

Memory Region	Register Address (hexadecimal)	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	type	Default setting
Default Settings	16h to 7Fh	reserved								reserved	NA
	Operational Registers	15h	SPI4	enable adv INT	new data INT	latch INT	shadow dis	wake_up pause	wake_up	control	1 0 0 0 0 0 0b
	14h	reserved			range<1:0>		bandwidth<2:0>			control	XXX 00 000b
	13h	customer_reserved 2 <7:0>								status	NA
	12h	customer_reserved 1 <7:0>								status	NA
	11h	any_motion_dur		HG_hyst<2:0>			LG_hyst<2:0>			settings	NA
	10h	any_motion thres<7:0>								settings	NA
	0Fh	HG_dur<7:0>								settings	NA
	0Eh	HG_thres<7:0>								settings	NA
	0Dh	LG_dur<7:0>								settings	NA
	0Ch	LG_thres<7:0>								settings	NA
	0Bh	alert	any motion	counter HG	counter LG	enable HG	enable LG			control	0
	0Ah	reserved	reset INT	reserved	reserved	self test 1	self test 0	soft reset	sleep	control	0
	09h	st result	reserved	reserved	alert phase	LG latched	HG latched	status LG	status HG	status	NA
	08h	unused									
	07h	acc z<9:2> (msb)								data	NA
	06h	acc z<1:0> (lsb)		acc y<9:2> (msb)			new data z			data	NA
	05h	acc y<1:0> (lsb)		acc x<9:2> (msb)			new data y			data	NA
	04h	acc x<1:0> (lsb)		unused			new data x			data	NA
	03h	acc x<1:0> (lsb)		unused			new data x			data	NA
	02h	acc x<1:0> (lsb)		unused			new data x			data	NA
	01h	al_version<3:0>				ml_version<3:0>				data	NA
	00h	unused				chip_id<2:0>				data	---- 010b

Tabelle 1: Messwert- und Konfigurations-Register des BMA020: Die Messwerte für die X-, Y- und Z-Achse finden sich im markierten Bereich in den Registern 0x02 bis 0x07 (aus [1]).

weshalb an dieser Stelle nicht genauer darauf eingegangen werden muss.

Sendet man nun die folgende Zeichenfolge über das Terminal-Programm „HTerm“ ans USB-I2C-Interface, liest dieses 6 Byte aus den Messwert-Registern des Beschleunigungssensors und gibt diese zum PC weiter.

s70 02 s71 06 p

Durch diese Zeichenfolge schreibt das USB-I2C-Interface 0x02 an das I2C-Gerät mit der Adresse 0x70 (das 3D-BS-Modul) und liest anschließend 6 Byte vom selben Gerät zurück.

Als Antwort erhält man z. B. die folgenden 6 Byte im Hexadecimalsformat:

C1 19 C1 EC 01 36

Ins Binärformat umgewandelt bedeutet das:

**11000001 00011001 11000001 11101100
00000001 00110110**

Die durch Unterstreichen markierten Bits zeigen an, dass es sich um neue Messwerte handelt (1 = neu). Die grau markierten Bits sind ungenutzt und daher immer 0. Die fett gedruckten Bits enthalten jeweils in 2 Teilen die 10-Bit-Messwerte für die X-, Y- und Z-Achse. Die genaue Interpretation

dieser Werte findet sich im Datenblatt des BMA020 und in der Tabelle 1.

Möchte man mit dem USB-I2C-Interface kontinuierlich Daten aus dem 3D-BS auslesen, kann man das mit Hilfe eines Makros sehr einfach machen. Dafür sind lediglich die folgenden 3 Zeilen nacheinander vom PC an das USB-I2C-Interface zu senden:

1. Makrospeicher löschen: **v00{}**
2. Neues Makro speichern: **v00{s70 02 s71 06 p >00}**
3. Makroausführung starten: **>00**

Da die genaue Beschreibung aller Funktionen des BMA020 den Umfang dieses Artikels sprengen würde, sei hier auf das ausführliche Datenblatt [1] zum BMA020 von Bosch Sensortec hingewiesen, dessen genaue Lektüre für die Verwendung des 3D-BS unbedingt empfohlen ist. **ELV**

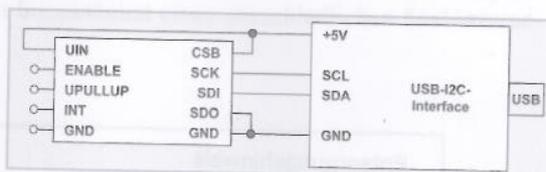


Bild 11: Anbindung des 3D-BS-Moduls ans USB-I2C-Interface über I2C

Links:

[1] Datenblatt zum BMA020:

http://www.bosch-sensortec.com/content/language1/downloads/BMA020_DataSheet_Rev.1.2_30May2008.pdf

[2] USB-I2C-Interface (Art.Nr. 841-23)

<http://www.elv.de/output/controller.aspx?cid=74&detail=10&detail2=24012>

[3] NXP Application Note AN97055 "Bi-directional level shifter for I2c-bus and other systems."

<http://ics.nxp.com/support/documents/i2c/pdf/an97055.pdf>