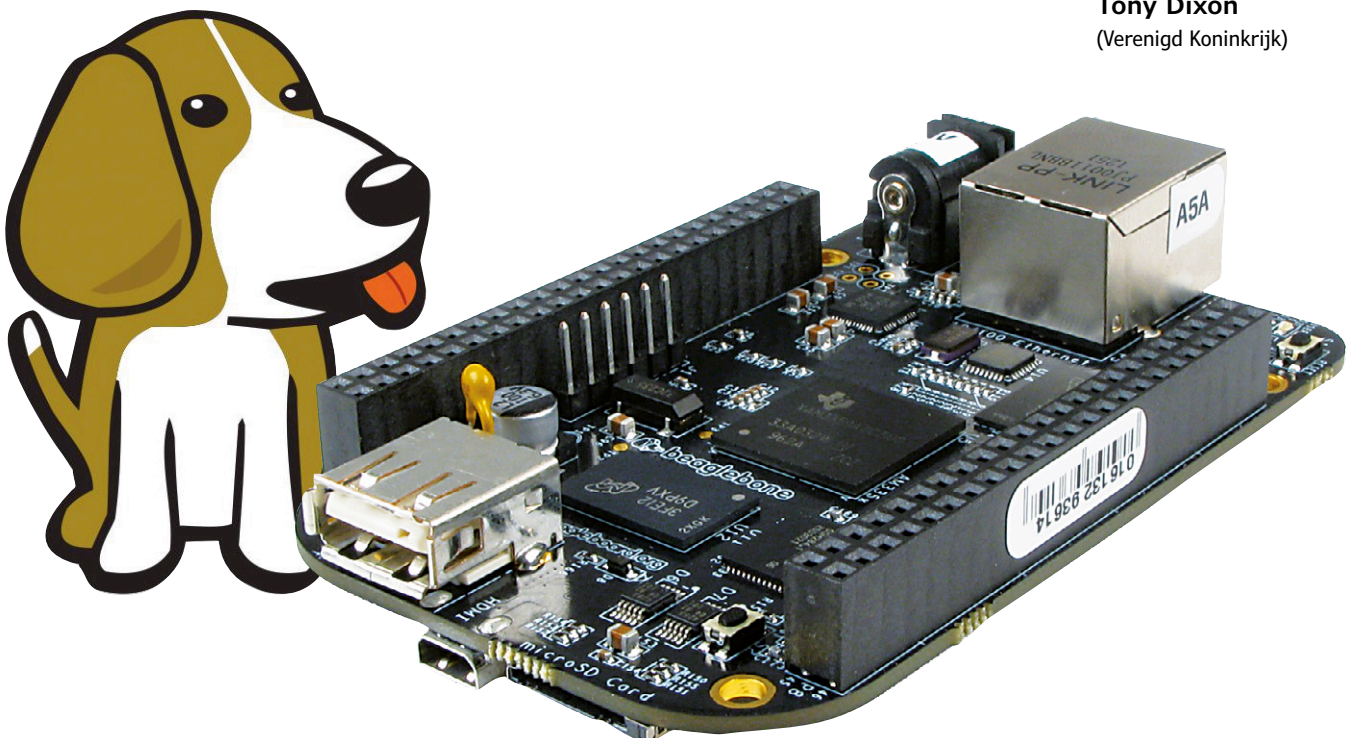


BeagleBone Black, The Sequel (3)

Deel 3: Analoge ingangen van de BBB

Tony Dixon
(Verenigd Koninkrijk)



In de eerste .POST over de BeagleBone Black (BBB) hebben we gekeken naar de digitale I/O. In deze aflevering komen de analoge mogelijkheden van de BBB aan bod. Dus pak de [USB]-lijn en laat de hond uit.

Als je ziet dat sommige mensen in Silicon Valley en omgeving tegenwoordig altijd rondhangen in de buurt van Ethernet-aansluitingen, zou je bijna gaan denken dat de hele wereld digitaal is, maar dat is hij niet!

Analoge I/O met de BBB

De ADC van de BBB heeft de volgende eigenschappen:

- 12-bits resolutie (0 tot 4095)
- 125 ns sampletijd
- Bereik 0 tot 1,8 V (!!!!)

Op de uitbreidingsconnectors van de BBB zijn 7 analoge ingangen beschikbaar. Zie **tabel 1** voor een snel overzicht van de analoge pennen en **tabel 2** voor de complete penbezetting van de BBB.

Naast de analoge signaallijnen zijn er ook aparte analoge voedingslijnen: AVCC (Analogue VCC) en AGND (Analog Ground).

Hoewel de digitale I/O-pennen (GPIO's) van de BBB compatibel zijn met 3,3 V, kunnen de analoge pennen maximaal 1,8 V verwerken. Let dus op wat voor spanningen u aansluit, anders komt uw BBB al snel in de hondenhemel terecht. Voor het meten van spanningen

Tabel 1: Penconfiguratie van de analoge poort.

Signaal (P9)	Pen
AIN0	39
AIN1	40
AIN2	37
AIN3	38
AIN4	35
AIN5	36
AIN6	33
AGND	34
AVCC	32

groter dan 1,8 V moet een spanningsdeler worden gebruikt. De weerstand tussen de ingangslijn en de massa in die spanningsdeler moet een waarde van 1 kΩ hebben.

Gebruik van sysfs

Net als bij de eerdere voorbeelden met digitale I/O-lijnen kunnen we ook nu weer gebruik

maken van de virtuele file/driver-structuur 'sysfs' van Linux om met de analoge pennen te werken, zonder één regel code te schrijven. Open een terminal-venster en begin met het inschakelen van de analoge driver. Type het volgende commando in het terminal-venster:

```
echo cape-bone-iio > /sys/devices/bone_capemgr.* /slots
```

Met het Linux-commando cat kunnen we de spanning in millivolts (mV) op AIN0 meten (en weergeven) door te typen:

```
cat /sys/bus/iio/devices/iio\:device0/in_voltage0_raw
```

Als we de ruwe waarde in de ADC willen zien in plaats van de spanning, kan dat met:

```
cat /sys/devices/ocp.2/helper.14/AIN0
```

Tabel 2: Volledige penconfiguratie van de BBB.

Signaal	P8			Signaal	P9			Signaal
GND	1		2	GND	1		2	GND
GPIO1_6	3		4	GPIO1_7	3		4	3.3V
GPIO1_2	5		6	GPIO1_3	5		6	5V
TIMER4	7		8	TIMER7	7		8	5V_SYS
TIMER5	9		10	TIMER6	9		10	PWR_BUTTON
GPIO1_13	11		12	GPIO1_12	11		12	UART4_RXD
EHRPWM2B	13		14	GPIO2_26	13		14	GPIO4_TXD
GPIO1_15	15		16	GPIO1_14	15		16	GPIO1_16
GPIO0_27	17		18	GPIO2_1	17		18	I2C1_SCL
EHRPWM2A	19		20	GPIO1_31	19		20	I2C2_SCL
GPIO1_30	21		22	GPIO1_5	21		22	UART2_TXD
GPIO1_4	23		24	GPIO1_1	23		24	GPIO1_17
GPIO1_0	25		26	GPIO1_29	25		26	GPIO3_21
GPIO2_22	27		28	GPIO2_24	27		28	GPIO3_19
GPIO2_23	29		30	GPIO2_25	29		30	SPI1_D0
UART5_CTS	31		32	UART5_RTS	31		32	SPI1_SCLK
UART4_RTS	33		34	UART3_RTS	33		34	AIN4
UART4_CTS	35		36	UART3_CTS	35		36	AIN6
UART5_TXD	37		38	UART5_RXD	37		38	AIN2
GPIO2_12	39		40	GPIO2_13	39		40	AIN0
GPIO2_10	41		42	GPIO2_11	41		42	GPIO_20
GPIO2_08	43		44	GPIO2_09	43		44	GND
GPIO2_6	45		46	GPIO2_07	45		46	GND

Coderen

Sysfs is heel handig om even snel iets te testen, maar we kunnen er ook op voortbouwen en de functionaliteit onderbrengen in een echt C(++)-programma.

Voor deze test sluiten we een potentiometer van 5 kΩ aan tussen AVCC (pen 32) en AGND (pen 34). De loper verbinden we met AIN0 (pen 39). Open een terminal-sessie en start de *nano* editor met:

```
nano analogue.cpp
```

Type het programma van listing 1 in. Als de code is ingevoerd, sla het programma dan op door Ctrl+X, Y en ENTER in te voeren. Wie niet wil typen, kan het programma 'analogue.cpp' downloaden van onze website [1]. Het staat in het bestand 130492-11.zip.

Als het programma is opgeslagen, kunnen we het vanuit de terminal compileren met het commando:

```
g++ analogue.cpp -o analogue
```

Als het programma zonder fouten is gecompileerd, kunnen we het uitvoeren met het commando:

```
./analogue
```

Nu wordt de spanning op de analoge pen elke seconde gemeten. Draai aan de potmeter en kijk naar de getallen die worden afgedrukt. We kunnen dit stukje code gemakkelijk gebruiken om temperatuur te meten met een TMP36 die toevallig juist een uitgangsspanning tussen 0 V en 1,8 V genereert.

(130492)

Weblinks

[1] Beagle-website: <http://beagleboard.org>

[2] www.elektor-magazine.nl/130492

Listing 1

```
#include <stdlib.h>
#include <stdio.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <unistd.h>

int main()
{
    int fd, fdstat;
    char buffer[1024];

    const char AIN0 [] = "/sys/bus/iio/devices/iio\:device0/in_voltage0_raw";

    /* Open sysfs to Analogue input */
    fd = open (AIN0, O_RDONLY);

    while (1)
    {
        /* Read Analogue input */
        fdstat = read(fd, buffer, sizeof(buffer));

        /* Print result */
        if (fdstat != -1)
        {
            buffer[fdstat] = '\0';
```

```
    /* Print string and value*/  
    printf("AIN0 value = %s \n", buffer);  
  
    lseek(fd, 0, 0);  
}  
  
/* Small delay */  
sleep(1);  
}  
  
/* Close sysfs & exit */  
close(fd);  
return 0;  
}
```