
A_{VSENSE}	VSENSE accuracy			122.803		mV/lsb
f_{VSENSE}	spi clock frequency		Note 1		200	KHz
ΔISENSE	Pack Current Sense	1mΩ shunt, fine sense, continuous	-100		+100	A
ΔISENSE	Pack Current Sense	1mΩ shunt, coarse sense, non-continuous	-2500		+2500	A
A_{ISENSE}	ISENSE accuracy	1mΩ shunt, fine		47.498		mA/lsb
A_{ISENSE}	ISENSE accuracy	1mΩ shunt, coarse		1.221		A/lsb
f_{ISENSE}	spi clock frequency		Note 2	250	320	KHz
ΔV_{DISC}	Discrete Input Voltage	applies to channels 0...6	0	12	20	VDC
I_{REL1}	Relay1 Contact Current	DPDT Relay1			5	A
I_{REL0}	Relay0 Contact Current	SPDT Relay0			10	A
V_{FPW_IH}	freq/pw input high voltage	Ref. to Vaux Rtn.	3.605		5.450	VDC
V_{FPW_IL}	freq/pw input low voltage	Ref. to Vaux Rtn	-0.3		1.03	VDC
V_{HSS}	output voltage high-side switch		8	12	16	VDC
I_{HSS}	output current high-side switch				1.2	A
V_{DAC}	DAC output voltage	$10.8V \leq \Delta VAUX \leq 16V$	-5.0		4.9976	VDC
f_{PWM}	PWM output frequency	PWM enabled	3.83		4e6	Hz
V_{ISO}	Isolation Voltage = VSENSE- - VAUX-		0		500	VDC
f_{SYS_CLK}	CPU Bus Speed			4		MHz

TABLE 1.0

Note 1 : Increased leakage currents at elevated temperatures cause the VSENSE a/d to droop, therefore it is recommended that $f_{clk} \geq$ 120KHz at 85C, $f_{clk} \geq$ 75KHz at 70C, and $f_{clk} \geq$ 1KHz at 25C

Note 2 : Increased leakage currents at elevated temperatures cause the ISENSE a/d to droop, therefore it is recommended that f_{clk}= 200KHz at 85C, $f_{clk} \geq$ 120KHz at 70C, and $f_{clk} \geq$ 1KHz at 25C

PIN FUNCTIONS

NUMBER	NAME	FUNCTION	DESCRIPTION
J1-1	GAS_SENSE0_VC	Gas sensor Vcc	Reference input to H_2 gas sensor #0
J1-2	GAS_SENSE0_SIG	Gas sensor signal	H_2 gas sensor#0 signal output
J1-3	GAS_SENSE0_RTN	signal rtn	H_2 gas sensor#0 signal output
J1-4	GAS_SENSE1_VC	Gas sensor Vcc	Reference input to H_2 gas sensor #1
J1-5	GAS_SENSE1_SIG	Gas sensor signal	H_2 gas sensor#1 signal output
J1-6	GAS_SENSE1_RTN	signal rtn	H_2 gas sensor#1 signal output
J1-7	A0_SIG	analog input #0	thermistor #0 signal
J1-8	A0_RTN	analog input return	thermistor signal return
J1-9	A1_SIG	analog input #1	thermistor #1 signal
J1-10	A1_RTN	analog input return	thermistor signal return
J1-11	A2_SIG	analog input #2	thermistor #2 signal
J1-12	A2_RTN	analog input return	thermistor signal return
J1-13	A3_SIG	analog input #3	thermistor #3 signal
J1-14	A3_RTN	analog input return	thermistor signal return
J1-15	A4_SIG	analog input #4	thermistor #4 signal
J1-16	A4_RTN	analog input return	thermistor signal return
J1-17	A5_SIG	analog input #5	thermistor #5 signal
J1-18	A5_RTN	analog input return	thermistor signal return
J1-19	A6_SIG	analog input #6	thermistor #6 signal
J1-20	A6_RTN	analog input return	thermistor signal return
J1-21	A7_SIG	analog input #7	thermistor #7 signal
J1-22	A7_RTN	analog input return	thermistor signal return
J1-23	A8_SIG	analog input #8	thermistor #8 signal
J1-24	A8_RTN	analog input return	thermistor signal return
J1-25	A9_SIG	analog input #9	thermistor #9 signal
J1-26	A9_RTN	analog input return	thermistor signal return
J1-27	NOT USED		
J1-28	DGND	digital ground	
J1-29	FREQ_PW_0	digital input	frequency/pulse width measurement input
J1-30	FREQ_PW_0_RTN	digital ground	frequency/pulse width measurement input return
J1-31	GAS_SENSE1_VH	switched +5V_DIG	H_2 gas sensor#1 heater power input
J1-32	GAS_SENSE0_VH	switched +5V_DIG	H_2 gas sensor#0 heater power input

TABLE 2.0

PIN FUNCTIONS

NUMBER	NAME	FUNCTION	DESCRIPTION
J2-1	J1850	J1850 output signal	
J2-2	TLM_BIAS_RXD	opto-isolator Bias	Bias voltage input used for isolated Telemetry RxD
J2-3	TLM_SIG_RXD	opto-isolator signal	Telemetry RxD signal
J2-4	TLM_SIG_BTXD	buffered transmit output	Telemetry/RS232 TxD signal
J2-5	TLM_RTN_BTXD	digital ground	Telemetry/RS232 TxD signal return
J2-6	RXD	SCI receive	RS232 RxD signal
J2-7	+5V_DIG	digital 5VDC	powers RS232 driver/receiver in prog/com cable
J2-8	5VRTN	digital 5V return	powers RS232 driver/receiver in prog/com cable
J2-9	MODE	program mode	Input pin used to put CC200 in program or run mode
J2-10	DISC0+	discrete input	Connection Present signal
J2-11	DISC0-	discrete input rtn	Connection Present return
J2-12	DISC1+	discrete input	
J2-13	DISC1-	discrete input rtn	
J2-14	DISC2+	discrete input	
J2-15	DISC2-	discrete input rtn	
J2-16	DISC3+	discrete input	
J2-17	DISC3-	discrete input rtn	
J2-18	DISC4+	discrete input	External shut down signal
J2-19	DISC4-	discrete input rtn	External shut down return
J2-20	DISC5+	discrete input	
J2-21	DISC5-	discrete input rtn	
J2-22	DISC6+	discrete input	
J2-23	DISC6-	discrete input rtn	
J2-24	PWM0_BIAS	bias voltage input	pulse width modulation bias voltage input
J2-25	PWM0_OUT	pwm output	pulse width modulated output
J2-26	PWM0_RTN	bias voltage return	pulse width modulation bias voltage input return
J2-27	PWM1_BIAS	bias voltage input	pulse width modulation bias voltage input
J2-28	PWM1_OUT	pwm output	pulse width modulated output
J2-29	PWM1_RTN	bias voltage return	pulse width modulation bias voltage input return
J2-30	VDAC	dac voltage output	
J2-31	VDAC_RTN	dac output rtn	
J2-32	VPPE	input pin	either 12.25V input for eprom programming, or xirq/ TTL level input to microcontroller; jumper configured.

TABLE 3.0

PIN FUNCTIONS

NUMBER	NAME	FUNCTION	DESCRIPTION
J3-1	ISENSE+	pack current sense	differential signal for pack current sense
J3-2	NOT USED		
J3-3	VAUX-	input power return	nominally 12V RTN of auxiliary battery
J3-4	LOSIDE6	low side switch input	referenced to 12V RTN
J3-5	VAUX+	input power	nominally +12VDC of auxiliary battery
J3-6	LOSIDE5	low side switch input	referenced to 12V RTN
J3-7	LOSIDE4	low side switch input	referenced to 12V RTN
J3-8	LOSIDE3	low side switch input	referenced to 12V RTN
J3-9	LOSIDE2	low side switch input	referenced to 12V RTN
J3-10	LOSIDE1	low side switch input	referenced to 12V RTN
J3-11	LOSIDE0	low side switch input	referenced to 12V RTN
J3-12	RELAY0_B	spdt relay contact	energized connection
J3-13	RELAY0_COM	spdt relay wiper	
J3-14	RELAY0_A	spdt relay contact	non-energized connection
J3-15	NOT USED		
J3-16	VSENSE+	pack voltage sense	differential signal for pack voltage sense
J3-17	ISENSE-	pack current sense	differential signal for pack current sense
J3-18	NOT USED		
J3-19	ISHIELD	pack current sense shield	referenced to 12V RTN
J3-20	VAUX-	input power return	nominally 12V RTN of auxiliary battery
J3-21	VAUX+	input power	nominally +12VDC of auxiliary battery
J3-22	HISIDE0	high side switch output	switches +12VDC of auxiliary battery
J3-23	HISIDE1	high side switch output	switches +12VDC of auxiliary battery
J3-24	RELAY1_A1	dpdt relay contact	non-energized connection
J3-25	RELAY1_COM1	dpdt relay wiper	
J3-26	RELAY1_B1	dpdt relay contact	energized connection
J3-27	RELAY1_A2	dpdt relay contact	non-energized connection
J3-28	RELAY1_COM2	dpdt relay wiper	
J3-29	RELAY1_B2	dpdt relay contact	energized connection
J3-30	NOT USED		
J3-31	VSENSE-	pack voltage sense	differential signal for pack voltage sense
J3-32	NOT USED		

TABLE 4.0

Memory Configuration

The CC200 has several memory mapping options available. Which one to choose will depend on user program, data space requirements, and the components installed in the controller.

Table 5.0 summarizes the choices and the hardware jumper configurations that are required (jumper defaults and hardware installed is in gray).

Memory (bytes)	Jumper Configuration
15,616 SRAM, 49088 FLASH	E2/E3, E8/E9
15,616 SRAM, 262,144 FLASH, 24,512 OTPROM	E1/E2, E7/E8
15,616 SRAM, 131,072 FLASH, 24,512 OTPROM	E1/E2, E7/E8

TABLE 5.0

For a detailed programmer's model, refer to Table 6.0 for a 64K address space summary, and Table 7.0. for an extended addessing memory map.

Memory Types

The CC200 has four memory types available for program or data storage: static ram (SRAM), electrically eraseable programmable read only memory (EEPROM), FLASH, and one-time programmable read only memory (OTPROM).

SRAM is intended to be used for volatile data storage. A part of this memory should also be allocated for the stack. SRAM resides both internal and external to the microcontroller. The CC200 can accept SRAM densities of 8K (no jumpers required), and 32K (jumper E4 to E5).

EEPROM can be used for non-volatile program and data storage. It is byte eraseable and byte programmable. CC200 eeprom resides both internal and external to the microcontroller. External EEPROM is accessed using the serial peripheral interface (SPI) bus, and is intended for calibration and battery parameter storage.

FLASH is non-volatile memory for program storage. It resides external to the microcontroller and must be bulk erased before it can be byte programmed. The CC200 can accept FLASH densities of 64K (jumper E2 to E3 and E8 to E9), 131K (jumper E1 to E2 and E7 to E8), or 262K (jumper E1 to E2 and E7 to E8).

OTPROM is non-volatile memory for program storage which is only available when using the extended addressing memory map configuration. To program OTPROM, jumper E26 to E27 and apply an externally generated +12.25VDC power source to J2-32. For normal operation remove jumper E26 to E27. Code housed in OTPROM is non-banked, executes faster, and therefore should be reserved for frequently called low-level drivers and library routines. CC200 OTPROM resides internal to the microcontroller.

ADDRESS	MEMORY DESCRIPTION	SIZE (BYTES)	CHIP SEL PIN USED
0000 007F	CPU REGISTER BLOCK	128	-
0080 037F	Internal Ram	768	
0380 0D7F	External Ram	2560	CSGP1/
0D80 0FFF	Internal EEPROM	640	-
1000 1FFF	External Ram	4096	CSGP1/
2000 3FFF	External Ram (E4/E5) or Spare (E5/E6)	8192	CSGP1/
4000 7FFF	External FLASH	16384	CSPROG/ *
8000 FFBF	External FLASH	32704	CSPROG/
FFC0 FFFF	External FLASH (interrupt vectors)	64	CSPROG/

* CSPROG/ = CSPROG/ + CSGP2/

TABLE 6.0 64K ADDRESS SPACE

ADDRESS	MEMORY DESCRIPTION	SIZE (BYTES)	CHIP SEL PIN USED
0000 007F	CPU REGISTER BLOCK	128	-
0080 037F	Internal Ram	768	
0380 0D7F	External Ram	2560	CSGP1/
0D80 0FAB	Internal EEPROM	556	-
0FAC 0FFF	Internal EEPROM (pseudo interrupt vectors)	84	-
1000 3FFF	External Ram	12288	CSGP1/
4000 7FFF	External FLASH (banked)	16384	CSPROG/
8000 9FFF **	External FLASH (non-banked)	8192	CSPROG/ *
A000 FFBF	Internal OTPROM	24512	-
FFC0 FFFF	Internal OTPROM (interrupt vectors)	64	-

* CSPROG/ = CSPROG/ + CSGP2/

0x8000 - 0x9FFF   0x4000 - 0x7FFF

** Non-banked is same as banked segment 6 (14) for 1Mbit (2Mbit) FLASH

START ADDR	BANK #	END ADDR
00000	0	03FFF
04000	1	07FFF
08000	2	0BFFF
0C000	3	0FFFF
10000	4	13FFF
14000	5	17FFF
18000	6	1BFFF
1C000	7	1FFFF
20000	8	23FFF
24000	9	27FFF
28000	10	2BFFF
2C000	11	2FFFF
30000	12	33FFF
34000	13	37FFF
38000	14	3BFFF
3C000	15	3FFFF

TABLE 7.0 EXTENDED ADDRESSING

Programming the CC200

The CC200 is programmed using a personal computer with a serial port, a Programming Support Utilities System Diskette, and a programming cable as shown in Figure 1.0. The program file that gets downloaded to the CC200 must be a Motorola S-record format. A sample linker command file and the resulting S-record file are provided in Figures 2.0 and 3.0 for reference.

To program using bank switching:
Ensure programming cable switch S1 is closed and type the following DOS commands:

```
a:\rtslo -c<com port> <cr>
a:\ldsev262 -f<filename.hex>
        c<com port>
        -r<cr>
```

where,

com port = com1 or com2
filename.hex = s-record file
-r= programming of OTP

To program using 64K address space:
Ensure programming cable switch S1 is closed and type the following :

```
a:\rtslo -c<com port> <cr>
a:\ldsev49 -f<filename.hex>
        c<com port> <cr>
```

where,

com port = com1 or com2
filename.hex = s-record file

Communicating with the CC200

J1850

The CC200 utilizes the J1850 to communicate to the charger and sometimes, depending on the controller configuration, the user development system. Charger control and status data is periodically received and can be transmitted over the J1850 at frequency that is user definable. Refer to Table 3.0 for connector pin details. The J1850 also is capable of user development communications, where diagnostic, charging status, and user editing screens are available for customizing the controller to the users needs. The user development communications is only transmitted over the J1850 when the controller is used in conjunction with the Distributed Energy Management Modules, otherwise, development communications is provided via the SCI interface.

SCI

The serial communication interface (SCI) is a universal asynchronous receiver transmitter (UART) and can operate at speeds of up to 19200 bits/sec. Refer to Table 3.0 for connector pin details. The SCI is used to perform three functions: Programming the controller, user development interface, and distributed energy management module (DEM) communication. The functions are performed independently of each other and cannot be done simultaneously.

Programming is done as described in the programming section. The controller is brought up in bootstrap mode and data is downloaded via non-isolated input pins using a level translator, cable, and laptop computer. Typically, the controller is loaded at the factory and will be ready to run when received by the customer.

The user development function is used when the development software package is installed in the controller. The software provides the user with the ability to access diagnostic screens for monitoring system problems, charging status to keep track of pack voltages or temperatures, and the user editor screens to customize the controllers charging modes and trip/set points. Communication takes place via the non-isolated input pins using a level translator, cable, and laptop computer.

The Distributed energy management module communications is used in conjunction with the DEM modules placed within the battery pack. The controller can communicate with each of the modules via an isolated input and non-isolated output pin. The DEM modules, which can monitor individual battery voltage and temperature, are each connected in series with each other. Each module in succession provides isolation from the previous one until the loop is completed by returning back to the controllers receive pin. See distributed energy management module application manual for more details regarding this configuration.

Modes of Operation

Bootstrap Mode

This mode is used to program a CC200. The CC200 is put into this mode by placing the programming cable switch S1 in the "download" position, and by typing the following DOS command :

a:\rtslo -c<com port> <cr>

After applying power, the loading utility downloads a bootloader program (less than 768 bytes) into microcontroller internal ram. As soon as the last byte is sent, this bootloader program begins to execute. It places the CC200 into a mode where it will program user firmware to the appropriate memory device. User firmware must be in a Motorola S-record format.

Normal Mode

This is the operational mode of the CC200, in which all its capabilities are fully functional. To enter this mode after successfully programming the CC200, place the programming cable switch S1 in the "monitor" position and apply power.

Traction Battery Voltage Sensing

The CC200 can sense battery pack voltage in one of two ways. The first consists of utilizing CC200 hardware the other is to use DEM modules.

The CC200 can sense pack voltage (ΔVSENSE) of up to 500 VDC with an resolution of 122 mV/lsb. The circuitry is optically isolated from the 12vdc auxiliary battery and can withstand 5000Vrms of potential difference.

Utilizing external Distributed Energy Management Modules (DEMS), battery pack voltage can be determined by collecting individual battery voltages and adding them together in the controller. The CC200 provides an optically isolated (5000VDC minimum) receiver input to interface with external DEM circuitry.

The CC200s voltage sensing circuitry is also used to detect a short between chassis (12V RTN) and a battery terminal in the pack. The value that determines a fault can be controlled by the user in the development configuration.

User firmware reads pack voltage over an isolated SPI bus. The maximum spi clock frequency that can be used is 200KHz.

Refer to Table 4.0 for connector pin details.

Traction Battery Current Sensing

The CC200 has two current sensing channels, fine and course. The range of measurable current is a function of the external shunt resistor used by the vehicle. The CC200 was designed for a 1 milliohm shunt resistor.

For the coarse channel the relationship is as follows:

$$-\frac{2.5}{R_{SH}} \leq ISENSE \leq +\frac{2.5}{R_{SH}}$$

With R_{SH}=1mΩ, the coarse channel can measure from -2500A to +2500A, with an accuracy of 1.22A/lsb.

More accurate measurements can be obtained using the fine channel. The dynamic range of the fine channel is described below.

$$-\frac{2.5}{R_{SH} * G} \leq ISENSE \leq +\frac{2.5}{R_{SH} * G}$$

where,

$$G = \frac{49.4K\Omega}{R_{147}} + 1$$

R_{147} is a resistor internal to the CC200, whose value is 2KΩ.

With R_{SH}=1mΩ and R_{147}=2KΩ, the fine channel can measure from -100A to +100A, with an accuracy of 0.047A/lsb.

User firmware also reads pack current over an isolated SPI bus. The maximum spi clock frequency that can be used is 320Khz. Refer to Table 4.0 for connector pin details.

Temperature Measurement

The temperature measurement capability of the CC200 consists of nine thermistor inputs. The nominal value of the thermistor should be 10KΩ. The relationship between thermistor resistance and a/d output data byte is as follows :

$$\frac{R_T + 100}{R_T + 10.1K} * 256 = Dout$$

Refer to Table 2.0 for connector pin details.

Charge port and conversion box temperatures are monitored and the controller will stop charging when a temperature of 85 degrees centigrade is reached.

Battery pack temperature can be monitored using the remaining seven sensors. The controller monitors a maximum battery pack temperature parameter and will stop charging if this temperature is reached or exceeded. The value of the parameter can be controlled using the development system. The default for MAX battery pack temperature is 59 degrees centigrade.

If selected, the controller has the capability to perform temperature compensation. The controllers algorithm selects the highest battery pack temperature read from the seven inputs, and the temperature compensation parameter to compute the compensation value. The temperature compensation parameter can be modified with the development system based on the users needs. The controllers default temperature compensation parameter is 4.44 millivolts per cell per degree centigrade at a nominal temperature of 25 degrees centigrade.

Discrete Input Measurement

The CC200 has the capability of measuring up to seven discrete inputs. Each input is optically isolated (5000VDC) from the measurement circuitry, and is designed for either a +5Vdc or +12Vdc signal and its respective return. Two discretes are currently used by the controller. DISC0 is used to detect connection present (paddle fully inserted into the charge port). The controller will remain asleep and not wake up until the connection present is detected. DISC4 is provided as a interlock and will stop the charging sequence and put the controller to sleep until it is removed.

High-Side Switching

The CC200 comes with two high-side switches, each capable of switching +12Vdc into a 1.1A load. See Table 4.0 for connector pin details.

Low-Side Switching

The CC200 has the capability of providing seven monolithic low-side switches. Each switch can handle 0.5A with all outputs equally conducting. Currently, two of the outputs are used. LOSIDE0 is used to control the charge port cooling fans. And LOSIDE1 is used to control the conversion box cooling fan. Once connection present is detected and charging is required the fans on the charge port and conversion box will be switched on. A short circuit protection feature is provided on all low-side switches. Jumpering E22 to E23 causes the respective output to immediately shut down upon sensing a short circuit (3A to 6A). By removing the jumper(default configuration) the output remains on in current limiting mode until the output short condition is removed or the chip thermal shutdown is reached. The low side switches are 12V RTN referenced. See Table 4.0 for connector pin details.

Relay Switching

Two relays are provided in each CC200. A single-pole-double-throw (SPDT) relay is capable of switching 5A. A double-pole-double-throw (DPDT) relay is capable of switching 5A for each pole. The SPDT relay is currently used as a interlock for the Dolphin inverter. The relay is configured such that the interlock goes through the normally closed contacts. The DPDT relay is used for switched power for the Dolphin inverter or other systems requiring switched 12vdc. This relay is configured such that 12vdc is supplied during charging mode. Once the controller goes to sleep the relay in de-energized and 12vdc is removed. The wiper and contacts for both relays are accessible on connector J3. See Table 4.0 for connector pin details.

Hydrogen Gas Sensing

The CC200 has interface circuitry to accommodate two Hydrogen gas sensors made by Figaro, p/n TGS821. Sensor heating requirements are met using +5VDC and biasing requirements are met using a +5V reference. Sensor outputs are measured using an 8-bit unipolar a/d. Refer to Table 2.0 for connector pin details.

Digital-to-Analog Converter

The CC200 has a 12-bit dac with an output voltage range of -5 to +4.9976VDC. The data format is 2's complement as shown below:

Input Data Word MSB LSB	Analog Output Voltage (VDC)
0111 1111 1111	+5V * (2047/2048)
0000 0000 0001	+5V * (1/2048)
0000 0000 0000	0V
1111 1111 1111	-5V * (1/2048)
1000 0000 0001	-5V * (2047/2048)
1000 0000 0000	-5V*(2048/2048)

Table 8.0 DAC Code Table

Refer to Table 3.0 for connector pin details.

Pulse-Width Modulation

The CC200 has two pwm outputs. Duty cycle is programmable from 0.0 to 1.0. Frequency can vary from 3.83 Hz to 10KHz. Either one of two clock sources can be used to derive the pwm signal. If clock A is used, the relationship between output frequency and pwm registers is as follows :

$$f_{PWM} = \frac{\text{clock A}}{\text{PWPER}}$$

where,

clock A = E, E/2, E/4, or E/8
PWPER = HC11 period register

If clock S is selected, then

$$f_{PWM} = \frac{\text{clock A}}{2 * \text{PWSCAL} * \text{PWPER}}$$

where,

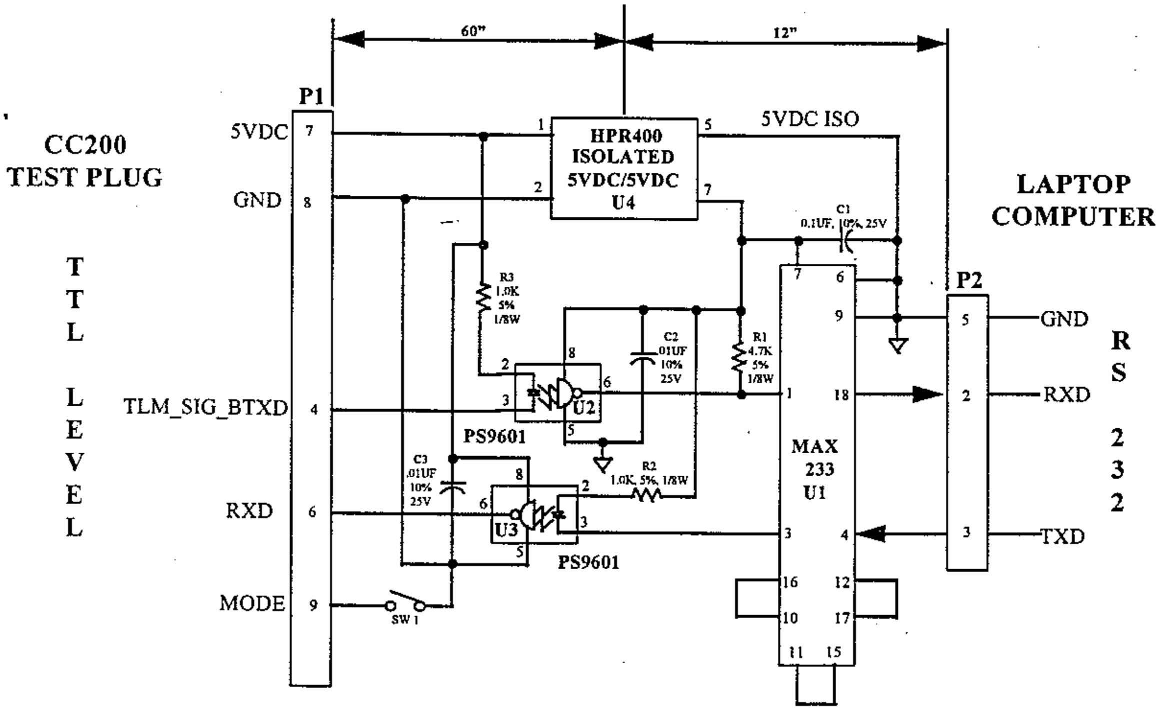
clock A = E, E/2, E/4, or E/8
PWSCAL = HC11 pwm prescaler
PWPER = HC11 period register

Frequency/Pulsewidth Measurement

The CC200 has a frequency/ pulsewidth measurement capability utilizing a bi-directional digital input/output pin. When configured as an input, it can be used to measure TTL level signal frequency or pulse width. Refer to table 1.0 for input voltage specifications.

NOTES:

1. Use 22 awg wire.
2. Use suggested part numbers or equivalent.
3. Make cable approximately 6 feet.



REF. DES.	QTY	PART NUMBER	DESCRIPTION	VENDOR
P1	1	205204-1	9-pin male connector	AMP
P1	1	66506	Male pins, 24-20 awg	AMP
P2	1	205203-1	9-pin female connector	AMP
P2	4	66504	Female sockets, 24-20 awg	AMP
P1, P2	2	749914-2	Back shell	AMP
U1	1	MAX233CPP	TTl/RS232 level translator	MAXIM
SW1	1	CKN1023-ND	SPDT Switch	DIGI-KEY
C1	1	P4201-ND	.1UF,25V, 80/-20% Ceramic disc cap	DIGI-KEY
C2, C3	2	P4300A-ND	.01UF,25V, 80/-20% Ceramic disc cap	DIGI-KEY
R1	1	4.7KEBK-ND	4.7Kohm, 1/8W, 5% resistor	DIGI-KEY
R2, R3	2	1.0KEBK-ND	1.0Kohm, 1/8W, 5% resistor	DIGI-KEY
U2, U3	2	PS9601NEC-ND	High speed opto isolators	DIGI-KEY
U4	1	HPR400	Isolated 5vdc/5vdc power reg.	Power Convertibles
	2	ED56083-ND	IC Socket 8 pin low profile	DIGI-KEY
	1	ED56203-ND	IC Socket- 20 pin low profile	DIGI-KEY

```
# main262a.lnk                                    23 July 96 Rony Mansour
#                                                 link command file for Charge Controller 200 program
+h                                                # multi-segment output
+map=main262a.map
-cd
-cb
-o main262a.h11
-ps 14 -pc .
+data -b0x1000 -m0x3000
+text -b0x00000 -o 0x4000 -m0x4000                # bank 0
main262.o                                         # cc200 main utility, calls other utilities
+text -b0x04000 -o 0x4000 -m0x4000                # bank 1
vsrom62.o                                         # serial eeprom utilities
+text -b0x08000 -o 0x4000 -m0x4000                # bank 2
evdac262.o                                        # serial dac utilities
+text -b0x0c000 -o 0x4000 -m0x4000                # bank 3
evsw262.o                                         # high side, low side, & relay drivers
+text -b0x10000 -o 0x4000 -m0x4000                # bank 4
evutil62.o                                        # misc. utilities : read, write int. eeprom
+text -b0x14000 -o 0x4000 -m0x4000                # bank 5
ev185062.o                                            # j1850 routines
+text -b0x18000 -o 0x4000 -m0x4000                # bank 6
evirqsr.o                                         # irq service routines
+text -b0x1c000 -o 0x4000 -m0x4000                # bank 7
evpwm62.o
+text -b0x20000 -o 0x4000 -m0x4000                # bank 8
#evlibi.h11                                       # libi.h11 less getchar & putchar routines
+text -b0x24000 -o 0x4000 -m0x4000                # bank 9
#errcode.o
+text -b0x28000 -o 0x4000 -m0x4000                # bank 10
#parsecom.o
+text -b0x2c000 -o 0x4000 -m0x4000                # bank 11
#toupper.o
+text -b0x30000 -o 0x4000 -m0x4000                # bank 12
#status.o
+text -b0x34000 -o 0x4000 -m0x4000                # bank 13
#wiperom.o
+text -b0x38000 -o 0x4000 -m0x4000                # bank 14
#c:\c\lib\libd\libd.h11
+text -b0x3c000 -o 0x4000 -m0x4000                # bank 15
#c:\c\lib\libi\libi.h11
+def __pdata=__text__
+text -b0x0d81 -o0x0d81                           # internal eeprom (startup code)
evsu262.o                                         # startup code
evirqsrs.o                                        # irq service routine shells
+text -b0x0f96 -o0x0f96                           # internal eeprom (pseudo irq vectors)
evirq262.o                                        # pseudo interrupt vectors
+text -b0xffc0 -o0xffc0                           # internal eprom (irq vectors)
evromirq.o                                        # rom-based interrupt vectors
+text -b0xa001 -o0xa001                           # internal eprom
#c:\c\lib\libm\libm.h11                           # complete machine library except wcalk, wcalc
wcalk.o                                           # customized for window #2 usage
getputch.o                                        # customized getchar & putchar routines
c:\c\lib\libd\libd.h11
c:\c\lib\libi\libi.h11
c:\c\lib\libm\libm.h11
+def __memory=__bss__
```

FIGURE 2
Linker Command File for 256K memory map

S00F00006D61696E323632612E74706DD1
S224000000000054004000740F40A0D09204D656E75204F7074696F6E73200D0A002D2D2D2D8B
S2240000202D2D2D2D2D2D2D2D2D2D2D2D2D2D2D2D2D2D2D2D2D2D2D2D2D2D2D2D2D2D2D2D1B
S2240000402D2D2D2D2D2D2D2D2D2D2D2D2D2D2D2D2D0D0A002020202020202020312E2020DB
S2240000602020202020726561642F7772697465207365726961C20656570726F6D0A0D002086
S224004040014F18CE4000BDA010BDD4FF05BDA001E6074F18CE4000BDA010BDD4FF05BDA0C4
S22400406001E6094F18CE4000BDA010BDD4FF05147C040E3838393C37363430E602D72A1380
S2240040802980FCD62AE7004F313838393C373634300F157C04BDA001CC000318CE4000BDA5
S22400C0003C373630EC00830001260CEC06270514000120031500001EC00260CEC06270514E4
S22400C02000022003150002383839400000000000000000014076000340EB3C37363C30E6F6
S2240102403C373630C616D73B1AEE00186F00C617D73BBDA00118CE423CBDA010BDD4FF058F
S2240102605FD73B3838393C373630C602D73BE6071AEE0018E700C603D73BBDA00118CE4256
S2240102803CBDA010BDD4FF055FD73B3838393C3C30CC3FFFED00EC002707830001ED0020B6
S224014060BDD4FF05E700C5082713E6014F3736E6003736CC4024BDA25431313131BDA001BC
S22401C06078EC1BED02EC00C3FFF6188FEC1D183CBDD1FAEC00C3FFF63736CC40083736BDCD
S22400C0C118E6001AEE346C3526026C3418E7006C3926026C386C3726D36C3620CFC6651AD5
S22400C0E1EE346C3526026C3418E700EC2626234F5FED32201DEC361AA3442CE1C6301AEE1D

FIGURE 3.0
Motorola S-record File Excerpt