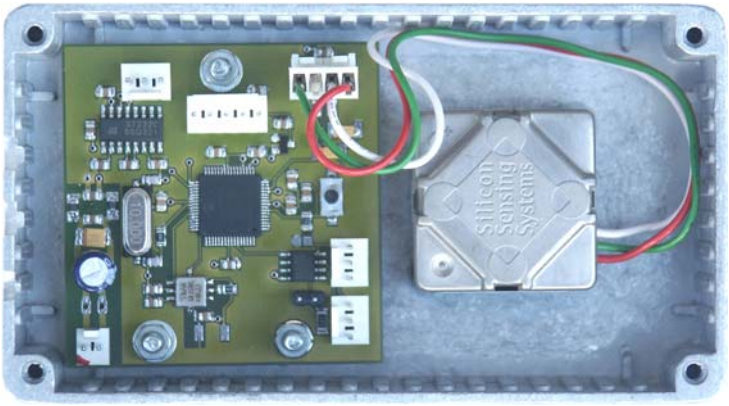
NEOBOTIX

Neobotix GyroBoard

Technical Description



author: Dr.-Ing. Oliver Barth

date: 03.2005

version: 1.02

Contents

1 Introduction

The GyroBoard is designed for sensor data acquisition and evaluation. It provides an on-board accelerometer, an on-board temperature sensor and an analog interface for an external gyro.

1.1 Technical Data GyroBoard

- 1 x analog input for gyro, voltage range 0 V – 5 V
- 1 x accelerometer, 2 axis measurement
- 1 x temperature sensor
- Single supply voltage +5V, 200 mA
- Communication interfaces CAN or RS-232

2 Software

The software is running on a 16 bit/ 20 MHz CPU. It provides access to all board functionality.

The board may be used by RS-232 interface or CAN bus or both at the same time.

The baud rate values are fixed:

CAN	RS 232
500.000 Baud	19.200 Baud

If other baud rates are requested, please contact Neobotix.

The CAN standard receive identifier is 0x121, the standard send identifier is 0x120. If other identifiers are requested, please contact Neobotix.

3 CAN communication protocol

The communication is based on 8 byte CAN messages, so the data length code of send and receive messages is 8 byte always.

Command values in CAN messages are taken from an enumeration named CmdGyroBoard:

```
enum CmdGyroBoard
{
	CMD_GYROBOARD_CONNECT,
	CMD_GYROBOARD_DISCONNECT,
	CMD_GYROBOARD_ZEROGYRO,
	CMD_GYROBOARD_GETGYRO,
	CMD_GYROBOARD_GETDATA,
	CMD_GYROBOARD_DEBUG,
	CMD_GYROBOARD_CMD1,
	CMD_GYROBOARD_CMD2,
	CMD_GYROBOARD_CMD3,
	CMD_GYROBOARD_CMD4,
	CMD_GYROBOARD_UNKNOWN
};
```

To evaluate received messages the command is returned in byte 8 of the message left shift by two. The first two bits of byte 8 are used for returning if the command was executed successfully or not. The enumeration *MsgState* defines return codes as follows:

```
enum MsgState
{
	MSG_OK,
	MSG_ERROR,
	MSG_NOT_ACCEPT,
	MSG_NO_ACTION
};
```


3.1 CAN Message description

Command	**CMD_GYROBOARD_CONNECT**
Purpose	Connect to the microcontroller, switches the internal state machine to connected
Message format	(CMD_GYROBOARD_CONNECT, 0, 0, 0, 0, 0, 0, 0)
Answer	(0x01, 0x02, 0x03, 0x04, 0x05, 0x06, 0x07, MSG_OK | (CMD_GYROBOARD_CONNECT << 2) - numbers in return message are for testing purposes only

Command	**CMD_GYROBOARD_DISCONNECT**
Purpose	Disconnect the microcontroller, switches the internal state machine to disconnected
Message format	(CMD_GYROBOARD_DISCONNECT, 0, 0, 0, 0, 0, 0, 0)
Answer	No answer

Command	**CMD_GYROBOARD_ZEROGYRO**
Purpose	Set the gyro to calibration mode. This means that the current gyro value is sampled, averaged and stored. This mode must only be used if the gyro is not rotated. During calibration a gyro value request will return zero.
Message format	(CMD_GYROBOARD_ZEROGYRO, Para1, 0, 0, 0, 0, 0, 0) iPara1: 1 = calibrate gyro, 0 = normal operation
Answer	No answer

Command	**CMD_GYROBOARD_GETGYRO**
Purpose	Return the current gyro value as signed long integer (32 bit).
Message format	(CMD_GYROBOARD_GETGYRO, 0, 0, 0, 0, 0, 0, 0)
Answer	(lGyro >> 24, lGyro >> 16, lGyro >> 8, lGyro, 0, 0, 0, MSG_OK | (CMD_GYROBOARD_GETGYRO << 2)) - lGyro: current gyro value, units processor internal, factory calibrated to +25°, calibration factor stored in C++ driver class delivered with Gyroboard

Command	**CMD_GYROBOARD_GETDATA**
Purpose	Return accelerometer data as singed integer (16 bit) and temperature data as signed integer (16 bit).
Message format	(CMD_GYROBOARD_GETGYRO, 0, 0, 0, 0, 0, 0, 0)
Answer	(iAcc1 >> 8, iAcc1, iAcc2 >> 8, iAcc2, iTemp >> 8, iTemp, 0, MSG_OK | (CMD_GYROBOARD_GETDATA << 2)) - iAcc1: acceleration axis1, units m/s² x 100 - iAcc2: acceleration axis2, units m/s² x 100 - iTemp: temperature, units degree

4 RS-232 communication protocol

The RS-232 protocol is identical to the CAN protocol. Send 8 byte formatted as described in chapter 3.1 without any delimiter like LF or CR.

5 GyroBoard States

The GyroBoard has two states

- *State Idle*
- *State Connected*

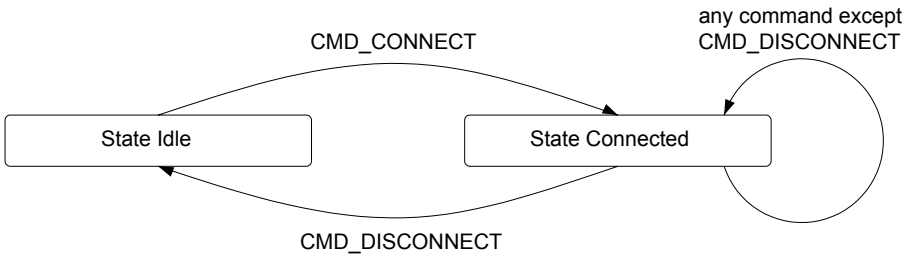


Figure 5-1: States of GyroBoard

Only in *State Connected* sensors can be activated/ deactivated or sensor values can be requested.

6 Picture of GyroBoard

Figure 6-1 shows a picture of GyroBoard. For pin assignment, please see chapter 7.

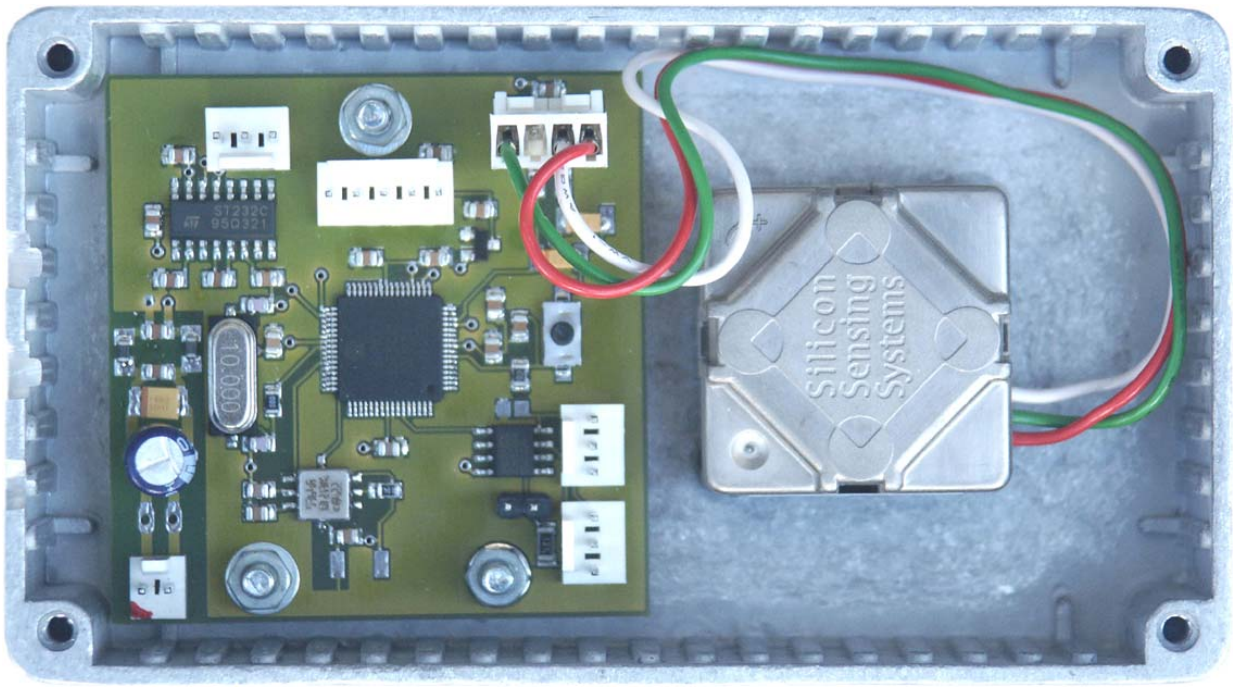


Figure 6-1: picture of GyroBoard

7 Connector assignment

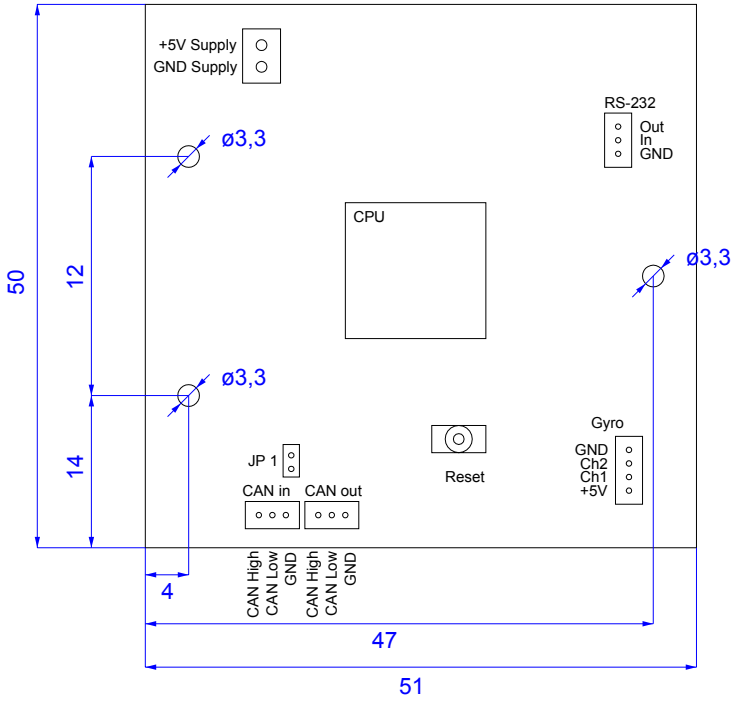


Figure 7-1: Connector assignment of GyroBoard

Note: Connect JP1 for activation of CAN terminal resistor (120 Ohm).

8 Appendix A: Parts

For all wire-to-board connectors type and suppliers are listed below.
(RS = RS-Components (www.rs-components.de). An alternative supplier is Farnell (www.farnell.de).

Connector for power supply, CAN, RS-232

Steckverbinder Molex	id	costs
Gerade Stiftleisten mit Sperre, 3 Pol (VP=10 Stück)	RS - 453-167	4,75
Buchsengehäuse, 3 Pol (VP=10 Stück)	RS - 467-605	2,58
Crimpkontakte (VP=100 Stück)	RS - 467-598	7,8

