MotionChip™ II QFP100

T E C H N O S O F T

Data Sheet

Preliminary

TECHNOSOFT

MotionChip™ II – QFP100
Data Sheet

P091.055.MCII-QFP100.DSH.0206
February 2006

Technosoft S.A.
Buchaux 38
CH-2022 BEVAIX
Switzerland
Tel.: +41 (0) 32 732 5500
Fax: +41 (0) 32 732 5504
contact@technosoftmotion.com
www.technosoftmotion.com/

Read This First

Whilst Technosoft believes that the information and guidance given in this manual is correct, all parties must rely upon their own skill and judgment when making use of it. Technosoft does not assume any liability to anyone for any loss or damage caused by any error or omission in the work, whether such error or omission is the result of negligence or any other cause. Any and all such liability is disclaimed.

If you Need Assistance …

If you want to …	Contact Technosoft at …
Visit Technosoft online	World Wide Web: http://www.technosoftmotion.com/
Receive general information or assistance	World Wide Web: http://www.technosoftmotion.com/ Email: contact@technosoftmotion.com
Ask questions about product operation or report suspected problems	Fax: (41) 32 732 55 04 Email: hotline@technosoftmotion.com
Make suggestions about or report errors in documentation	Mail: Technosoft SA Buchaux 38 CH-2022 Bevaix, NE Switzerland

Contents

Figures

Tables

This page is empty

1. Features

- High-performance ready to run DSP Motion Controller, 40MHz, 40MIPS
- Single-Chip solution for control of
  - DC Brush Motors
  - DC Brushless Motors
  - AC Brushless Motors (PMSM)
- Operation Modes
  - Stand-alone – executes motion sequences from a local memory
  - Slave – multi-axis cases
- Communication channels:
  - Serial RS-232 / RS-485
  - CAN-bus
- Flexible structure allowing:
  - Position control
  - Speed control
  - Torque control (field-oriented for AC motors)
  - Open-loop (V/f for AC motors)
- Typical sampling rates: 10kHz torque loop, 1kHz speed/position loop
- Feedback signals
  - 1- 3 Currents
  - Digital Hall sensors
  - Linear Hall sensors
  - Position read from incremental encoder (on-chip interface)
  - Speed estimated from position
  - Two temperature sensors
  - DC-bus voltage (V_{DC})
- High-level programming using the Technosoft Motion Language (TML)
- No DSP Code Development Required
- 23 motion modes, decision blocks, functions, arithmetic & logic unit
- Accurate profile generator with automatic round-off correction
  - Position range: 32-bits
  - Speed/acceleration range: 16-bit integer part, 16-bit fractional part
- 18 programmable events were motion parameters and/or modes can be changed on-the-fly
- 12 programmable TML interrupts
- Main Inputs:
  - Power Drive Fault (interrupt input)
  - Enable/Disable (interrupt input)
  - 2 Limit switches (interrupt inputs)
  - Encoder Index (capture input)
  - 2^{nd} Encoder Index (capture input)
  - Start mode: AUTORUN / Wait commands
- Main Outputs:
  - 4-6 PWM commands
  - Brake transistor command
  - Interrupt to host
  - Ready
- Up to 19 general-purpose I/O
- Advanced PWM command methods
  - V_{DC} variation compensation;
  - Dead-time compensation;
  - 3^{rd} harmonic injection;
  - Wobbling for EMI reduction
- Brake transistor control
- Integrated protections: over current, over voltage, under voltage, over temperature, I^2t,
- Development tools available:
  - IPM Motion Studio including:
    - Motor and drive configuration dialogs;
    - Motion sequence setup wizard;
    - Compiler, command interpreter;
    - Graphical display for traced variables, watch functions.
  - Application notes (web page)
  - Reference designs (CD)

This page is empty

2. General Description

The Technosoft **MotionChip™ II – PQF100** (part number *P055.001.E006*) is a high performance ready-to-run motion controller, based on the latest DSP structures. It does not require any DSP Code development. With the advantage to contain all necessary configurable interfaces on a single chip, the MotionChip™ II is ideal for rapid and cost effective design of fully digital, intelligent drives for various motor types.

The MotionChip™ II can:

- Operate in a stand-alone or in a master / slave, multiple axis configuration
- Control three motor types: DC brush, DC brushless, AC brushless (PMSM)
- Implement various command structures: open loop, torque, speed, position / external loop control
- Work with different motion and protection sensors (position, speed, current, torque, voltage, temperature)
- Use different communication channels such as SCI for RS232/485 and CAN-bus
- Execute advanced motion language commands and motion sequences

Intended to cover a major part of basic and complex motion applications, the MotionChip™ II has the special advantage to be a highly flexible structure at the level of:

- Motion structure configuration (selection of motor technology, control type, sensors type)
- Motion implementation with high-level motion language commands

Relative to existing solutions the MotionChip™ II offers many advantages:

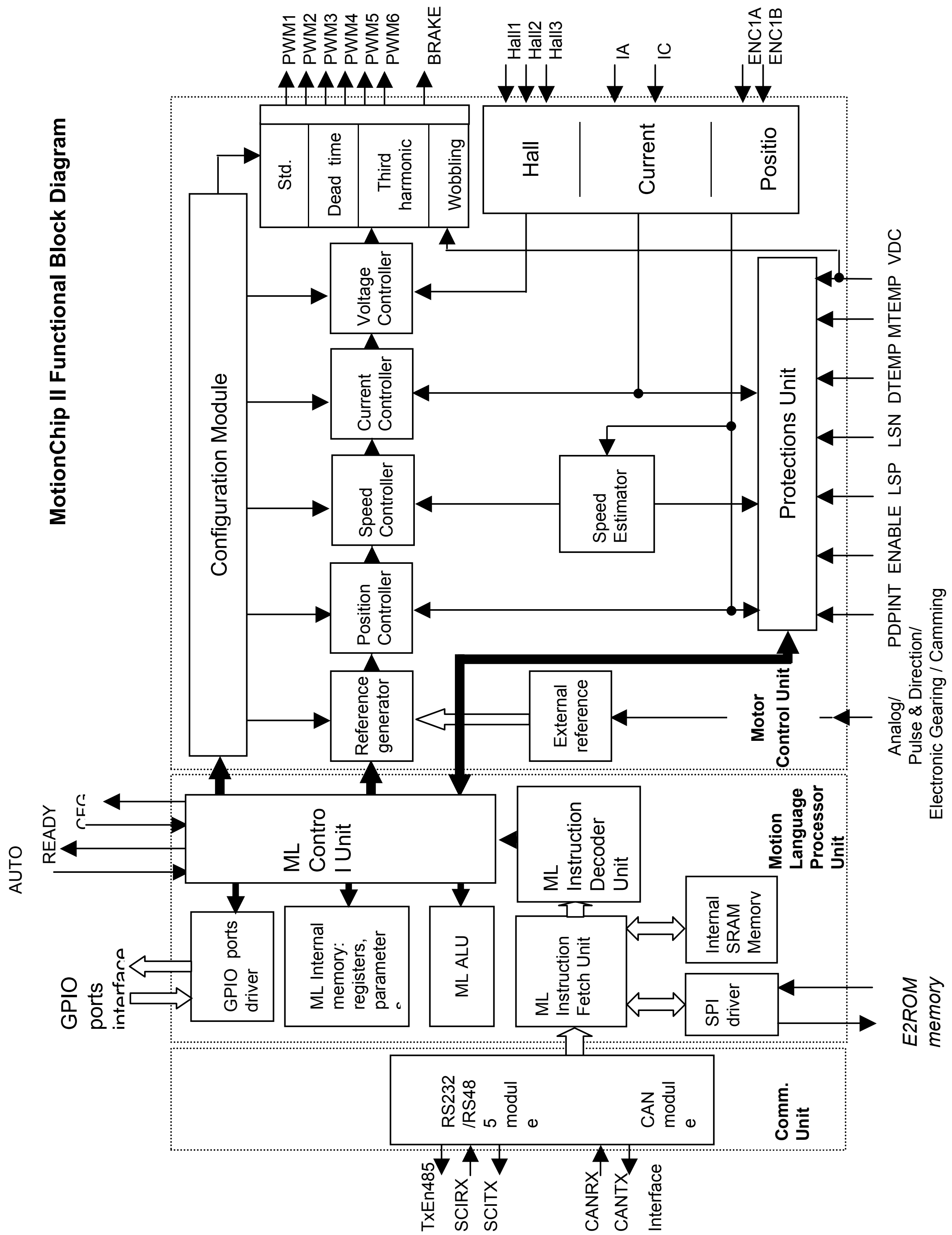
- Usable for different motor technologies
- Implementation of multiple motion control configurations, including vector control for AC drives
- Implementation of complete digital control loops, including current/torque control
- Powerful motion language including 23 motion modes, decision blocks, function calls, event-driven motion updates, interrupts
- Stand-alone or slave operation
- Minimal requirements for setup configuration and use
- Easy embedding in user hardware structures
- Software-less device (no programming effort required)
- High-level development tools for application setup, test and debug

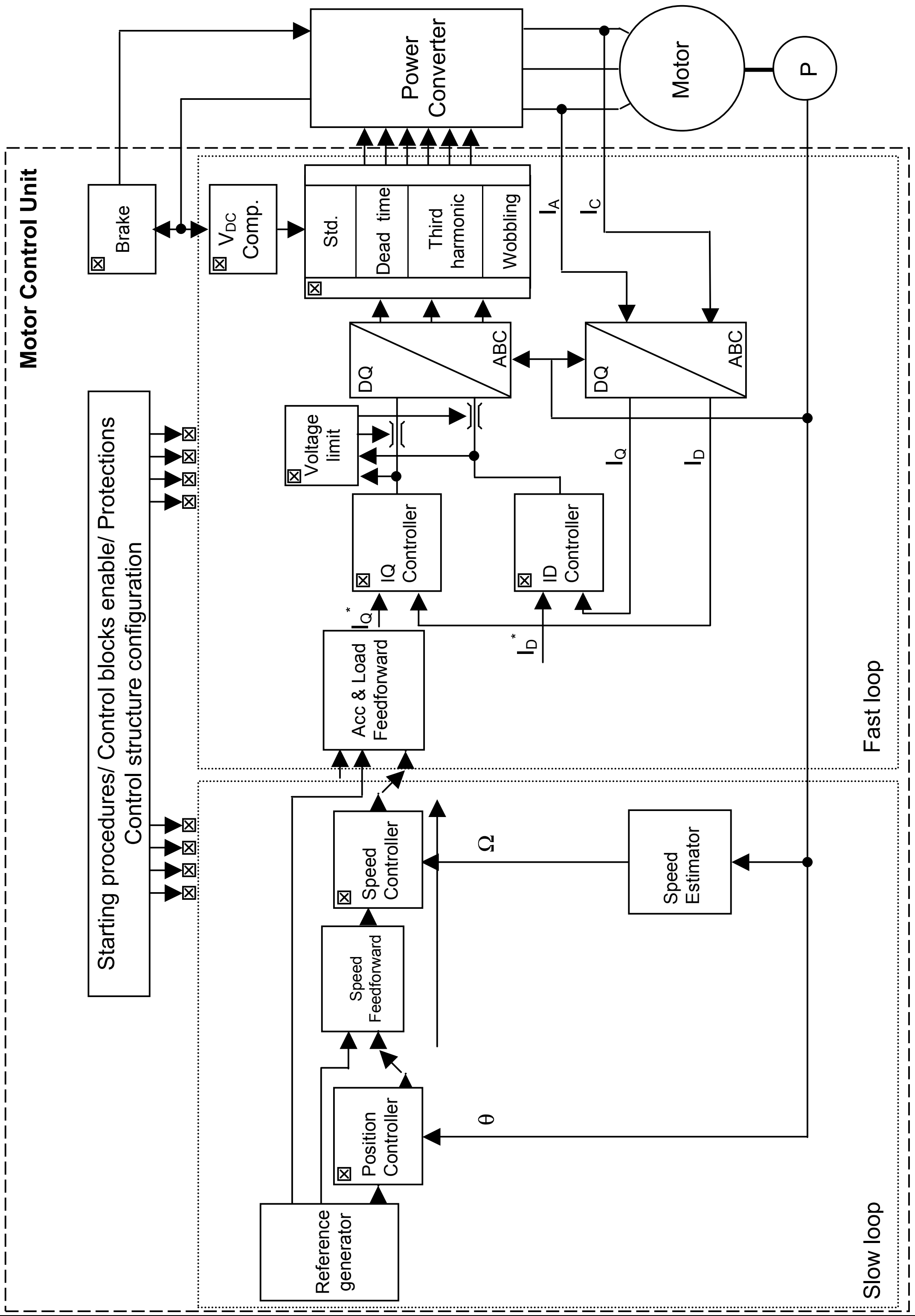
2.1. Functional Block Diagram

The functional block diagram provides a high-level description of the basic components of the device. The MotionChip II has three functional units:

- the communication unit, through which the MotionChip II can receive and send motion language commands using serial (RS-232 or RS-485) or CAN-bus;
- the “motion language processor” unit which decodes motion language commands received from communication unit or fetches them from a local memory
- the motor control unit that executes the motion commands. The motor control unit diagram presents in detail the components of this unit.

MotionChip II Functional Block Diagram





Motor Control Unit
Brake
V_{DC} Comp.
Power Converter
Motor
P
Std.
Dead time
Third harmonic
Wobbling
I_A
I_C
DQ
ABC
DQ
ABC
Voltage limit
IQ Controller
ID Controller
I_Q
I_D
I_Q^*
I_D^*
Acc & Load Feedforward
Starting procedures/ Control blocks enable/ Protections
Control structure configuration
Fast loop
Speed Controller
Ω
Speed Estimator
Speed Feedforward
Position Controller
θ
Reference generator
Slow loop

2.2. Motion Configurations

MotionChip II is designed to control brushless motors and brush DC motors.
Brushless motors are vector controlled and can be driven with sinusoidal commutation (PMSM) or with trapezoidal commutation (BLDC).

Each type of motor can be controlled in position, speed and current/torque closed loop or can be commanded in open loop.

The MotionChip II provides 23 motion modes. Each motion mode designates a reference mode and a control structure. Table 2.1 summarizes for each type of control the reference modes accepted.

Table 2.1. *Motion modes*

Motion Modes	Control Type			
	Position	**Speed**	**Torque**	**Voltage**
Position Profiles	√	–	–	–
Speed Profiles	–	√	–	–
Contouring (point to point with linear interpolation)	√	√	√	√
External, reference read from the analogue input REFERENCE	√	√	√ SL	√ SL
			√ FL	√ SL
Pulse and direction	√	√	–	–
Electronic Gearing/Camming – master	√	–	–	–
Electronic Gearing – slave	√	√	–	–
Electronic Camming – slave	√	–	–	–
Stop	–	√	√	√
Test (limited ramp)	–	–	√ FL	√ FL

Legend:
SL – Reference update is done on the slow (position/speed) loop
FL – Reference update is done on the fast (current/torque) loop

2.3. Controlled Loops

The MotionChip II control structure includes three control loops. The outer loop is used for motor position control. The outer loop controller is a PID with filter on the derivative term. The PID output can be used as speed or torque command for the motor.

The middle loop implements the speed control. The speed loop controller is a PI with speed feedforward. The speed loop controller output it is used as torque command for the motor.

The inner loop performs the current/torque control. For PMSM motors, torque control is performed using a field oriented control scheme. The inner loop has two PI controllers one for torque control (Q axis controller) and the other for flux control (D axis controller). The inner loop provides a voltage command, which is translated into PWM commands.

2.4. Sensors

Table 2.2 presents the categories of sensors accepted by the MotionChip II, where are they used, their type and the required interface or connections.

Table 2.2. *MotionChip™ II accepted sensors*

Category	Usage	Type	Interface/Connections
Position	Position feedback	Incremental encoder	On-chip. Dedicated inputs: ENC1A for A, ENC1B for B, ENC1Z for Z (if available)
		3 Linear Hall, 120° apart	Dedicated inputs: LH1, LH2, LH3
Speed	Speed feedback	Computed from position	The position sensor interface
Current	Current feedback	1/2 current transducers on motor lines or lower legs of the inverter	Dedicated inputs: I_A, I_C
Temperature	Motor and/or power stage protection	1/2 analogue sensors	Dedicated inputs: MTEMP and DTEMP
Digital Hall	Commutation	120° apart	Dedicated inputs: HALL1, HALL2, HALL3
Voltage	Over-voltage and under-voltage protections. PWM control with VDC compensation	Analogue sensor	Dedicated input: VDC

2.5. Special Features

The MotionChip II includes a set of programmable special features. These offer the possibility to select different strategies according with the application specific. Table 2.3 summarizes these features.

Table 2.3. *Special features*

Function	Options
Current Offset Detection	Automatic, with motor supplied (PWM outputs active)
	Automatic, without motor supplied (PWM outputs inactive)
	Automatic detection is disabled. User provides the offset
Start method for the brushless AC (PMSM) motors	Motor is aligned on phase A, by injecting a current in phases B and A
	Motor is aligned on phase A, by applying a voltage on phases B and A
	Motor starts as a brushless DC using Hall commutation. After first Hall transition, commutes to brushless AC mode (FOC, sinusoidal currents)
PWM command Techniques	With compensation of DC-bus voltage variation (needs DC-bus voltage sensor)
	With compensation of dead-time
	With 3rd harmonic injection to increase maximum applicable voltage

	With PWM frequency wobbling around programmed value to reduce EMI (electromagnetic interference)
Brake command	DC-bus voltage control during motor brakes, by command of a brake transistor

PWM Frequency and Sampling Rates

The MotionChip II uses a fast loop for current/torque control and a slow loop for position/speed sampling. The sampling rates of these loops are synchronized and linked in a fixed ratio with the PWM frequency in order to eliminate the beat-frequency problems. The maximum sampling frequency on the fast loop can be half of the PWM frequency. The PWM frequency and the divider ratios for fast and slow loops are user programmable in a wide range. The maximum values for PWM frequency and sampling rate frequencies depend on the application configuration. Table 2.4 shows the typical (default) values, which cover all motion application configurations.

Table 2.4.*Typical values for PWM frequency and sampling rates*

PWM	Fast loop (current/torque)	Slow loop (position/speed)
20kHz	10kHz	1kHz

3. Electrical and mechanical specifications

Detailed specifications will be available into the next revision of datasheet.

Table 3.1 describes the MotionChip^{TM} II – QFP100 pins.

Table 3.1. *MotionChip^{TM} II - QFP100 pins*

Pin Number	Pin name	Description
PWM		
39	PWM1	PWM1 command
37	$\overline{\text{PWM2}}$	PWM2 command
36	PWM3	PWM3 command
33	$\overline{\text{PWM4}}$	PWM4 command
31	PWM5	PWM5 command
28	$\overline{\text{PWM6}}$	PWM6 command
I/O's, ENCODERS, HALLS, DEDICATED INPUTS		
57	IN#3 / ENC1A	• Channel A signal from encoder 1 • General-Purpose Input
55	IN#4 / ENC1B	• Channel B signal from encoder 1 • General-Purpose Input
52	IN#5 / ENC1Z	• Channel Z signal from encoder 1 • General-Purpose Input
13	IO#13	General-Purpose I/O
11	IO#14	General-Purpose I/O
26	IO#15	General-Purpose I/O
60	IN#32 / ENC2A	• General-Purpose Input • Channel A signal from encoder 2 Must be connected together with IN#35 / HALL3. **The second encoder interface cannot be used simultaneously with digital Hall interface!**
56	IN#33 / ENC2B / HALL1	• General-Purpose Input • Channel B signal from encoder 2 • Hall 1 sensor Input **The second encoder interface cannot be used simultaneously with digital Hall interface!**
48	IN#34 / ENC2Z / HALL2	• General-Purpose Input • Channel Z signal from encoder 2 • Hall 2 sensor Input **The second encoder interface cannot be used simultaneously with digital Hall interface!**
7	IN#35 / HALL3	• General-Purpose Input • Hall 3 sensor Input. Must be connected together with IN#32 / ENC2A. **The digital Hall interface cannot be used simultaneously with second encoder interface!**

Pin Number	Pin name	Description
43	IO#27	General-Purpose I/O
41	IO#28	General-Purpose I/O
38	IO#29	General-Purpose I/O
32	IO#30	General-Purpose I/O
27	IO#31	General-Purpose I/O
5	IO#36	General-Purpose I/O
2	IO#37 / DIR	• Direction signal for P&D mode • General-Purpose I/O
89	IO#38 / PULSE	• Pulse signal for P&D mode • General-Purpose I/O
92	IO#39	General-purpose I/O
16	IO#2 / LSP	• Limit Switch - Positive • General-Purpose I/O
15	IO#24 / LSN	• Limit Switch - Negative • General-Purpose I/O
14	IN#16 / ENABLE	• ENABLE Input • General-Purpose Input
51	OUT#25 / $\overline{\text{READY}}$	Drive READY output signal
12	OUT#12 / $\overline{\text{ERROR}}$	ERROR output signal
93	$\overline{\text{RS}}$	Device Reset (in) and Watchdog Reset (out).
PROTECTION INPUTS		
6	$\overline{\text{PDPA}}$	Power drive protection input. When activated, puts the PWM output pins in the high-impedance state. Must be connected together with $\overline{\text{PDPB}}$.
95	$\overline{\text{PDPB}}$	Power drive protection input. When activated, puts the PWM output pins in the high-impedance state. Must be connected together with $\overline{\text{PDPA}}$.
EXTERNAL CONFIGURATION		
80	RS232/485	Select RS485 (0) / RS232 (1)
78	$\overline{\text{AUTO}}$	Select AUTORUN (0) / slave mode (1)
76	AxisID4	Axis ID 5^{th} bit
75	AxisID3	Axis ID 4^{th} bit
73	AxisID2	Axis ID 3^{rd} bit
71	AxisID1	Axis ID 2^{nd} bit
68	AxisID0	Axis ID 1^{st} bit
ANALOG INPUTS		
79	LH1	Linear Hall 1
77	DTEMP	Drive temperature
74	AD2	Not used, connect to the ground
72	REF	Analogue reference
70	VDC	Motor supply voltage
69	I_C	Current in phase C
67	I_A	Current in phase A

Pin Number	Pin name	Description
66	LH2	Linear Hall 2
65	LH3 / MTEMP	• Linear Hall 3 • Motor temperature
82	V_{REFHI}	ADC analog high-voltage reference input
81	V_{REFLO}	ADC analog low-voltage reference input
83	V_{CCA}	Analog supply voltage for ADC (3.3 V)
84	V_{SSA}	Analog ground reference for ADC
COMMUNICATION (SCI, SPI, CAN)		
49	CANRX	CAN receive data
50	CANTX	CAN transmit data
17	SCITX	SCI asynchronous serial port transmit data
18	SCIRX	SCI asynchronous serial port receive data
85	TxEn485	RS485 Enable transmission: "0" = receive, "1" = transmit
24	SPICLK	SPI clock
21	SPITX	SPI transmission
22	SPIRX	SPI reception
23	$\overline{\text{SPICS}}$	SPI chip select
OSCILLATOR, PLL AND MISCELLANEOUS		
87	XTAL1/CLKIN	PLL oscillator input pin. Crystal input to PLL/clock source input to PLL. XTAL1/CLKIN is tied to one side of a reference crystal.
88	XTAL2	Crystal output. PLL oscillator output pin.
10	PLLVCCA	PLL supply (3.3 V)
9	PLLF	PLL loop filter input 1
8	PLLF2	PLL loop filter input 2
40	VCCP (5V)	Connect to a +5V supply.
42	TP1	Test pin 1. *Do not connect.*
44	TP2	Test pin 2. *Do not connect.*
INTERNAL CONFIGURATION		
61	CFG0	Configuration pin 0. Connect to +3.3V.
62	CFG1	Configuration pin 1. Connect to +3.3V.
25	CFG2	Configuration pin 2. Connect to +3.3V.
100	CFG3	Configuration pin 3. Connect to ground.
94	CFG4	Configuration pin 4. Connect to ground.
96	CFG5	Configuration pin 5. Connect to ground.
1	CFG6	Configuration pin 6. Connect to ground.
99	CFG7	Configuration pin 7. Do not connect.
86	CFG8	Configuration pin 8. Connect to +3.3V with minimum of pull-up 1.6 Kohms.
45	CFG9	Configuration pin 9. Do not connect.

Pin Number	Pin name	Description
POWER SUPPLY		
20	VDD	Core supply +3.3 V. Digital logic supply voltage.
35		
59		
91		
4	VDDO	I/O buffer supply +3.3 V. Digital logic and buffer supply voltage.
30		
47		
54		
64		
98		
19	VSS	Core ground. Digital logic ground reference.
34		
58		
90		
3	VSSO	I/O buffer ground. Digital logic and buffer ground reference.
29		
46		
53		
63		
97		

4. MotionChip™ II I/O

4.1. Predefined Digital Outputs

4.1.1. PWM Outputs

The MotionChip II provides 6 PWM outputs, 3 active low ($\overline{\text{PWM2}}$, $\overline{\text{PWM4}}$ and $\overline{\text{PWM6}}$) and 3 active high (PWM1, PWM2 and PWM3).

In application configurations with three-phase motors (DC brushless and PMSM), each of the six PWM outputs drives one switching device of a three-phase inverter. PWM1, PWM3 and PWM5 drive the upper switching devices. PWM2, PWM4 and PWM6 drive the lower switching devices, see Figure 4.1.

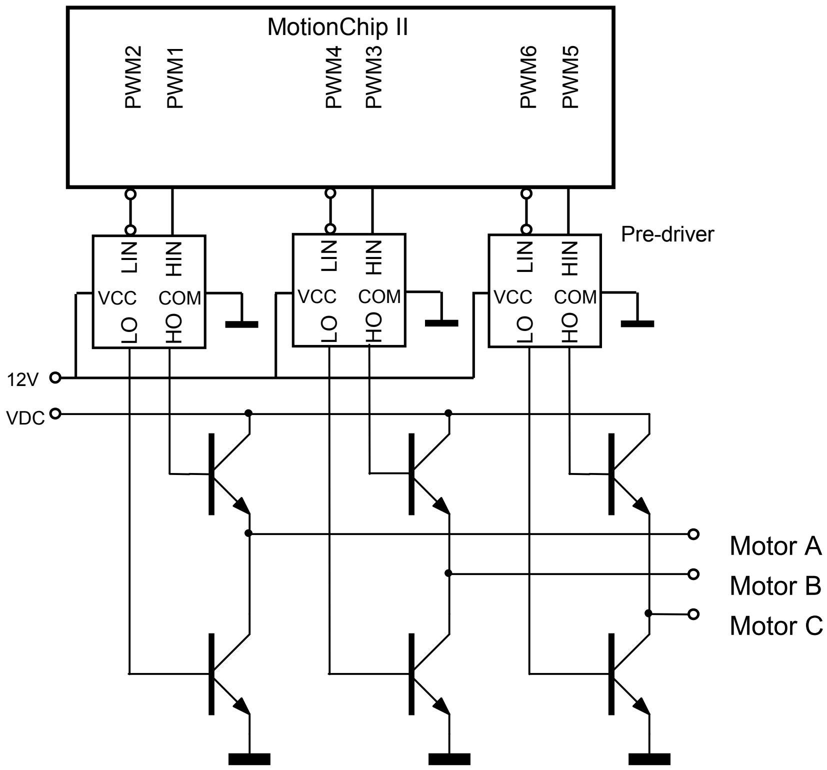


Figure 4.1. *PWM outputs with three phase inverter*

In application configurations with DC motors, only 4 PWM outputs are used, each drives one switching device of an H-bridge inverter. PWM1 and PWM3 drive the upper switching devices. PWM2 and PWM4 drive the lower switching devices. PWM1 & $\overline{\text{PWM2}}$ drive one bridge leg, PWM3 & $\overline{\text{PWM4}}$ drive the other leg (see Figure 4.2). PWM5 & $\overline{\text{PWM6}}$ are not used and are set high (inactive); they may be left unconnected for application configurations with DC motors.

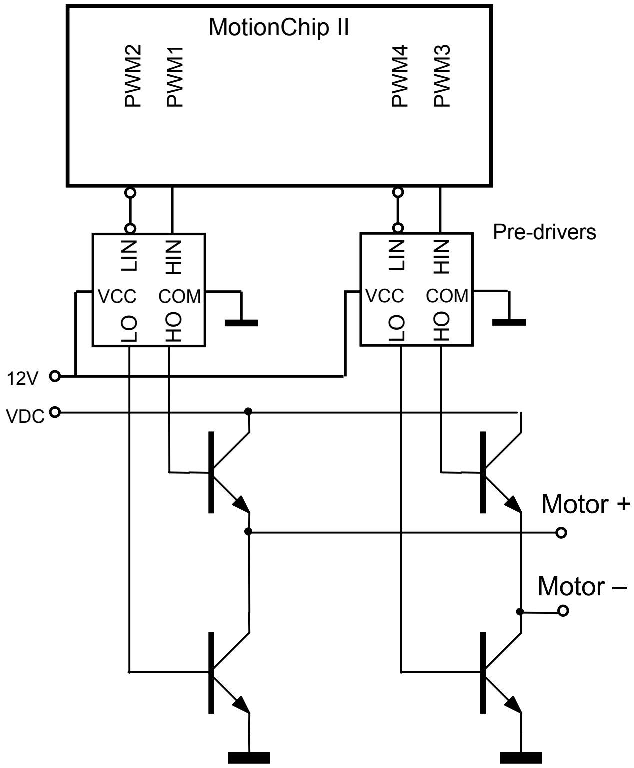


Figure 4.2. *PWM outputs usage with H bridges (two-phase inverters)*

At power-on or after any MotionChip II reset, all PWM outputs remain in the high-Z state until TML command `AXISON` is first executed. PWM outputs go immediately to the high-impedance state in one of the following cases:

- when $\overline{\text{PDPINT}}$ input goes low (high to low transition)
- when ENABLE input goes low (high to low transition) and as long as ENABLE remains low
- when TML command `AXISOFF` is executed;

The MotionChip II provides a programmable dead-band for each PWM output pair (1-2, 3-4, 5-6) from 0 to 12µs. The default value is 0µs. Hence you must set the TML parameter DBT with a nonzero value or use external dead-band logic in order to drive the power stage of a three-phase motor or that of a DC brush motor i.e. the configurations where each PWM output drives one switching device. For more details about power converter setup see MotionChip II Configuration Setup User Manual.

4.1.2. BRAKE Output

The MotionChip II can monitor the DC-bus voltage level if a voltage feedback is provided on the analogue input VDC. In some cases, during motor brakes, the DC-bus voltage may grow to dangerous values due to the current injected by the motor into the DC-link. One solution to reduce this over-voltage is to connect temporary a resistor across the DC-link through a switching device (see Figure 4.3). The MotionChip II has an active-high PWM output called BRAKE dedicated to drive the switching device from a brake circuit. When this function is enabled, the BRAKE output is automatically activated with a 50% duty cycle, if the DC-bus voltage bypasses a programmed trigger value. The BRAKE output remains active until the DC-bus voltage drops under this trigger value. The activation of the BRAKE command can be signaled by a TML interrupt (see MotionChip™ II TML Programming User Manual) if the over-voltage protection limit is set at the same value as the BRAKE trigger level.

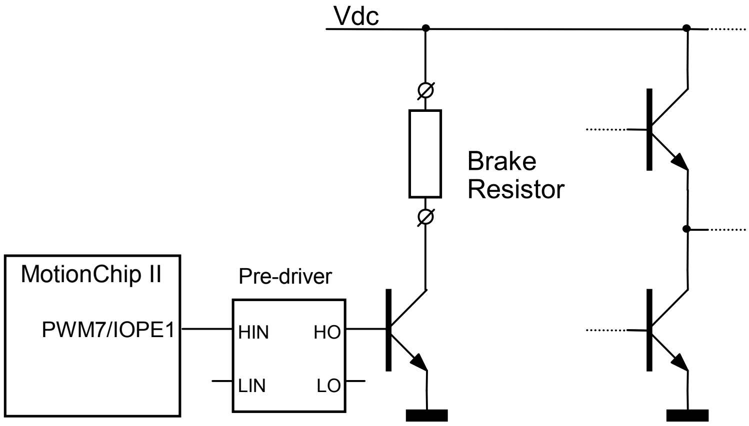


Figure 4.3. *BRAKE pin usage*

4.1.3. READY and ERROR Outputs

The OUT#25 / READY output is set high immediately after reset. It goes low after the MotionChip II internal initialization ends. This output can be used to signal to an external device that the MotionChip is powered and is ready to receive and execute TML commands.

The OUT#12 / ERROR output is set high at firmware initialization and it is set low in the TML interrupts when a protection is triggered and the interrupts are enabled.

4.2. Predefined Digital Inputs

4.2.1. PDPA, PDPB – Power Drive Protection Inputs

$\overline{\text{PDPA}}$ and $\overline{\text{PDPB}}$ are two protection inputs that immediately shuts-down (deactivates) the PWM outputs when high to low transition occurs on this pin. The goal of this input is to offer quick reacting protection to the power stage if at this level an error occurs. When the falling edge is sensed, all 6 PWM outputs plus the BRAKE signal (if activated) are set to High-Z state.

4.2.2. LSP, LSN - Limit Switch Inputs

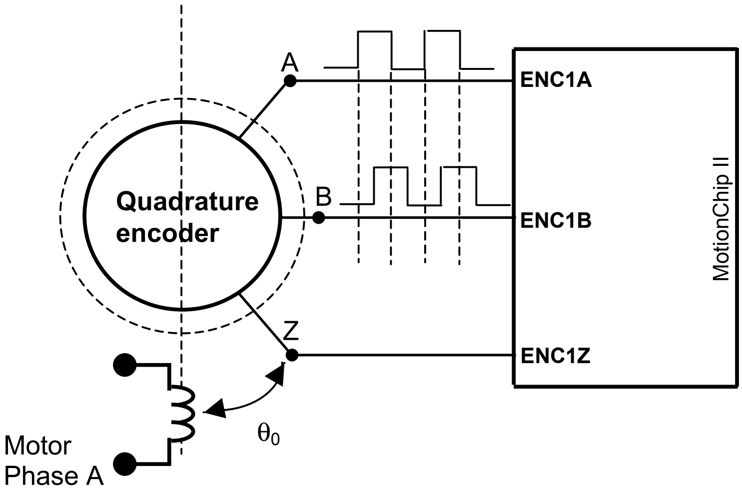
The MotionChip II has two dedicated interrupt inputs for limit switches: Limit Switch Positive (IO#24 / LSP) and Limit Switch Negative (IO#2 / LSN). The inputs can be programmed to detect transitions from low to high or high to low. The following actions can be programmed when a transition is detected on a limit switch input:

- Generate a TML interrupt. The TML interrupt service routine accepts any TML command, hence when a limit switch is reached, any desired action can be programmed;
- Generate a TML event. On event occurrence, the motion mode and/or parameters can be automatically changed. This feature is useful if the motor is on-purpose moved towards a limit switch, with a goal to execute a specific motion sequence when the limit switch is reached.

4.2.3. Quadrature Encoder Inputs

The MotionChip II includes two incremental encoder interfaces. The first interface is used to read motor position (ENC1A, ENC1B, ENC1Z), and the second interface (ENC2A, ENC2B, ENC2Z) can be used to read master position from master's second encoder.

If the encoder quadrature signals A and B are CMOS level compatible, they can be connected directly to the MotionChip II inputs ENC1A and ENC1B; else the signals level must be adapted. The position is incremented or decremented by the rising or falling edge of the two input signals (four times the frequency of either input pulse). The minimum pulse length on any encoder input is 112ns.



Remark *The second encoder inputs are used also for Hall sensors inputs, so the use of one type of sensor excludes the other one.*

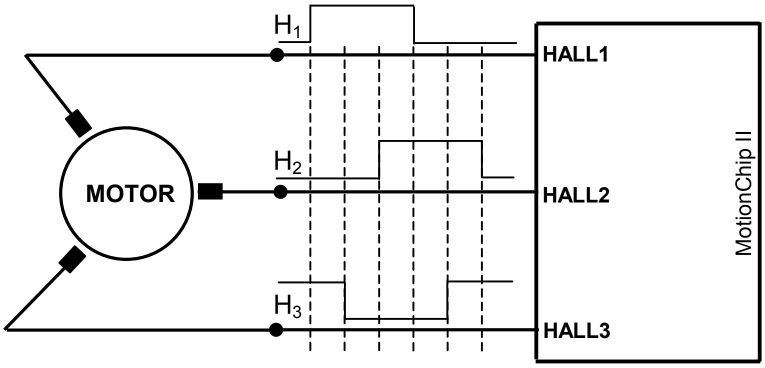
The second encoder can't be used with DC brushless motors, since they require feedback from Hall sensors.

Digital inputs ENC1Z and ENC2Z can be used as capture inputs they can be programmed to sense either a low to high or high to low transition. When the programmed transition occurs, the motor position or the master position is captured e.g. memorized in dedicated variables. These are CAPPOS for the motor position and CAPPOS2 for the master position. The captured position is very accurate as the whole process is very fast (less than 200 ns if the position sensor is an encoder).

4.2.4. HALL Sensors Inputs

The Hall sensors signals coming from a brushless DC motor can be directly connected to MotionChip II HALL1, HALL2 and HALL3 inputs if the signals are CMOS level compatible; else the signals level must be adapted.

The MotionChip II can work with Hall signals phased at 120 electrical degrees. In the relation between the voltage succession and Hall sensors values, there are 12 possible cases. The MotionChip II supports all, hence there is no restriction concerning the Hall sensors sequence or polarity. In application that doesn't require Hall sensors feedback this inputs can be used to read master position from the second quadrature encoder.



Remark: *In a motion application based on a brushless DC motor, these pins will be automatically set as inputs to read Hall signals, when `ENDINIT` TML instruction is executed.*

The second encoder can not be used in motion application with brushless DC motor because uses the same pins as Hall sensors.

4.2.5. ENABLE Input

IN#16 / ENABLE input is an external emergency stop. When asserted high, the MotionChip II executes the TML command `AXISOFF`, which has the following consequences: the controllers are disabled, the PWM outputs go to high-impedance state and a TML interrupt can be generated. The ENABLE input acts like a non-maskable interrupt, which is always executed, no matter of the MotionChip II operation context. As long as this pin is held high, the `AXISON` command has no effect. When the ENABLE input is asserted low MotionChip II operates normally.

4.3. General Purpose digital I/O

The MotionChip II has 21 digital inputs/outputs pins. Some of these I/O pins have also second shared function (see Table 4.1). These pins can be used as I/O only if the shared function is disabled. For example IO#33 / ENC2B / HALL1, IO#34 / ENC2Z / HALL2 and IO#35 / HALL3 may be used as general-purpose inputs in motion application configuration where Hall sensors feedback is not required. In a motion application based on a brushless DC motor, these pins will be automatically set as inputs to read Hall signals, when `ENDINIT` TML instruction is executed and cannot be used as general-purpose inputs.

Table 4.1. *MotionChip™ II IO's*

TML Name	Default State	Type	Description
IO#2 / LSP	LSP	I/O	• Limit Switch - Positive • General-Purpose I/O
IN#3 / ENC1A	ENC1A	I	• Channel A signal from encoder 1 • General-Purpose Input
IN#4 / ENC1B	ENC1B	I	• Channel B signal from encoder 1 • General-Purpose Input
IN#5 / ENC1Z	ENC1Z	I	• Channel Z signal from encoder 1 • General-Purpose Input
IO#13	Out high	I/O	General purpose I/O
IO#14	Out high	I/O	General purpose I/O
IO#15	Out high	I/O	General purpose I/O
IO#24 / LSN	LSN	I/O	• Limit Switch - Negative • General-Purpose I/O
IO#27	Out high	I/O	General purpose I/O
IO#28	Out high	I/O	General purpose I/O
IO#29	Out high	I/O	General purpose I/O
IO#30	Out high	I/O	General purpose I/O
IO#31	Out high	I/O	General purpose I/O
IN#32 / ENC2A	ENC2A	I	• General-Purpose Input • Channel A signal from encoder 2
IN#33 / ENC2B / HALL1	ENC2B	I	• General-Purpose Input • Channel B signal from encoder 2 • Hall 1 sensor Input
IN#34 / ENC2Z / HALL2	ENC2Z	I	• General-Purpose Input • Channel Z signal from encoder 2 • Hall 2 sensor Input
IN#35 / HALL3	In	I	• General-Purpose Input • Hall 3 sensor Input
IO#36	In	I/O	General purpose I/O
IO#37 / DIR	In	I/O	• Direction signal for P&D mode • General-Purpose I/O
IO#38 / PULSE	In	I/O	• Pulse signal for P&D mode • General-Purpose I/O
IO#39	In	I/O	General purpose I/O

4.4. Analogue Inputs

The MotionChip II has 9 analogue inputs to measure currents, motor and drive temperatures, DC-bus voltage, analogue external reference and linear Hall sensors.

The A/D channels functions is described in Table 4.2.

Table 4.2. *MotionChip^{TM} II analogue inputs*

PIN Name	Type	Description
LH1	I	Read the Linear Hall 1
DTEMP	I	Used to measure drive temperature
AD2	I	Analogue Feedback input
REF	I	Analogue Reference input
VDC	I	Used to measure DC-bus voltage
I_C	I	Measure the current through motor phase C
I_A	I	Measure the current through motor phase A
LH2	I	Read the Linear Hall 1
LH3 / MTEMP	I	Read the Linear Hall 3 / Used to measure motor temperature

4.4.1. IA, IC – Motor Currents

For torque/current control, the MotionChip II needs current feedback. In case of three-phase motors, the MotionChip II uses for torque control two currents, the motor current in phase A, read on input I_A, and the motor current in phase C read on I_C input. In case of DC brush motors, the MotionChip II reads the motor current on I_A input.

The current sensors may be placed on the motor lines or on the lower legs of the inverter. Figure 4.4 shows how to measure the currents for DC brush or three-phase motors using shunts on the lower legs of the inverter. Current feedback signal polarity relative to motor current is emphasized.

By default, the values measured on I_A and I_C inputs are interpreted with an offset of half A/D input range. Hence, values higher than mid-point are interpreted as positive currents, while values lower than mid-point are interpreted as negative values. Due to current measurement scheme imperfections, sometimes the A/D mid-point doesn't corresponding exactly to a zero current. For these situations, the MotionChip II can detect automatically the true offset, for each of the A/D inputs. This offset will be considered for current feedback interpretation.

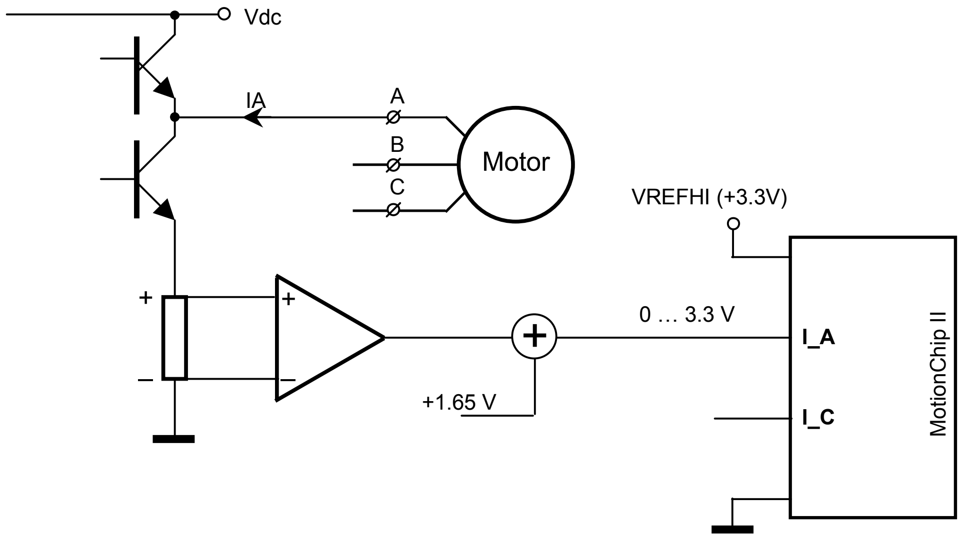


Figure 4.4. *Current measurement from lower-legs of a 2 or 3-phase converter*

4.4.2. Linear Hall Inputs

The MotionChip II can use as position feedback the signals of 3 Linear Hall sensors, read on inputs LH1, LH2 and LH3/MTEMP. The Linear Hall signals must have the same amplitude and be phased at 120 electrical degrees. The sensors signals must be adapted to a range smaller, or at most equal with the A/D input range, i.e. 0-3.3 V, and **never** exceed this range.

The MotionChip II interprets the Linear Hall signals using offsets and gains determined experimentally. Due to this approach, the MotionChip II is independent of the measurement scheme. Also, the imperfections of the measurement scheme and/or of the Linear Hall sensors are eliminated; sometimes the signals' amplitudes aren't equal. The gains and the offsets are determined using a specific test.

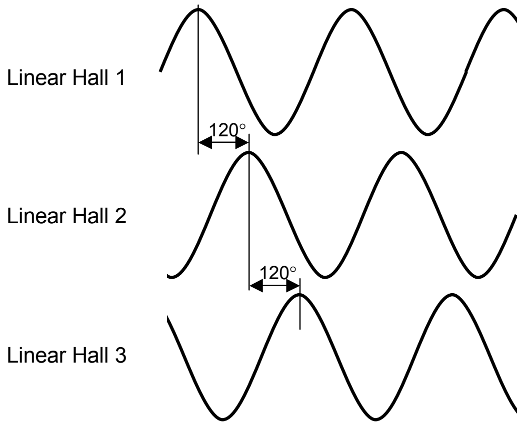


Figure 4.5. *Linear Hall waveforms*

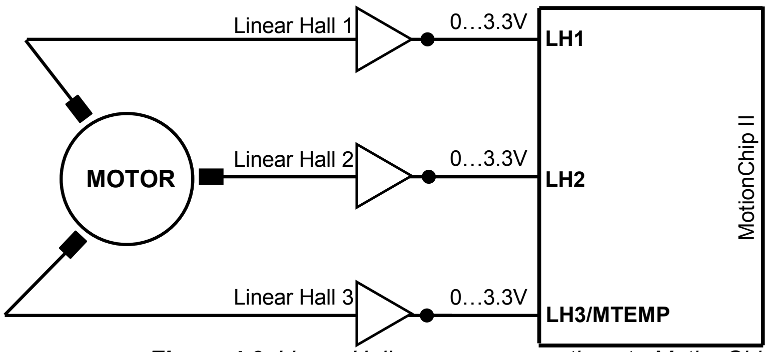


Figure 4.6. *Linear Hall sensors connections to MotionChip II*

Remarks:

1. *In configurations with position feedback from linear Hall sensors, the motor temperature monitoring is not available, since the AD input for monitoring the motor temperature is used to read the third linear Hall signal.*
2. *Add a low pass filter to remove the noise from the linear Hall signals. The cutoff frequency of the filter should take in account the maximum velocity of the application, i.e. the cutoff frequency should be greater than the frequency of linear Hall signals at maximum velocity.*

4.4.3. VDC – DC-bus Voltage

The MotionChip II has a dedicated analogue input - VDC for DC-bus voltage measure. When this feedback is provided, the following features of the MotionChip II can be enabled:

- Compensation of the DC-bus voltage variation;
- Brake control
- Over-voltage and under-voltage protections

Since DC-link/supply voltage can have only positive values, the value read on VDC input is interpreted without any offset, the DC-link/supply maximum voltage correspond to analogue input maximum voltage (i.e. 3.3V). Figure 4.7 presents an example how to connect the voltage sensor at MotionChip II analogue input.

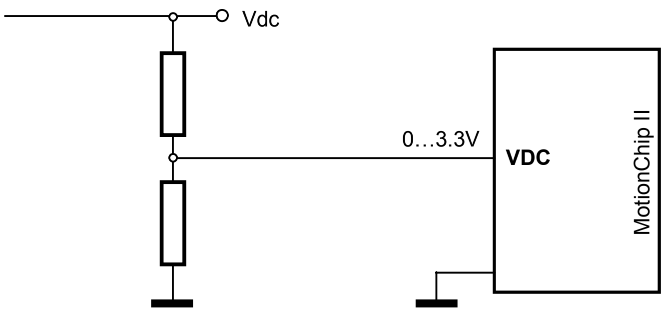


Figure 4.7. *DC-link/supply voltage measurement*

4.4.4. REF – Analogue Reference

In motion modes with analogue external reference, when internal trajectory generator is disabled, the reference is provided on the analogue input REF. For example, a potentiometer can be connected on this input in order to set the speed reference of a motor. By default, the values measured on REF input are interpreted with an offset of half A/D input range. For applications where the analogue reference can have only positive or only negative values, the default offset can be changed in order to accommodate with these cases.

Figure 4.8 presents an example how to apply an analogue external reference to MotionChip II.

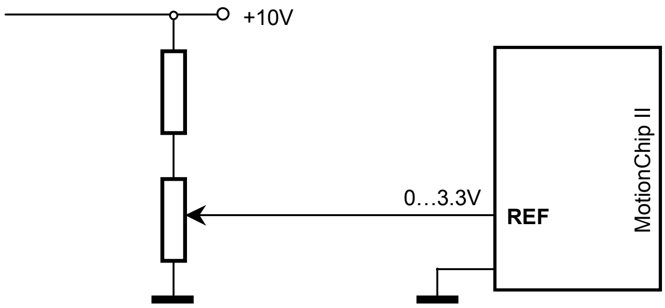


Figure 4.8. *Analogue external reference*

4.4.5. MTEMP, DTEMP – Temperature sensors

The MotionChip II includes two analogue inputs MTEMP to monitor the motor temperature and DTEMP to monitor the power stage temperature. The values measured from temperature sensors are used by the software protections. The protections can be programmed to signal when the values read from MTEMP, DTEMP inputs bypass certain programmed limits.

The MTEMP and DTEMP inputs can have only positive values. Hence, the values read are interpreted without any A/D offset.

Motor temperature sensor can be PTC or NTC. For power stage temperature sensor the user can specify the sensor offset (sensor output at 0°C), which is added to the measured value.

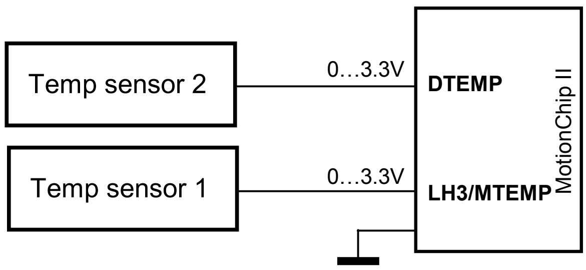


Figure 4.9. *Temperature sensors connected to MotionChip II*

Remark: *In configurations that use linear Hall sensors, the analogue input for motor temperature sensor is used to read the third linear Hall sensor. Hence, in such configurations the motor temperature monitoring is not available.*

4.5. External Configuration Inputs

The external configuration pins are described in Table 4.3.

Table 4.3. *MotionChip™ II external configuration inputs*

PIN Name	Type	Description
RS232/485	I	Select serial communication protocol RS232 or RS485
Auto	I	Enable/Disable Autorun mode
AxisID4	I	5th AxisID bit
AxisID3	I	4th AxisID bit
AxisID2	I	3rd AxisID bit
AxisID1	I	2nd AxisID bit
AxisID0	I	1st AxisID bit

- The RS232/485 channel allows selecting, at system initialization (at power on or after a reset), the serial communication protocol RS-232 or RS-485. To select the RS-232 communication protocol the pin must be connected to high; For the RS-485 communication protocol the pin must be connected to low.
- The Auto channel allows enabling/disabling, at system initialization, the AUTORUN mode. To enable the AUTORUN mode the pin must be connected to low; the AUTORUN mode is disabled (MotionChip II waits for commands) when the pin must be connected to high.
- In multiple-axis configurations, each MotionChip II based drive needs to be identified through a unique number – the **Axis ID**. This is a number between 1 and 255. The axis ID is initially set at power on by reading the MotionChip II input lines AxisID0 to AxisID4, as the Table 4.4 shows:

Table 4.4. *Axis ID / Address configuration*

AxisID4	AxisID3	AxisID2	AxisID1	AxisID0	Axis ID
HIGH	HIGH	HIGH	HIGH	HIGH	255
HIGH	HIGH	HIGH	HIGH	LOW	1
HIGH	HIGH	HIGH	LOW	HIGH	2
HIGH	HIGH	HIGH	LOW	LOW	3
HIGH	HIGH	LOW	HIGH	HIGH	4
HIGH	HIGH	LOW	HIGH	LOW	5
HIGH	HIGH	LOW	LOW	HIGH	6
HIGH	HIGH	LOW	LOW	LOW	7
HIGH	LOW	HIGH	HIGH	HIGH	8
HIGH	LOW	HIGH	HIGH	LOW	9
HIGH	LOW	HIGH	LOW	HIGH	10
HIGH	LOW	HIGH	LOW	LOW	11
HIGH	LOW	LOW	HIGH	HIGH	12
HIGH	LOW	LOW	HIGH	LOW	13
HIGH	LOW	LOW	LOW	HIGH	14
HIGH	LOW	LOW	LOW	LOW	15
LOW	HIGH	HIGH	HIGH	HIGH	16
LOW	HIGH	HIGH	HIGH	LOW	17
LOW	HIGH	HIGH	LOW	HIGH	18
LOW	HIGH	HIGH	LOW	LOW	19
LOW	HIGH	LOW	HIGH	HIGH	20

AxisID4	AxisID3	AxisID2	AxisID1	AxisID0	Axis ID
LOW	HIGH	LOW	HIGH	LOW	21
LOW	HIGH	LOW	LOW	HIGH	22
LOW	HIGH	LOW	LOW	LOW	23
LOW	LOW	HIGH	HIGH	HIGH	24
LOW	LOW	HIGH	HIGH	LOW	25
LOW	LOW	HIGH	LOW	HIGH	26
LOW	LOW	HIGH	LOW	LOW	27
LOW	LOW	LOW	HIGH	HIGH	28
LOW	LOW	LOW	HIGH	LOW	29
LOW	LOW	LOW	LOW	HIGH	30
LOW	LOW	LOW	LOW	LOW	31

Note1: Other Axis ID values (32 to 255) can be set by software, with the *AXISID* instruction.

Figure 4.10 presents an example how to connect the external configuration inputs for setting AXISID using a DIP switch.

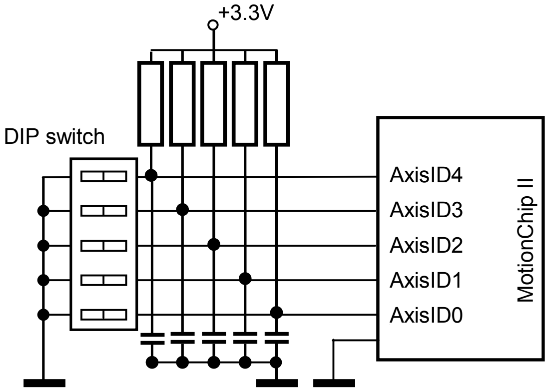


Figure 4.10. *AxisID selection using a DIP switch*

5. MotionChip™ II Memory

The MotionChip II works with 2 separate address spaces: one for TML programs and the other for data memory. Each space accommodates a total of 64K 16-bit word.

The first 16K of the TML program space (0 to 3FFFh) is reserved and can't be used. The next 16K, from 4000h up to 7FFFh (for 256kbits E^2ROM) is mapped to a serial SPI-connected E^2ROM. This space can be used to store TML programs, cam tables or other user data in a non-volatile memory. The E^2ROM size can be: 64kbits, 128kbits or 256kbits.

The SPI clock rates can be set at 1MHz, 2MHz or 5MHz. The default rate after reset is 1MHz.

Figure 5.1 presents how to interface a serial SPI - E^2ROM with MotionChip II.

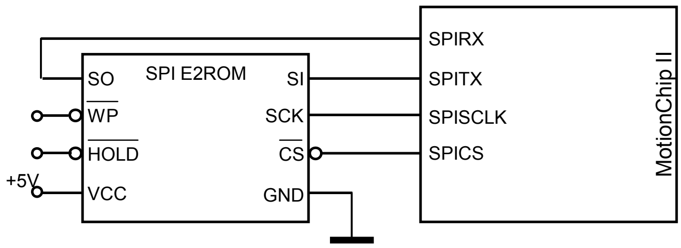


Figure 5.1. *MotionChip™ II connections with E^2ROM*

The data memory space is used to store the TML data (registers, parameters, variables), the cam tables during runtime (after being copied from the E^2ROM memory) and for data acquisitions. The TML data are stored in reserved area, while the others are using the same MotionChip II internal SRAM memory.

The SRAM memory from 8280h to 87FFh may be used to temporary store TML programs. This memory is shared with data memory from A80h to FFFh.

The SRAM memory from A80h to FFFh (corresponding to 8280h to 87FFh in the TML program memory space) may be used for data acquisitions and/or to store cam tables during runtime. As this space is available in both the TML program space and the data space it is the user responsibility to decide how to split it between the two and to avoid overlapping them.

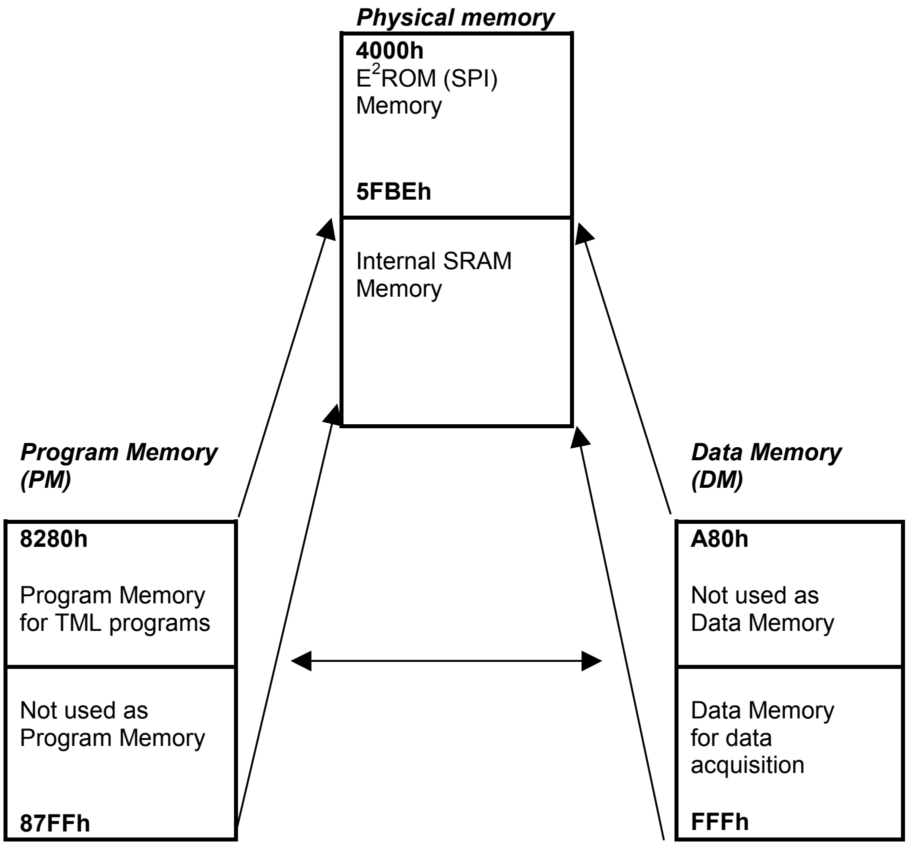


Figure 5.2. *MotionChip™ II memory map*

6. MotionChip™ II Communication

6.1. Communication channels

The MotionChip II accepts two types of communication channels:

- Serial RS-232 or RS-485 with on-chip controller
- CAN-bus with on-chip CAN controller

The serial RS-232 communication channel can be used to connect a host with a single MotionChip II based device (see Figure 6.1). The serial RS-485 and the CAN-bus communication channels can be used to create a distributed control network with a host and up to 255 MotionChip II based devices (see Figure 6.2 and Figure 6.3).

When CAN-bus communication is used, any MotionChip II based device from the network may also be connected through RS-232 with a host (see Figure 6.4). In this structure, the axis connected to the host, apart from executing the commands received from host or other axes acts also as a retransmission relay which:

- Receives through RS-232, commands from host for another axis and retransmits them to the destination through CAN-bus
- Receives through CAN-bus data requested by host from another axis and retransmits them to the host through RS-232

This flexibility enables a host to program and monitor a CAN-bus network of devices based on the MotionChip II, using only one RS-232 connection, without the need to have a CAN-bus interface. In this case the CAN-bus protocol is completely transparent for the host.

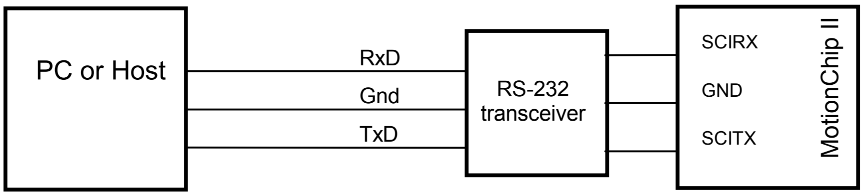


Figure 6.1. *Serial RS-232 communication between a host and the MotionChip II*

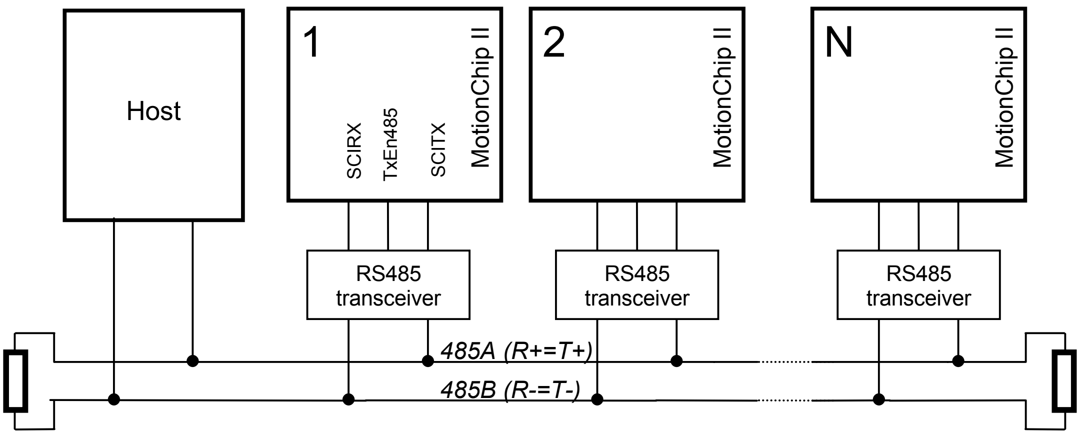


Figure 6.2.*Multi-drop network using serial RS-485 communication*

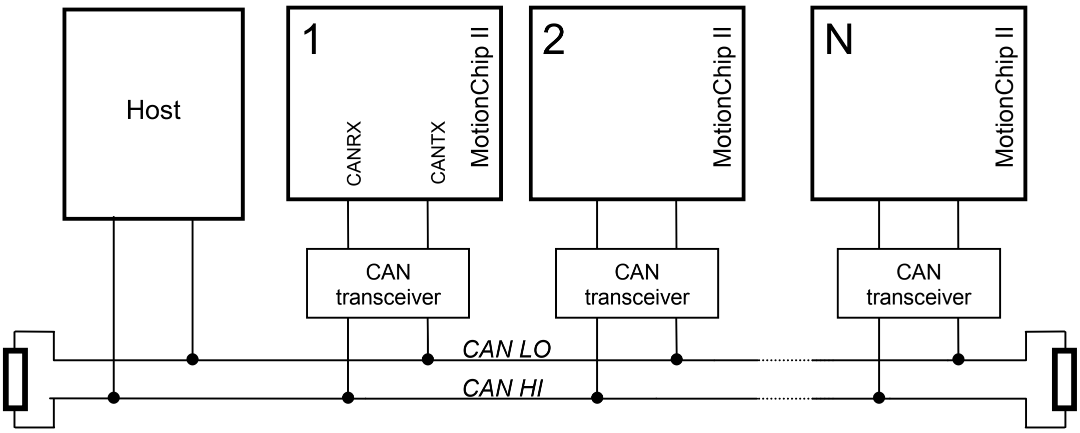


Figure 6.3. *Multi-drop network using CAN-bus communication*

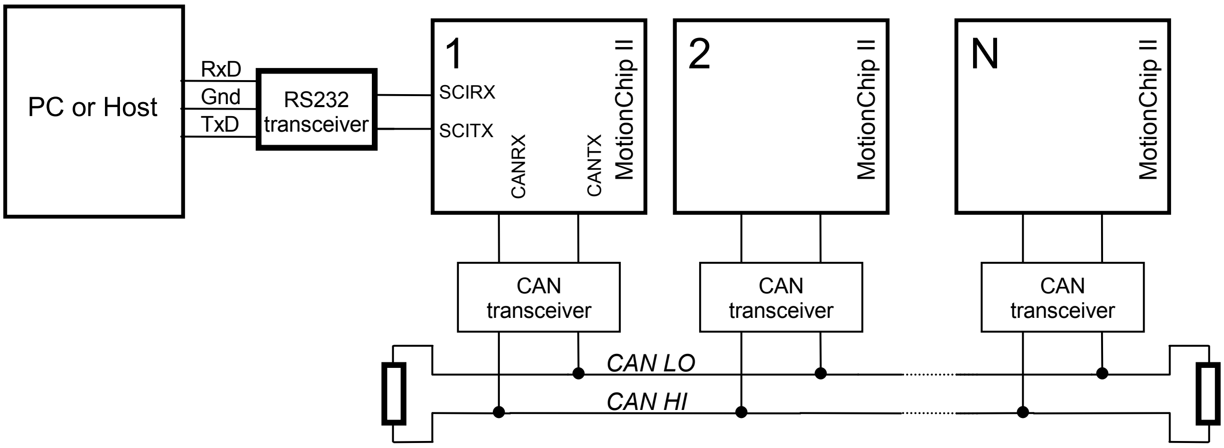


Figure 6.4.*Multi-drop network using CAN-bus communication with host connected through RS-232 to an axis used as communication relay*

6.2. Serial RS-232 and RS485 Interface

The serial communication is based on the on-chip serial communication interface (SCI). This interface provides two pins: SCITX for transmission and SCIRX for reception. The SCI interface may be used for serial RS-232 or RS-485 communication.

For RS-232, the SCIRX and SCITX pins should be directly linked to an RS-232 transceiver (see Figure 6.5).

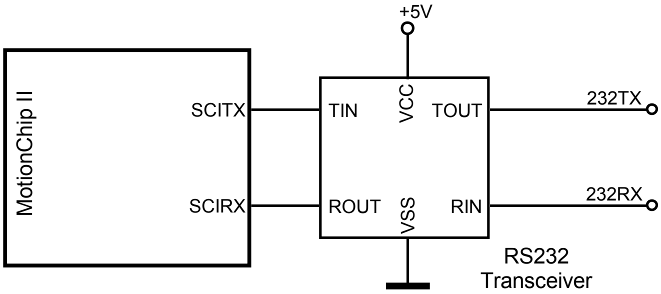


Figure 6.5. *Interface with an RS-232 transceiver*

The communication on RS-485 is enabled when the voltage applied at the input RS232/485 is lower then 1.65V. For RS-485, the two SCI pins and the RS-485 transmission/reception enable pin TxEN485, must be connected to an RS-485 transceiver.

The TxEN485 pin is used only when RS-485 communication is selected. The TxEN485 pin is kept low unless asserted high during transmission cycles. Hence, when TxEN485 is asserted low, the reception must be enabled and the transmission disabled. When TxEN485 is asserted high, the transmission must be enabled and the reception disabled.

The Figure 6.6 shows an example how to interface the MotionChip II with an RS-485 transceiver.

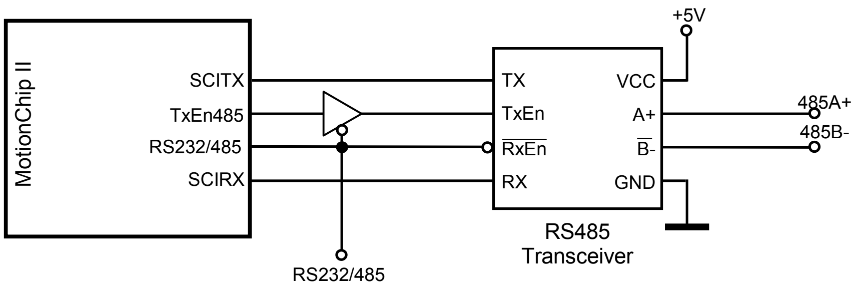


Figure 6.6. *Interface with an RS-485 transceiver*

6.3. CAN-bus Interface

The CAN-bus communication is based on the on-chip CAN controller. The CAN controller requires an external transceiver. Figure 6.7 presents the MotionChip II interface with the CAN transceiver. By default, the CAN-bus baud-rate is set at 500kb.

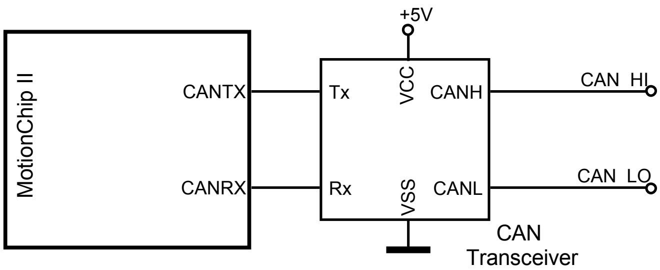


Figure 6.7. *Interface with a CAN transceiver*

This page is empty

7. Development Tools

A complete range of development tools is available for the MotionChip II user. It starts with a basic evaluation kit, the MotionChip™ II Starter Kit (**MCIISK**), accompanied by power module PM-50. On the software side, the powerful **IPM Motion Studio** platform allows you to configure, parameterize and program the motion for your application.

The MotionChip™ II Starter Kit boards contains the MotionChip II, external E^2ROM and RAM memories, an RS-232 interface, and extension connectors, to interface the MotionChip II with external power modules and/or external I/O interfaces. The MCIISK kit can be combined with specific power amplifier modules and motors, thus becoming a complete motion structure for the evaluation of motion applications based on the MotionChip II. See the "**MotionChip™ II Starter Kit User Manual**" for more details about MCIISK kit contents and features.

With the MCIISK board you receive a template for IPM Motion Studio, which is a high level graphical Windows environment for MotionChip II applications development. It allows you to configure and parameterize a motion system (including tuning and auto tuning of controllers) and to define motion sequences using high-level integrated tools, which automatically generate Technosoft Motion Language source code (TML instructions). Embedded code development tools allow you to further edit or directly compile, link and generate executable code to be downloaded to the MotionChip II. Finally, advanced graphics tools — like data logger, control panel and view/watch of TML parameters, registers and memory — can be used to analyze the behavior of the motion system. See the "**IPM Motion Studio User Manual**" for more details about IPM Motion Studio contents and features.

This page is empty

T E C H N O S O F T