

Webgestütztes GPIO Management am Beispiel des BeagleBone Black

**Bachelorarbeit im Fachbereich Medienproduktionstechnik an der
Fachhochschule Köln**

Caspar Friedrich

Geboren am 16. Oktober 1986

Mat.-Nr. 11062078

Köln, den 30. September 2014

Betreut durch Prof. Dr. Klaus Ruelberg

Zweitprüfer: Prof. Dr. Luigi Lolacono

Inhaltsverzeichnis

1. Einleitung	9
I. Grundlagen	11
2. Hardware	13
2.1. Single-board Computer (SBC)	13
2.2. System on a Chip (SOC)	14
2.3. BeagleBone Black	14
3. Betriebssysteme	15
3.1. Linux	15
3.1.1. Linux Distributionen	15
4. Webtechnologien	17
4.1. Webserver	17
4.1.1. Lighttpd	17
4.1.2. Weitere Webserver	17
4.2. WebSockets	17
4.2.1. WebSockets vs. PHP	17
4.3. Node.js	17
II. Konfiguration	19
5. Betriebssystem	21
6. Pakete	23
6.1. nodejs	23
III. Implementierung	25
7. Implementierung	27
7.1. boneserver	27
7.2. Webinterface	27

IV. Fazit	29
V. Anhang	31

Abbildungsverzeichnis

7.1. Test figure 27

Tabellenverzeichnis

- .1. BeagleBone Black Expansion Header (P8) 33
- .2. BeagleBone Black Expansion Header (P9) 34

1. Einleitung

In der heutigen Welt ist fast alles mit dem Internet verbunden. Tendenz steigend...

Teil I.

Grundlagen der verwendeten Technologien und Hardware

2. Hardware

2.1. Single-board Computer (SBC)

Ein Single Board Computer oder auch SBC, zu deutsch ein Einplatinenrechner, ist ein Computersystem bei dem alle für die grundsätzliche Funktion nötigen Bauteile auf einer einzelnen Platine verbaut sind. Hierbei sind neben den essenziellen Komponenten wie Prozessor, RAM und ROM oft auch Controller für verschiedene I/O-Schnittstellen, Oszillatoren oder Co-Prozessoren verbaut. Single Board Computer werden vor allem in der Industrie als Steuersysteme eingesetzt, da sie oft billiger und flexibler sind als fest verdrahtete Steuersysteme. Mit zunehmender Miniaturisierung und steigender Leistungsfähigkeit finden sich SBC's heute auch in alltäglichen Geräten wie Autos, Waschmaschinen oder Fernbedienungen.

Technisch gesehen sind auch erste Heimcomputer wie der *C64* oder *Atari ST* Single Board Computer, allerdings lassen sich diese ohne Ein- und Ausgabegeräte wie Maus, Tastatur, Bildschirm nicht sinnvoll einsetzen und werden in der Regel nicht als solche bezeichnet.

Schnittstellen

Single Board Computer verfügen, je nach Anwendungsgebiet, über eine Vielzahl verschiedener analoger und digitaler I/O-Schnittstellen.

Übliche Schnittstellen sind

- GPIO
- PWM
- Analog/Digital Converter (ADC)
- UART¹
- SPI
- I²C

¹Hierüber ist eine Implementierung der verbreiteten RS232/422/485-Schnittstelle möglich und auch üblich

2. Hardware

Aktuelle (Entwickler-)Systeme haben in der Regel einen oder mehrere USB-Anschlüsse (sowohl Client als auch Host Ports sind üblich), oder zumindest einen JTAG-Port, was die Programmierung wesentlich vereinfacht. Des weiteren verfügen Leistungsstärkere Systeme oft auch über einen Grafikausgang².

2.2. System on a Chip (SOC)

Eng verknüpft mit der Entwicklung der SBC ist das Konzept der System-on-a-Chip bzw. SOC. Hierbei werden viele Komponenten eines Systems direkt in einem Einzelnen IC verbaut.

Heutige Single-board Computer mit einem SOC sind sehr leistungsstark und können Taktraten von mehreren GHz haben. Diese Computer sind vom Design her stark an Desktop-Systeme angepasst und können oft mit einem vollwertiges Linux- oder Windows-System betrieben werden.

2.3. BeagleBone Black

Für diese Arbeit verwende ich einen BeagleBone Black³, Ein quelloffenes Entwickler-Board Mit einem ARM® Cortex™-A8 Prozessor von Texas Instruments.

Für diese Arbeit wichtigste Features:

- 1GHz ARM-Prozessor
- 512MB DDR3 RAM
- 2GB⁴ Onboard Flash Memory
- 10/100 Mbit/s Ethernet
- 69 GPIO⁵
- ca. 45 €

²Meist HDMI oder eine der Miniaturvarianten

³Rev. A5C

⁴4GB bei Rev. C

⁵Laut Dokumentation. 27 sind ohne weitere Konfiguration direkt verfügbar

3. Betriebssysteme

Da die Recourcen des BeagleBone Black sehr begrenzt sind, wird für diese Arbeit ein schlankes Betriebssystem benötigt, welches nur wenig Speicher benötigt und geringen Leistungs-Overhead verursacht. Für diesen Zweck gibt es spezielle Versionen der bekannten Betriebssysteme wie Microsoft Windows oder Linux sowie verschiedene „uinoxoide“ Betriebssysteme.

3.1. Linux

Linux hat den Vorteil, dass nahezu alle Software als source code verfügbar ist und im Zweifel angepasst werden kann. Zu dem ist es üblich Lizenzen zu verwenden, die eine nicht-kommerzielle Anwendung sowie Anpassungen kostenfrei zulassen.

Ein eigenes Linux zu entwickeln wäre aus Sicht der Performance sicherlich die beste Wahl und ist auch in der Industrie weitgehend üblich, würde allerdings den Rahmen dieser Arbeit sprengen. Zu dem gibt es einige sehr schlanke und bereits für den BeagleBone angepasste Linux Distributionen.

3.1.1. Linux Distributionen

BeagleBoard.org bietet auf für den BeagleBone Black zwei verschiedene Distributionen an: Ångström und Debian. Beide Distributionen haben ihre Vor- und Nachteile. Ein weiteres Projekt, welches sich unter Entwicklern großer Beliebtheit erfreut ist Arch Linux, welches auch als Basis für diese Anwendung dienen soll.

The Ångström Distribution ist auf dem BeagleBone vorinstalliert und stellt die Hauptdistribution dar. Diese Distribution findet im wesentlichen Anwendung bei Speichersystemen wie NAS oder FTP-Server, wichtigstes feature ist daher der gringe Leistungs- und Speicherbedarf.

Debian Linux gilt im allgemeinen als (rock-)stable und ist eine der verbreitetsten Distributionen, zu dem basieren einige weitere namhafte Distributionen auf Debian Linux. Stärke und gleichzeitig auch Schwäche dieser Distribution sind die langen und umfangreichen Softwaretests. Wenn ein Paket in den offiziellen repositories verfügbar ist kann man zwar davon ausgehen, dass es fehlerfrei funktioniert und zu allen anderen angebotenen Paketen kompatibel ist, allerdings liegt es meist nicht mehr in der aktuellen Version vor. Das kann gerade bei Software aus dem Bereich Netzwerk/Internet problematisch werden.

3. Betriebssysteme

ArchLinux ...

4. Webtechnologien

4.1. Webserver

Was ist ein Webserver? Welche sind die verbreitetsten und was sind ihre Besonderheiten.

4.1.1. Lighttpd

Warum wird Lighttpd verwendet?

4.1.2. Weitere Webserver

Apache

4.2. WebSockets

4.2.1. WebSockets vs. PHP

4.3. Node.js

Was ist Node.js, wie wird es verwendet.

Teil II.

Konfiguration des Betriebssystems

5. Betriebssystem

Hier werden alle, von den Defaults der Distribution abweichenden, Einstellungen beschrieben.

6. Verwendete Pakete

Zusätzlich zu den mitgelieferten Paketen der Distribution werden noch ein HTTP server, ein FTP server und die JavaScript/Node.js engine. Zusätzlich wird noch ein Proxy server benötigt um mit geringem Aufwand SSL-Verschlüsselte Verbindungen zu ermöglichen.

haproxy HAProxy ist eigentlich ein Proxy server, der eingesetzt wird um HTTP-Anfragen auf mehrere Server zu verteilen. Wesentlich interessanter für diese Arbeit ist allerdings, dass der HAProxy nativ SSL-Verschlüsselte Verbindungen verarbeiten kann und dabei in der Basis sehr leicht zu konfigurieren ist.⁽¹⁾

In diesem wird wird HAProxy eingesetzt um WebSocket requests von regulären HTTP requests zu trennen und auf unterschiedliche Dienste weiter zuleiten. Ziel dieser Maßnahme ist es nach außen die gesamte Website hinter einem Port zu betreiben obwohl die beiden Prozessen völlig von einander getrennt sind. So ist die gefahr, dass, bei einem Feldeinsatz, der Port für den WebSocket server von einer Firewall blockiert wird minimal. Die website ist entweder vollständig oder überhaupt nicht zu erreichen. Auch ist der der WebSocket server, der systembedingt mit root-Rechten laufen muss, ausschließlich per WebSocket über den Proxy zu erreichen und ist so gegenüber Angriffen von außen wetgehend sicher.

Ein weiterer wichtiger Punkt ist, dass sich für jeden Server die maximale Anzahl der aktiven Verbindungen bequem per Config file einstellen lassen. So kann ohne besondere Programmierung sichergestellt werden, dass immer nur eine Verbindung zum WebSocket server besteht. Alle weiteren verbindungsanfragen werden auf pending gesetzt und weitergeleitet sobald ein Slot frei wird.

lighttpd Ein webserver...

vsftpd Ein ftp-server...

6.1. nodejs

bonescript

Beschreibung der bonescript¹ library.

¹<https://github.com/jadonk/bonescript>

6. Pakete

WS

Beschreibung der verwendeten websocket library.

Teil III.

Implementierung

7. Implementierung

Das webinterface besteht im aus zwei Teilen: Einem WebSocket server, der die Steuerung der GPIO erledigt und einem Webserver, der die Dokumente ausliefert.

7.1. boneserver

Der WebSocket server ist via Node.js implementiert und und verwendet die *bonescript*¹ library zur steuerung der GPIO.

7.2. Webinterface

¹<https://github.com/jadonk/bonescript>

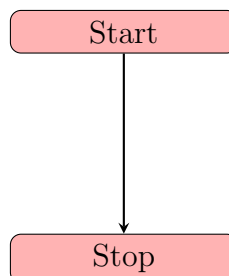


Abbildung 7.1.: Test figure

Teil IV.

Fazit und Erweiterungsmöglichkeiten

Teil V.

Anhang

Head pin	SPINS	ADDR/OFFSET	GPIO NO.	Name	Mode7	Mode6	Mode5	Mode4	Mode3	Mode2	Mode1	Mode0	PIN	Notes
P8_01				DGND										Ground
P8_02				DGND										Ground
P8_03	6	0x815/018	38	GPIO1_6	gpio1[6]						mmc1_dat6	gpmc_ad6	P9	Used on Board (Group: pinmux, emmc2_pins)
P8_04	7	0x817/01c	39	GPIO1_7	gpio1[7]						mmc1_dat7	gpmc_ad7	P9	Used on Board (Group: pinmux, emmc2_pins)
P8_05	2	0x809/008	34	GPIO1_2	gpio1[2]						mmc1_dat2	gpmc_ad2	P8	Used on Board (Group: pinmux, emmc2_pins)
P8_06	3	0x80c/00c	35	GPIO1_3	gpio1[3]						mmc1_dat3	gpmc_ad3	P8	Used on Board (Group: pinmux, emmc2_pins)
P8_07	36	0x800/090	66	TIMER4	timer4							gpmc_adn_ale	R7	
P8_08	37	0x804/094	67	TIMER7	timer7							gpmc_oen_ren	R7	
P8_09	39	0x895/09c	69	TIMER5	timer5						gpmc_be0n_cle		T6	
P8_10	38	0x898/098	68	TIMER6	timer6						gpmc_wen		U6	
P8_11	13	0x834/034	45	GPIO1_13	gpio1[13]						lcd_data18	gpmc_ad13	R12	
P8_12	12	0x820/030	44	GPIO1_12	gpio1[12]						LCD_DATA19	GPAC_AD12	T12	
P8_13	9	0x824/024	23	EHFPMW2B	gpio2[23]					MMC1_DATA1		gpmc_ad9	T10	
P8_14	10	0x829/028	26	GPIO0_26	gpio0[26]					mmc1_dat2		gpmc_ad10	T11	
P8_15	15	0x83c/03c	47	GPIO1_15	gpio1[15]					mmc1_dat7		gpmc_ad15	U13	
P8_16	14	0x838/038	46	GPIO1_14	gpio1[14]					mmc1_dat6		gpmc_ad14	U13	
P8_17	11	0x82c/02c	27	GPIO0_27	gpio0[27]					mmc1_dat3		gpmc_ad11	U12	
P8_18	35	0x88c/08c	65	GPIO2_1	gpio2[1]					gpmc_wait1	lcd_memory_clk	gpmc_clk_mux0	V12	
P8_19	8	0x820/020	22	EHFPMW2A	gpio2[22]					mmc1_dat0		gpmc_ad8	U10	
P8_20	33	0x880/084	63	GPIO1_31	gpio1[31]					mmc1_cmd		gpmc_cs02	V9	Used on Board (Group: pinmux, emmc2_pins)
P8_21	32	0x880/080	62	GPIO1_30	gpio1[30]					gpmc_clk		gpmc_cs01	U9	
P8_22	5	0x814/014	37	GPIO1_5	gpio1[5]					mmc1_dat5		gpmc_ad5	V8	Used on Board (Group: pinmux, emmc2_pins)
P8_23	4	0x810/010	36	GPIO1_4	gpio1[4]					mmc1_dat4		gpmc_ad4	U8	Used on Board (Group: pinmux, emmc2_pins)
P8_24	1	0x804/004	33	GPIO1_1	gpio1[1]					mmc1_dat1		gpmc_ad1	V7	Used on Board (Group: pinmux, emmc2_pins)
P8_25	0	0x800/000	32	GPIO1_0	gpio1[0]					mmc1_dat0		gpmc_ad0	U7	Used on Board (Group: pinmux, emmc2_pins)
P8_26	31	0x87c/07c	61	GPIO1_29	gpio1[29]							gpmc_cs00	V6	
P8_27	56	0x8e0/0e0	86	GPIO2_22	gpio2[22]					gpmc_a8		lcd_vsync	U5	Allocated (Group: nsp_hdmi, bonelt_pins)
P8_28	58	0x8e8/0e8	88	GPIO2_24	gpio2[24]					gpmc_a10		lcd_pclk	V5	Allocated (Group: nsp_hdmi, bonelt_pins)
P8_29	57	0x8e4/0e4	87	GPIO2_23	gpio2[23]					gpmc_a9		lcd_hsync	R5	Allocated (Group: nsp_hdmi, bonelt_pins)
P8_30	59	0x8ec/0ec	89	GPIO2_25	gpio2[25]					gpmc_a11	lcd_ac_bias_en		R6	Allocated (Group: nsp_hdmi, bonelt_pins)
P8_31	54	0x8d8/0d8	10	UART5_CTSN	gpio0[10]					gpmc_a18		lcd_data14	V4	Allocated (Group: nsp_hdmi, bonelt_pins)
P8_32	55	0x8dc/0dc	11	UART5_RTSN	gpio0[11]					gpmc_a19		lcd_data15	T5	Allocated (Group: nsp_hdmi, bonelt_pins)
P8_33	53	0x8d4/0d4	9	UART4_RTSN	gpio0[9]					gpmc_a17		lcd_data13	V3	Allocated (Group: nsp_hdmi, bonelt_pins)
P8_34	51	0x8cc/0cc	81	UART3_RTSN	gpio2[17]					gpmc_a15		lcd_data11	U4	Allocated (Group: nsp_hdmi, bonelt_pins)
P8_35	52	0x8d0/0d0	8	UART4_CTSN	gpio0[8]					gpmc_a16		lcd_data12	V2	Allocated (Group: nsp_hdmi, bonelt_pins)
P8_36	50	0x8c8/0c8	80	UART3_CTSN	gpio2[16]					gpmc_a12		lcd_data10	U3	Allocated (Group: nsp_hdmi, bonelt_pins)
P8_37	48	0x8c0/0c0	78	UART5_TXD	gpio2[14]					gpmc_a12		lcd_data8	U1	Allocated (Group: nsp_hdmi, bonelt_pins)
P8_38	49	0x8c4/0c4	79	UART5_RXD	gpio2[15]					gpmc_a13		lcd_data9	U2	Allocated (Group: nsp_hdmi, bonelt_pins)
P8_39	46	0x8b8/0b8	76	GPIO2_12	gpio2[12]					gpmc_a6		lcd_data6	T3	Allocated (Group: nsp_hdmi, bonelt_pins)
P8_40	47	0x8bc/0bc	77	GPIO2_13	gpio2[13]					gpmc_a7		lcd_data7	T4	Allocated (Group: nsp_hdmi, bonelt_pins)
P8_41	44	0x8b0/0b0	74	GPIO2_10	gpio2[10]					gpmc_a4		lcd_data4	T1	Allocated (Group: nsp_hdmi, bonelt_pins)
P8_42	45	0x8b4/0b4	75	GPIO2_11	gpio2[11]					gpmc_a5		lcd_data5	T2	Allocated (Group: nsp_hdmi, bonelt_pins)
P8_43	42	0x8a8/0a8	72	GPIO2_8	gpio2[8]					gpmc_a2		lcd_data2	R3	Allocated (Group: nsp_hdmi, bonelt_pins)
P8_44	43	0x8ac/0ac	73	GPIO2_9	gpio2[9]					gpmc_a3		lcd_data3	R4	Allocated (Group: nsp_hdmi, bonelt_pins)
P8_45	40	0x8a0/0a0	70	GPIO2_6	gpio2[6]					gpmc_a0		lcd_data0	R1	Allocated (Group: nsp_hdmi, bonelt_pins)
P8_46	41	0x8a4/0a4	71	GPIO2_7	gpio2[7]					gpmc_a1		lcd_data1	R2	Allocated (Group: nsp_hdmi, bonelt_pins)
P9 Header	cat \$PINS	ADDR + 44e10000	GPIO NO. (Mode 7)	Name	Mode 7	Mode 6	Mode 5	Mode 4	Mode 3	Mode 2	Mode 1	Mode 0	CPU PIN	Updates Available at www.derekmolloy.ie
Offset from: 44e10800														

GPIO Settings			
Bit 6	Bit 5	Bit 4	Bit 3
Slew Control	Receiver Active	Pullup/Pulldown	Enable Pullup/down
0 Fast	0 Disable	0 Pulldown select	000 Mode 0 to
1 Slow	1 Enable	1 Pullup select	111 Mode 7

e.g. OUTPUT GPIO(mode7) 0x07 pullup, 0x17 pullup, 0x27 pullup, 0x37 pullup, 0x47 pullup, 0x57 pullup, 0x67 pullup, 0x77 pullup, 0x87 pullup, 0x97 pullup, 0xa7 pullup, 0xb7 pullup, 0xc7 pullup, 0xd7 pullup, 0xe7 pullup, 0xf7 pullup

by Derek Molloy (www.derekmolloy.ie)

Tabelle 1.: BeagleBone Black Expansion Header (P8)

Head pin	SPINS	ADDR/OFFSET	Name	GPIO NO.	Mode7	Mode6	Mode5	Mode4	Mode3	Mode2	Mode1	Mode0	PIN	Notes
P9_01			GND											Ground
P9_02			GND											Ground
P9_03			DC 3.3V											250mA Max Current
P9_04			DC 3.3V											250mA Max Current
P9_05			VDD 5V											1A Max Current (only if DC jack powered)
P9_06			VDD 5V											1A Max Current (only if DC jack powered)
P9_07			SYS 5V											250mA Max Current
P9_08			SYS 5V											250mA Max Current
P9_09			PMR_BUT											Has a 5V level (pulled up by TP565217Q)
P9_10			SYS_RESETn											
P9_11	28	0x870/070	UART1_RXD	30	gpio[30]			mmc1_scd	rm12_cs_dv	gpmc_cs14	mi2_cs	RESET_OUT	A10	
P9_12	30	0x878/078	GPIO_28	60	gpio[28]			gpmc_dir	mi2_dat3	gpmc_cs16	mi2_cs	gpmc_wait0	T17	NB: GPIOs limit current to 4-6mA output and approx. 8mA on input.
P9_13	29	0x879/079	UART1_TXD	31	gpio[31]			mmc2_scd	mi2_nerr	gpmc_cs15	mi2_cs	gpmc_de_in	U18	
P9_14	18	0x848/048	EMIPWM1A	50	gpio[18]			gpmc_a18	mmc2_dat1	gpmc_cs13	mi2_nerr	gpmc_wpn	U17	
P9_15	16	0x849/049	GPIO_16	48	gpio[16]			gpmc_a16	mmc2_ben	gpmc_a18	mi2_bds	gpmc_a2	U14	
P9_16	19	0x847/047	EMIPWM1B	51	gpio[19]			gpmc_a19	mmc2_dat2	gpmc_a16	mi2_ben	gpmc_a0	U13	
P9_17	87	0x957/757	I2C1_SCL	5	gpio[5]				ehrpwm0_syncl	gpmc_a19	mi2_bds	gpmc_a3	U14	
P9_18	86	0x958/758	I2C1_SDA	4	gpio[4]				ehrpwm0_syncl	gpmc_a19	mi2_bds	gpmc_a3	U14	
P9_19	95	0x977/777	I2C2_SCL	13	gpio[13]			spl_cs1	I2C2_SCL	I2C1_SDA	mmc2_sdwp	spi0_cs0	A16	
P9_20	94	0x978/778	I2C2_SDA	12	gpio[12]			spl_cs0	I2C2_SDA	I2C1_SDA	mmc1_sdwp	spi0_d1	B16	Allocated (Group: pinmux_1x2_pins)
P9_21	85	0x954/754	UART2_TXD	3	gpio[3]				I2C2_SDA	dcant0_tx	timer5	uart1_rsn	D17	Allocated (Group: pinmux_1x2_pins)
P9_22	84	0x950/750	UART2_RXD	2	gpio[2]				ehrpwm0B	dcant0_tx	timer6	uart1_rsn	D18	
P9_23	17	0x844/044	GPIO_17	49	gpio[17]			EMU2_mux1	ehrpwm0A	I2C2_SCL	uart2_bds	spi0_d0	B17	
P9_24	97	0x984/784	UART1_TXD	15	gpio[15]			ehrpwm0_synco	mmc2_dat0	I2C2_SDA	uart2_rdv	spi0_sclk	A17	
P9_25	107	0x987/787	GPIO3_21	117	gpio[21]			EMU4_mux2	I2C1_SCL	dcant1_rx	gmi2_rdv	uart1_bds	V14	
P9_26	96	0x980/780	UART1_RXD	14	gpio[14]				mmc1_sclk	mmcsp0_axr3	eQEP0_strobe	mmcsp0_ahckx	A14	Allocated (Group: mcasp0_pins)
P9_27	105	0x989/789	GPIO3_19	115	gpio[19]			EMU2_mux2	I2C1_SDA	mmcsp0_axr2	mmcsp0_syncl	uart1_rdv	D16	
P9_28	103	0x997/797	SPI1_CS0	113	gpio[17]			eCAP2_in_PWM2_out	mmcsp1_fsx	mmcsp0_axr3	ehrpwm0B	mmcsp0_fsr	C13	
P9_29	101	0x994/794	SPI1_D0	111	gpio[15]			mmc1_scd_mux1	spl_d0	mmcsp0_axr2	ehrpwm0B	mmcsp0_fsx	C12	Allocated (Group: mcasp0_pins)
P9_30	102	0x998/798	SPI1_D1	112	gpio[16]			mmc2_scd_mux1	spl_d1	mmcsp0_axr0	ehrpwm0B	mmcsp0_fsx	B13	Allocated (Group: mcasp0_pins)
P9_31	100	0x990/790	SPI1_SCLK	110	gpio[14]			mmc0_scd_mux1	spl_sclk	mmcsp0_axr0	ehrpwm0B	mmcsp0_fsx	D12	
P9_32			VADC							mmcsp0_axr0	ehrpwm0A	mmcsp0_ahckx	A13	Allocated (Group: mcasp0_pins)
P9_33			AIN4										C8	Voltage Reference for ADC (NB: 1.8V)
P9_34			AGND											NB: 1.8V tolerant
P9_35			AIN6										A8	Ground for ADC
P9_36			AIN5										B8	NB: 1.8V tolerant
P9_37			AIN2										B7	NB: 1.8V tolerant
P9_38			AIN3										A7	NB: 1.8V tolerant
P9_39			AIN0										B6	NB: 1.8V tolerant
P9_40			AIN1										C7	NB: 1.8V tolerant
P9_41A	109	0x9b4/7b4	CLKOUT2	20	gpio[20]			EMU3_mux0	clkout2	tdkin	xdma_event_intr1	xdma_event_intr1	D14	Both signals are connected to P21 of P11
P9_41B			GPIO3_20	116	gpio[20]			emu3	Mcasp1_axr0		eQEP0_index	mcasp0_axr1	D13	Both signals are connected to P21 of P11
P9_42A	89	0x964/764	GPIO_7	7	gpio[7]			xdma_event_intr2	pr1_escap0_escap_capin_apwm_o	spl_cs1	uart3_bds	eCAP0_in_PWM0_out	C18	Both signals are connected to P22 of P11
P9_42B			GPIO3_18	114	gpio[18]				Mcasp1_ahckx	Mcasp0_axr2	eQEP0A_in	Mcasp0_ackr	B12	Both signals are connected to P22 of P11
P9_43			GND											Allocated (Group: mcasp0_pins) - See Pg.50 of the SRM
P9_44			GND											Ground
P9_45			GND											Ground
P9_46			GND											Ground
P9_Header	cat SPINS	ADDR +	Name	GPIO NO.	Mode 7									Notes
	Allocated	44e10000	Offset from:	(Mode 7)										For updates see: www.derekmolloy.ie
		44e10800	Offset from:											Please e-mail me directly at: derek@derekmolloy.ie if you notice a mistake

Bit 6	Bit 5	Bit 4	Bit 3	Bit 2,1,0
Slow Control	Receiver Active	Enable Pullup/Pulldown	Enable Pullup/Pulldown	Mux Mode
0 Fast	0 Disable	0 Pulldown select	0 Enabled	000 Mode 0 to
1 Slow	1 Enable	1 Pullup select	1 Disabled	111 Mode 7

e.g. OUTPUT GPIO(mode7) 0x27 pullup, 0x37 pullup, 0x7F no pullup/down
e.g. INPUT GPIO(mode7) 0x27 pullup, 0x37 pullup, 0x7F no pullup/down

by Derek Molloy (www.derekmolloy.ie)

Tabelle .2.: BeagleBone Black Expansion Header (P9)

Literaturverzeichnis

- [1] KÜHNAST, Charly: Passthrough und Offloading: HTTPS balancieren mit HA-Proxy 1.5. In: *ADMIN Magazin* (2014), Nr. 3, S. 32–34
- [2] WEISER, Mark: The Computer for the 21st Century. In: *Scientific American* (1991), Nr. 265, S. 94–104