

---

# CM-3G 周辺モジュール拡張技術文書 BH1750FVI センサ(光)

(第 1 版)

Copyright (C)2015 株式会社コンピューテックス

---

## 目次

1. はじめに.....	1
2. BH1750FVI について.....	1
3. 接続図.....	1
4. アプリケーション・ソース(C 言語).....	2
5. アプリケーションのコンパイル.....	6
6. アプリケーションの実行.....	6

---

## 1. はじめに

本書は、CM-3G 開発キットで BH1750FVI センサを使用するための補足マニュアルです。  
CM-3G の拡張コネクタに接続可能なセンサを使用するための手順を記述します。

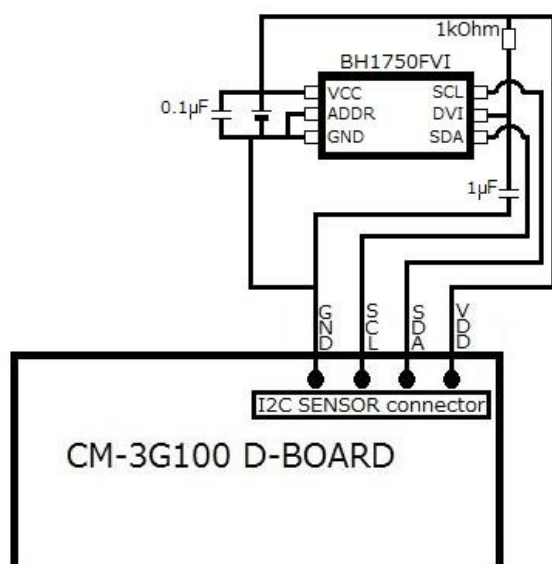
## 2. BH1750FVI について

BH1750FVI は I2C で接続可能な光センサです。

## 3. 接続図

CM-3G100 D-BOARD のセンサコネクタに BH1750FVI を接続します。

BH1750FVI	CM-3G D-Board SENSOR connector
VCC	VDD
ADDR	GND
SCL	SCL
SDA	SDA
DVI	GND (through a capacitor)
GND	GND



## 4. アプリケーション・ソース(C言語)

Linux アプリケーションの C 言語ソースと Makefile は以下の通りです。

```
/*
 * bh1750fvi.c
 *
 * Copyright (C) 2012 Computex Co.,Ltd.
 *
 * Sample to test BH1750FVI using CM-3G CHECK BOARD
 */

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <errno.h>
#include <unistd.h>
#include <fcntl.h>
#include <linux/i2c-dev.h>

#define I2C_DEVICE          "/dev/i2c-3"
#define DEV_ADDRESS        0x23

void usage(char *app_name)
{
    printf("%s - Displays the H-Resolution Mode, H-Resolution Mode2 and L-Resolution mode
values for BH1750FVI sensor\n", app_name);
    printf("Usage: %s -d [device path]\n", app_name);
    printf("\t\t-d - i2c device path (Default: \"I2C_DEVICE\")\n");
    printf("Eg: %s\n", app_name);
    exit(0);
}

/*****
 * Main
 *****/
int main(int argc, char* argv[])
{
    unsigned char buf[10];
    int d;
    char shortoptions[] = "d:h";
    char input_device[16];
    float hres, hres2, lres;

    input_device[0] = 0;
    for (;;) {
        d = getopt_long(argc, argv, shortoptions, (void *)NULL, &index);
        if (-1 == d)
            {
```

```

        break;
    }
    switch (d) {
    case 'd':
        strcpy(input_device, optarg);
        break;
    default:
        usage(argv[0]);
    }
}
if(input_device[0] == 0)
    strcpy(input_device, I2C_DEVICE);

int file = open(input_device, O_RDWR);
if(file < 0) {
    printf("Error: "I2C_DEVICE " open failed\n");
    exit(1);
}

if(ioctl(file, I2C_SLAVE, DEV_ADDRESS) < 0) {
    printf("Error: %s setting device address to 0x%xfailed\n", input_device,
DEV_ADDRESS);
    goto out;
}
/* power on the device */
buf[0] = 0x01;
if (write(file, buf, 1) != 1) {
    printf("Failed to write to i2c bus. Check if BH1750FVI is connected\n");
    goto out;
}
/* reset the device */
buf[0] = 0x07;
if (write(file, buf, 1) != 1) {
    goto err_write;
}

printf("H-RES      H-RES2      L-RES\n");

/* measure one time values */
while(1)
{
    /* power on the device */
    buf[0] = 0x01;
    if (write(file, buf, 1) != 1) {
        goto err_write;
    }
    /* H-Resolution Mode */
    buf[0] = 0x20;
    if (write(file, buf, 1) != 1) {
        goto err_write;
    }
}

```

```

        usleep(120000);                /* 120 ms wait */
        buf[0] = 0x0;
        if (read(file, buf, 2) != 2) {
            goto err_read;
        }
        hres = ((float)(buf[1] | (buf[0]<<8)))/1.2;

        /* power on the device */
        buf[0] = 0x01;
        if (write(file, buf, 1) != 1) {
            goto err_write;
        }
        /* H-Resolution Mode2 */
        buf[0] = 0x21;
        if (write(file, buf, 1) != 1) {
            goto err_write;
        }
        usleep(120000);                /* 120 ms wait */
        buf[0] = 0x0;
        if (read(file, buf, 2) != 2) {
            goto err_read;
        }
        hres2 = ((float)(buf[1] | (buf[0]<<8)))/1.2;

        /* power on the device */
        buf[0] = 0x01;
        if (write(file, buf, 1) != 1) {
            goto err_write;
        }
        /* L-Resolution Mode */
        buf[0] = 0x23;
        if (write(file, buf, 1) != 1) {
            goto err_write;
        }
        usleep(120000);                /* 120 ms wait */
        buf[0] = 0x0;
        if (read(file, buf, 2) != 2) {
            goto err_read;
        }
        lres = ((float)(buf[1] | (buf[0]<<8)))/1.2;

        printf("%-11.3f %-11.3f %-11.3f\n", hres, hres2, lres);
    }

err_write:
    printf("Failed to write to i2c bus\n");
    goto out;
err_read:
    printf("Failed to read from the i2c bus.\n");
out:
    close(file);

```

```
    return 0;
}
```

```
# Makefile

BUILDR00T_DIR:=./../../../../../
OUTPUT_DIR:=output
CROSS_COMPILE:=$(BUILDR00T_DIR)/$(OUTPUT_DIR)/host/usr/bin/arm-linux-
#APP_DEBUG:=-gdb

CC = $(CROSS_COMPILE)gcc $(APP_DEBUG) -I$(TSLIB_DIR) -I$(KERNEL_HEADERS) -00 -lm
APP= bh1750fvi
LIB= -ldl -lpthread

SRCS= bh1750fvi.c

all: $(SRCS)
    $(CC) -o $(APP) $(SRCS) $(LIB)

clean:
    rm -f *.o *~ $(APP)
```

## 5. アプリケーションのコンパイル

1. `${buildroot}/board/ckb/cm-3g/samples/` の下にフォルダを作成します。  
`$ mkdir -p ${buildroot}/board/ckb/cm-3g/samples/service-patch/BH1750FVI`
2. ソース(`bh1750fvi.c`)と `Makefile` を作成したフォルダにコピーして `make` を実行します。  
`$ make -C ${buildroot}/board/ckb/cm-3g/samples/service-patch/BH1750FVI/  
[board/ckb/cm-3g/samples/service-patch/BH1750FVI] フォルダに、実行バイナリ(bh1750fvi)が作成されます。`
3. 上記で作成された実行バイナリを SD カードの `/usr/bin` にコピーしてください。
4. `$ sudo cp ${buildroot}/board/ckb/cm-3g/samples/service-patch/BH1750FVI/bh1750fv /media/rootfs/usr/bin`

## 6. アプリケーションの実行

以下のコマンドで、アプリケーションを実行します。

```
# bh1750fv  
To terminate the application press CTRL-C
```

## 変更履歴

日付	版	内容
2015-09	1	初版

---



- 
- 本書の内容の一部、または全部を無断で使用することや、複製することはできません。
  - 本書の内容、および仕様に関しては将来予告なしに変更することがあります。
  - 本書は万全の注意を払って生産されていますが、ご利用になった結果について当社は一切の責任を負いかねますのでご了承ください。
  - COMPUTEX は、(株)コンピューテックスの登録商標です
  - その他本書で取り上げる会社名および製品名などは、一般に各メーカーの商標、または登録商標です。

---



株式会社コンピューテックス

本社

〒605-0846 京都市東山区五条橋東 4-432-13 対嵐坊ビル  
TEL:075-551-0528(代) FAX:075-551-2585

東京営業所

TEL:03-5753-9911(代) FAX:03-5753-9917

テクニカルセンタ

TEL:075-551-0373 FAX:075-551-2585

**CM-3G 周辺モジュール拡張技術文書**  
**BH1750FVI センサ(光)**  
**2015年9月 第1版**  
**CX516(A)1509**

---