

# Budowa projektora DLP

## z użyciem Raspberry Pi 3 oraz modułu TI LightCrafter Display 2000

*Już od dobrych kilkunastu lat na rynku komponentów elektronicznych mamy do czynienia z rewolucją w obszarze układów mikroelektromechanicznych (MEMS). Układy te, jeszcze kilka lat temu znane wyłącznie z zastosowań w mikrofonach i czujnikach ciśnienia, dziś są niemal nieodłącznym elementem większości urządzeń mobilnych. Badania prowadzone nad układami MEMS (pod kątem ich miniaturyzacji, nowych obszarów zastosowań i obniżenia kosztów produkcji) wymusiły również rewolucję w obszarze układów MOEMS, czyli układów łączących zagadnienia mikrooptyki i MEMS. Jednym z najbardziej reprezentatywnych przykładów MOEMS-ów są projektory DLP oraz zastosowana w nich technologia sterowanej matrycy milionów mikroluster. To co do niedawna wydawało się tematyką rodem z filmów science fiction, dziś trafia pod strzechy i umożliwia zbudowanie własnego miniprojektora w domowym zaciszu.*

W artykule omówiono podłączenie i obsługę modułu projektora DLP LightCrafter Display 2000 produkcji Texas Instruments, do komputera jednopłytkowego Raspberry Pi 3. Moduł ten został pierwotnie zaprojektowany do łatwej współpracy z komputerami jednopłytkowymi BeagleBone. Niestety, pod kątem wydajności, zwłaszcza w aplikacjach multimedialnych, BeagleBone znacznie ustępuje konkurencyjnemu rozwiązaniu, jakim jest Raspberry Pi 3. Jak jednak zostanie

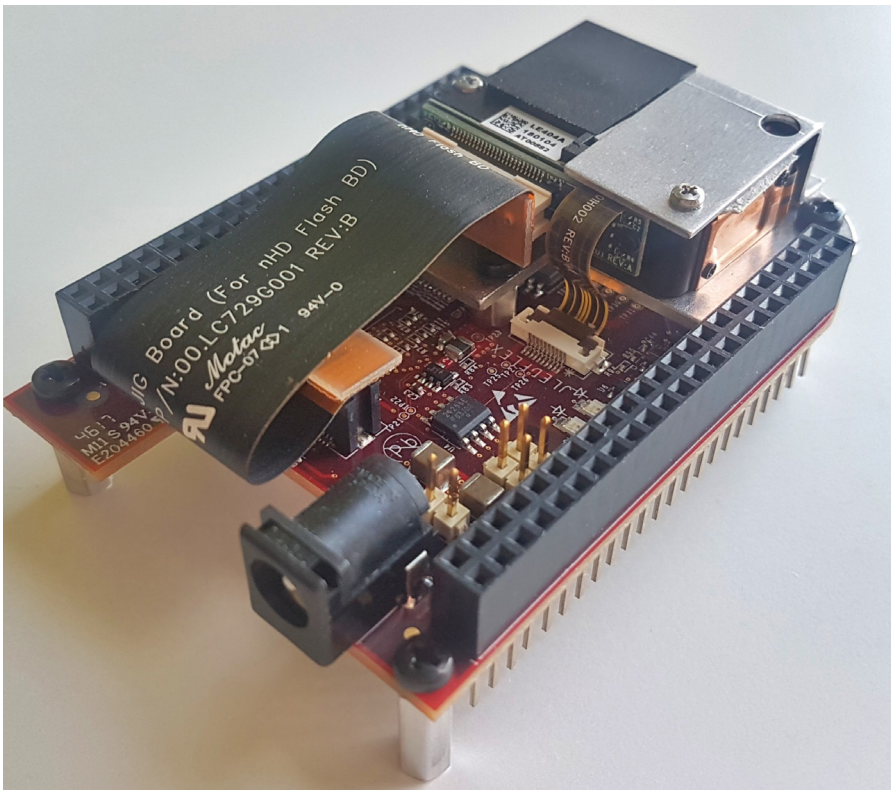
to przedstawione w artykule, konfiguracja Raspberry Pi do pracy z modułem LightCrafter Display 2000 jest stosunkowo prosta a radość z samodzielnej budowy bezgłośnego i energooszczędnego projektora multimedialnego – bezcenna.

### **Projektory DLP. Moduł DLP LightCrafter Display 2000**

W świecie elektroniki użytkowej od lat obserwujemy toczące się wojny standardów

i rozwiązań technologicznych, jak np. rozstrzygnięte już starcia formatów Blue-ray i HD-DVD czy technologii plazmowej i LCD. W segmencie projektorów multimedialnych klienci wciąż zadają pytanie – wybrać projektor LCD bazujący na panelach ciekłokrystalicznych, czy jednak zdecydować się na technologię mikroluster zastosowaną w projektorach DLP? Nie podejmując się tutaj żadnych prób rozstrzygnięcia tego sporu, warto podkreślić jedno – w świecie projektorów DLP niepodzielny prym wiedzie firma Texas Instruments, która jest autorem i właścicielem patentów z zakresu technologii DLP. Choć przywykło się mówić, że wszystkie wojny technologiczne wygrywa końcowy klient – otrzymujący tańsze i lepsze rozwiązanie – nie mniejsze zwycięstwo odnoszą również... konstruktorzy i programiści.

Firma Texas Instruments swoją przewagę rynkową próbuje wypracować nie tylko na poziomie czysto technicznym (projektory DLP to wyższy kontrast obrazu, głębsze czernie i niższy pobór mocy), ale również w wykorzystywanym modelu biznesowym – dostarczając konstruktorom i producentom urządzeń końcowych kompleksowy



Rysunek 1. Moduł projektora DLP LightCrafter Display 2000

zestaw układów, zestawów deweloperskich i narzędzi do szybkiego i łatwego zaadaptowania technologii DLP w konstruowanych projektorach. Przykładem takich działań jest zestaw DLP LightCrafter Display 2000 (DLP-DLCR2000EVM) – **rysunek 1**.

Sercem każdego projektora DLP jest układ DMD (z ang. *Digital Micromirror Device*), oparty na technologii MOEMS i będący sterowaną cyfrową tablicą tysięcy mikroluster (każde o powierzchni kilkunastu mikrometrów kwadratowych) odbijających lub rozpraszających padający na nie strumień świetlny. W zależności od konstrukcji projektora możemy wyróżnić urządzenia z pojedynczym układem DMD (wówczas światło odbite przepuszczane jest przez wirującą trójbarwny dysk umożliwiający uzyskanie wielobarwnego obrazu) lub z wieloma matrycami mikroluster (każdy z trzech składowych kolorów RGB jest wyświetlany przez oddzielny układ DMD). Wadą pierwszego z rozwiązań jest efekt tęczy, natomiast

drugiego wysoka cena. Układ DMD, filtry, soczewki oraz źródło światła stanowią część optyczną w blokowej budowie typowego projektora DLP. Uzupełnieniem toru optycznego jest część elektroniczna, w skład której zaliczamy układ kontrolera matrycy DMD, układ PMIC do zarządzania zasilaniem i sterowaniem diodami LED, pamięci EEPROM/Flash oraz w zależności od typu pracy projektora – wbudowany mikroprocesor lub układ interfejsowy (do konwersji sygnałów HDMI, VGA, ...). Blokowy schemat typowego projektora DLP został przedstawiony na **rysunku 2**.

W zależności od docelowego przeznaczenia produktu (rynek IoT, projektory mobilne do automatyki domowej, kino domowe, ...) firma Texas Instruments przygotowała szereg układów DMD różniących się m.in. rozmiarem matrycy a tym samym rozdzielczością wyświetlanego obrazu – od standardu nHD (układ DLP2000) poprzez WVGA (DLP2010 i DLP3000), 720p (DLP3010), WXGA (DLP4500) aż do rozdzielczości 1080p (DLP4710) i 4K. Dla

P1			P2		
1	GND	GND	2	GND	GND
3	VCC	-	4	-	4
5	-	-	6	-	6
7	-	-	8	-	8
9	-	-	10	-	10
11	-	-	12	-	12
13	-	-	14	-	14
15	WRD_ON_SCK	-	16	-	16
17	-	-	18	-	18
19	IEC_SCL	IEC_SDA	20	-	20
21	-	-	22	-	22
23	-	-	24	-	24
25	-	-	26	-	26
27	-	-	28	-	28
29	-	-	30	-	30
31	-	-	32	-	32
33	-	-	34	-	34
35	-	-	36	-	36
37	-	-	38	-	38
39	-	-	40	-	40
41	-	-	42	-	42
43	HOST	GND	44	-	44
45	GND	GND	46	-	46

Rysunek 3. Schemat wyprowadzeń projektora DLP LightCrafter Display 2000

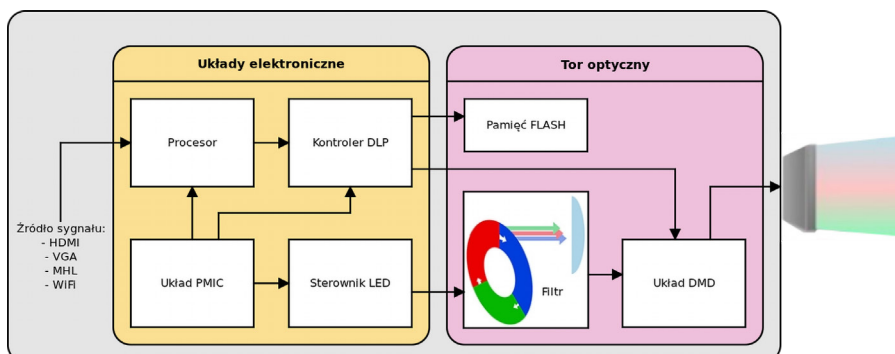
każdej z matryc DMD przygotowano również kontroler matrycy oraz układ PMIC. Tak dobrane zestawy tworzą gotowe platformy do budowy własnych projektorów DLP. Co więcej, dla każdego z zestawów przygotowano również platformy referencyjne, ułatwiające pracę na wczesnym etapie prototypowania. Zbiornicze informacje o wybranych układach DMD oraz zalecanych dla nich kontrolerach i układach PMIC przedstawiono w **tabeli 1**.

Jak pokazano to w tabeli 1, zestaw DLP LightCrafter Display 2000 (DLP-DLCR2000EVM) został wyposażony w układ DLP2000 o rozdzielczości nHD (640×480), który jest rekomendowany do budowy najtańszych i najbardziej energooszczędnych projektorów mobilnych. Zestaw ten nie ma wbudowanych kontrolerów interfejsów HDMI, VGA czy MHL – na złącza szpilkowe w rozstawie kompatybilnym z komputerami BeagleBone wyprowadzony został komplet sygnałów interfejsu równoległego RGB (24-bitowy RGB wraz z sygnałami HSYNC, VSYNC, CLK oraz DE), linie I<sup>2</sup>C do komunikacji z kontrolerem DLPC2607 oraz linie sterujące pracą modułu – **rysunek 3**.

### Raspberry Pi – interfejs DPI oraz schemat przyłączenia projektora

Wszystkie komputery Raspberry Pi zostały wyposażone w kontroler DPI (z ang. *Display Parallel Interface*) umożliwiający bezpośrednio podłączenie wyświetlaczy LCD/projektorów z interfejsem równoległym RGB, do wyprowadzeń GPIO. W zależności od wybranej konfiguracji, kontroler DPI umożliwia wysterowanie wyświetlaczy

REKLAMA



Rysunek 2. Schemat blokowy typowego projektora DLP (na podstawie: ti.com)

Specjalistyczne szkolenia dla elektroników i automatyków

techdays@techdays.pl  
TECHDAYS.PL

CERTYFIKOWANY PARTNER SZKOLENIOWY



**Tabela 1. Portfolio układów DMD wraz z listą zalecanych kontrolerów (źródło: ti.com)**

	.2" nHD	.2" WVGA	.3" 720p	.47" 1080p
Rozdzielczość	640×360	854×480	1280×720	1920×1080
Wielkość plamki	7.6 μm	5.4 μm	5.4 μm	5.4 μm
Jasność (w lumenach)	20-30	50-100	200-300	500+
Maks. przekątna obrazu w dobrze oświetlonym pomieszczeniu	15-20"	20-30"	35-45"	50-60"
Maks. przekątna obrazu w słabo oświetlonym pomieszczeniu	40"+	50"+	60"+	70"+
Układ DMD	DLP2000	DLP2010	DLP3010	DLP4710
Kontroler DMD	DLPC2607	DLPC3430	DLPC3433	DLPC3439
Układ PMIC	DLPA1000	DLPA2000	DLPA2005	DLPA3000
Zestaw ewaluacyjny	DLPDLCR2000EVM	DLPDLCR2010EVM	DLPDLCR3010EVM	DLPDLCR4710EVM

pracujących w formacie RGB888/RGB24 (po 8 bitów na kolor), RGB666 (po 6 bitów na kolor) oraz RGB565 (5 linii dla koloru czerwonego i niebieskiego oraz 6 linii dla koloru zielonego). Wszystkie sygnały kontrolera (sygnały R, G, B, HSYNC, VSYNC, CLK oraz DE) zostały wyprowadzone na 40-pinowe złącze rozszerzeń, po uprzedniej konfiguracji wyprowadzeń GPIO do pełnienia alternatywnej funkcji ALT2. Przypisanie poszczególnych linii interfejsu DPI do wyprowadzeń GPIO zostało przedstawione w tabeli 2.

Ze względu na możliwość konfiguracji kontrolera do pracy w trybach RGB888, RGB666 oraz RGB565, poszczególne wyprowadzenia GPIO są mapowane do funkcji linii R[x], G[x] oraz B[x] zgodnie z informacjami z tabeli 3.

Czas przystąpić do konfigurowania komputera Raspberry Pi, którą rozpoczynamy od pobrania ze strony producenta ostatniej wersji dystrybucji Raspbian [1] (będącej modyfikacją Debiana dla komputerów Raspberry Pi) oraz wgrania obrazu systemu na kartę SD. Następnym krokiem jest konfiguracja wyprowadzeń GPIO do pełnienia alternatywnej funkcji ALT2, a więc funkcji przypisanej do kontrolera DPI. Konfiguracja ta zostanie przeprowadzona z wykorzystaniem pliku konfiguracyjnego *config.txt* – umieszczonego na partycji */boot* w obrazie systemu – oraz predefiniowanych warstw opisu *Device Tree* (dostępnych również na partycji */boot* w katalogu *overlays*). Warstwy opisu *Device Tree* ułatwiają organizację wsparcia dla wielu konfiguracji

**Tabela 2. Przypisanie linii interfejsu DPI do wyprowadzeń GPIO**

Wyprowadzenie GPIO	Funkcja (konfiguracja ALT2)	Wyprowadzenie GPIO	Funkcja (konfiguracja ALT2)
GPIO 0	PCLK	GPIO 14	DPI_D10
GPIO 1	DE	GPIO 15	DPI_D11
GPIO 2	VSYNC	GPIO 16	DPI_D12
GPIO 3	HSYNC	GPIO 17	DPI_D13
GPIO 4	DPI_D0	GPIO 18	DPI_D14
GPIO 5	DPI_D1	GPIO 19	DPI_D15
GPIO 6	DPI_D2	GPIO 20	DPI_D16
GPIO 7	DPI_D3	GPIO 21	DPI_D17
GPIO 8	DPI_D4	GPIO 22	DPI_D18
GPIO 9	DPI_D5	GPIO 23	DPI_D19
GPIO 10	DPI_D6	GPIO 24	DPI_D20
GPIO 11	DPI_D7	GPIO 25	DPI_D21
GPIO 12	DPI_D8	GPIO 26	DPI_D22
GPIO 13	DPI_D9	GPIO 27	DPI_D23

sprzętowych z wykorzystaniem tego samego jądra systemu. Listę przygotowanych przez producenta warstw wraz z ich opisem i listą przyjmowanych argumentów można znaleźć w pliku */boot/overlays/README*. Wśród plików z rozszerzeniem *\*.dtbo* (a więc skompilowanych warstw opisu *Device Tree*), w katalogu */boot/overlays* możemy odszukać również pliki *dpi18.dtbo* oraz *dpi24.dtbo*, które zgodnie z zawartym w *README* opisem, konfiguruje wyprowadzenia GPIO do pracy w trybie DPI z magistralą 18- lub 24-bitową. Jak możemy szybko przeliczyć, konfiguracja magistrali w trybie 24-bitowym pochłonie wszystkie 28 wyprowadzeń GPIO (24 linie RGB oraz cztery sygnały VSYNC, HSYNC, CLK i DE) – nie

zostawiając tym samym możliwości komunikacji z kontrolerem DLPC2607 po magistrali I<sup>2</sup>C. W tym miejscu użytkownik może zdecydować się na wykorzystanie konwerterów sygnałów USB ↔ I<sup>2</sup>C i pełnej 24-bitowej konfiguracji lub skonfigurować interfejs DPI do pracy w trybie 18-bitowym. Włączenie wybranej warstwy *Device Tree* jest realizowane poprzez umieszczenie w pliku */boot/config.txt* wpisu:

*dtoverlay=<nazwa warstwy DTBO>*

Dla omawianej konfiguracji 18-bitowej:

*dtoverlay=dpi18*

Podczas konfigurowania *Device Tree* należy upewnić się, że skonfigurowana warstwa nie koliduje z inną, uprzednio wykonaną konfiguracją sprzętu. W przypadku

**Tabela 3. Przypisanie linii interfejsu DPI do wyprowadzeń GPIO (na podstawie: raspberrypi.org)**

Tryb	RGB	Numer wyprowadzenia GPIO																											
		27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4				
1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
2	565	-	-	-	-	-	-	-	-	7	6	5	4	3	7	6	5	4	3	2	7	6	5	4	3	-	-	-	-
3	565	-	-	-	7	6	5	4	3	-	-	7	6	5	4	3	2	-	-	-	7	6	5	4	3	-	-	-	-
4	565	-	-	7	6	5	4	3	-	-	-	7	6	5	4	3	2	-	-	7	6	5	4	3	-	-	-	-	-
5	666	-	-	-	-	-	-	7	6	5	4	3	2	7	6	5	4	3	2	7	6	5	4	3	2	-	-	-	-
6	666	-	-	-	-	-	-	-	-	-	-	7	6	5	4	3	2	-	-	7	6	5	4	3	2	-	-	-	-
7	888	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0	7	6	5	4

Tabela 4. Parametry konfiguracji warstwy i2c-gpio

Format zapisu: dtoverlay=i2c-gpio,<parametr>=<wartość>	
Parametr	Opis
i2c_gpio_sda	Numer wyprowadzenia GPIO dla linii SDA (domyślnie "23")
i2c_gpio_scl	Numer wyprowadzenia GPIO dla linii SCL (domyślnie "24")
i2c_gpio_delay_us	Opóźnienie zegara wyrażone w mikrosekundach (domyślnie "2" = 100 kHz)

Tabela 5. Parametry konfiguracji wyświetlacza/projektora dla wpisu hdmi\_timings

hdmi_timings=<parametr1> <parametr2>	
<h_active_pixels>	Szerokość ekranu wyrażona w pikselach
<h_sync_polarity>	Polaryzacja sygnału synchronizacji pionowej
<h_front_porch>	Opóźnienie sygnału HSYNC (przed sygnałem DE)
<h_sync_pulse>	Szerokość sygnału HSYNC
<h_back_porch>	Opóźnienie sygnału HSYNC (po sygnale DE)
<v_active_lines>	Wysokość ekranu wyrażona w pikselach
<v_sync_polarity>	Polaryzacja sygnału synchronizacji poziomej
<v_front_porch>	Opóźnienie sygnału VSYNC (przed sygnałem DE)
<v_sync_pulse>	Szerokość sygnału VSYNC
<v_back_porch>	Opóźnienie sygnału VSYNC (po sygnale DE)
<v_sync_offset_a>	<domyślnie 0>
<v_sync_offset_b>	<domyślnie 0>
<pixel_rep>	<domyślnie 0>
<frame_rate>	Odświeżanie ekranu wyrażone w Hz
<interlaced>	<domyślnie 0>
<pixel_freq>	Częstotliwość zegara
<aspect_ratio>	Proporcje obrazu: HDMI_ASPECT_4_3 = 1 HDMI_ASPECT_14_9 = 2 HDMI_ASPECT_16_9 = 3 HDMI_ASPECT_5_4 = 4 HDMI_ASPECT_16_10 = 5 HDMI_ASPECT_15_9 = 6 HDMI_ASPECT_21_9 = 7 HDMI_ASPECT_64_27 = 8

warstw *dpi18* oraz *dpi24* należy wyłączyć sprzętowe kontrolery magistral I<sup>2</sup>C oraz SPI (o ile zostały one uprzednio włączone):

```
dtparam=i2c_arm=off
```

```
dtparam=spi=off
```

lub

```
#dtparam=i2s=on
```

```
#dtparam=spi=on
```

Wiadomo już zatem, że linie interfejsu DPI kolidują z wyprowadzeniami sprzętowego

kontrolera magistrali I<sup>2</sup>C. Aby jednak zachować możliwość konfiguracji projektora, na niewykorzystanych dotychczas wyprowadzeniach GPIO23 oraz GPIO24 włączamy programowy kontroler interfejsu I<sup>2</sup>C. Do tego celu wykorzystamy warstwę *i2c-gpio*, która konfiguruje wskazane wyprowadzenia do pracy w trybie linii SDA oraz SCL, obsługiwanych w sposób czysto programowy. Opcje konfiguracji warstwy *i2c-gpio* umieszczono w **tabeli 4**.

Listing 1. Opis Device Tree dla modułu projektora DLP i komputera BeagleBone display-timings

```
{
    native-mode = <timing0>;
    timing0: 864x480
    {
        clock-frequency = <26000000>;
        hactive = <640>;
        hvactive = <480>;
        hfront-porch = <14>;
        hback-porch = <12>;
        hsync-len = <4>;
        vback-porch = <0>;
        vfront-porch = <2>;
        vsync-len = <3>;
        hsync-active = <0>;
        vsync-active = <0>;
    };
};
```

Zgodnie z informacjami przedstawionymi w tabeli 4, do pliku */boot/config.txt* dopisujemy zatem kolejną linię konfiguracji: `dtoverlay=i2c-gpio,i2c_gpio_sda=23,i2c_gpio_scl=24,i2c_gpio_delay_us=2`

Zanim przystąpimy do wykonania połączeń sprzętowych, niezbędnym etapem jest konfiguracja parametrów pracy kontrolera DPI. Korzystając z dokumentacji interfejsu DPI, udostępnionej pod adresem:

<http://bit.ly/2M9gcG>

<http://bit.ly/2Lk6qFO>

oraz opisującej konfigurację parametrów wideo, w pliku konfiguracyjnym */boot/config.txt* umieszczamy wpisy:

```
enable_dpi_lcd=1
```

```
display_default_lcd=1
```

umożliwiający dołączenie zewnętrznego wyświetlacza do kontrolera DPI oraz wskazujący, że wyświetlacz ten jest domyślny dla systemu (w przeciwnym razie funkcję tę pełni ekran podłączony do interfejsu HDMI). Następnie, do parametrów *dpi\_group* oraz *dpi\_mode* przypisujemy odpowiednio wartości 2 i 87, co pozwoli nam na samodzielną konfigurację parametrów pracy wyświetlacza poprzez pole *hdmi\_timings* (w innych przypadkach pola *dpi\_group* oraz *dpi\_mode* umożliwiają wybór predefiniowanych konfiguracji):

```
dpi_group=2
```

```
dpi_mode=87
```

Przedostatnim krokiem w edycji pliku */boot/config.txt* jest zatem określenie

REKLAMA

Listing 2. Plik /boot/config.txt z konfiguracją dla modułu LightCrafter Display 2000

```
# uncomment this if your display has a black border of unused pixels visible
# and your display can output without overscan
disable_overscan=1
# Enable audio (loads snd_bcm2835)
dtparam=audio=on
# Add support for software i2c on gpio pins
dtoverlay=i2c-gpio,i2c_gpio_sda=23,i2c_gpio_scl=24,i2c_gpio_delay_us=2
# DPI Video Setup
dtoverlay=dpi18
overscan_left=0
overscan_right=0
overscan_top=0
overscan_bottom=0
framebuffer_width=640
framebuffer_height=360
enable_dpi_lcd=1
display_default_lcd=1
dpi_group=2
dpi_mode=87
dpi_output_format=458773
hdmi_timings=640 0 14 4 12 360 0 2 3 9 0 0 0 60 0 32000000 3
```

Specjalistyczne szkolenia dla elektroników i automatyków

STM32

TECHDAYS

techdays@techdays.pl  
TECHDAYS.PL

CERTYFIKOWANY PARTNER SZKOLENIOWY

Tabela 6. Konfiguracja parametru dpi\_output\_format

Parametr dpi_output_format		
Bity	Funkcja	Wartość
0:3	Format wyjściowy (patrz Tabela 3)	1 = DPI_OUTPUT_FORMAT_9BIT_666 2 = DPI_OUTPUT_FORMAT_16BIT_565_CFG1 3 = DPI_OUTPUT_FORMAT_16BIT_565_CFG2 4 = DPI_OUTPUT_FORMAT_16BIT_565_CFG3 5 = DPI_OUTPUT_FORMAT_18BIT_666_CFG1 6 = DPI_OUTPUT_FORMAT_18BIT_666_CFG2 7 = DPI_OUTPUT_FORMAT_24BIT_888
4:7	Kolejność RGB	1 = DPI_RGB_ORDER_RGB 2 = DPI_RGB_ORDER_BGR 3 = DPI_RGB_ORDER_GRB 4 = DPI_RGB_ORDER_BRG
8	Konfiguracja DE	0 = DPI_OUTPUT_ENABLE_MODE_DATA_VALID 1 = DPI_OUTPUT_ENABLE_MODE_COMBINED_SYNCS
9	Polaryzacja zegara	0 = Zmiana sygnałów RGB na zboczu rosnącym 1 = Zmiana sygnałów RGB na zboczu opadającym
10	Wyłączenie HSYNC	0 = Sygnał włączony 1 = Sygnał wyłączony
11	Wyłączenie VSYNC	0 = Sygnał włączony 1 = Sygnał wyłączony
12	Wyłączenie DE	0 = Sygnał włączony 1 = Sygnał wyłączony
13	Polaryzacja HSYNC	0 = Domyślna dla trybu hdmi 1 = Odwrócona
14	Polaryzacja VSYNC	0 = Domyślna dla trybu hdmi 1 = Odwrócona
15	Polaryzacja DE	0 = Domyślna dla trybu hdmi 1 = Odwrócona
16	Zbocze sygnału HSYNC	0 = DPI_PHASE_POSEDGE 1 = DPI_PHASE_NEGEDGE
17	Zbocze sygnału VSYNC	0 = DPI_PHASE_POSEDGE 1 = DPI_PHASE_NEGEDGE
18	Zbocze sygnału DE	0 = DPI_PHASE_POSEDGE 1 = DPI_PHASE_NEGEDGE

parametrów pracy wyświetlacza/projektora poprzez umieszczenie wpisu `hdmi_timings`, którego pełny format został przedstawiony w tabeli 5.

Jednym z najprostszych sposobów ustalenia prawidłowych wartości dla wpisu `hdmi_timings` – bez potrzeby żmudnego analizowania dokumentacji modułu projektora – jest wykorzystanie gotowego fragmentu opisu *Device Tree* dla komputera BeagleBone (który w oficjalnych obrazach systemu zawiera wsparcie dla zestawu DLP LightCrafter Display 2000). Fragment opisu *Device Tree* dotyczący projektora DLP został przedstawiony na listingu 1.

Bazując na danych zawartych na listingu 1, w prosty sposób możemy uzupełnić plik `/boot/config.txt` o wpis konfigurację `hdmi_timings`:

```
hdmi_timings=640 0 14 4 12 360 0
2 3 9 0 0 0 60 0 32000000 3
```

Ostatnim krokiem konfiguracji Raspberry Pi jest określenie trybu pracy interfejsu DPI – tryb pracy RGB565/RGB666/RGB888, włączenie/wyłączenie sygnałów polaryzacji pionowej i poziomej, ustalenie polaryzacji sygnałów HSYNC/VSYNC, itd. Konfiguracja ta jest realizowana poprzez

zapis do parametru `dpi_output_format` wartości liczbowej, wyliczonej na podstawie konfiguracji poszczególnych pól bitowych, jak przedstawiono w tabeli 6.

Na podstawie tabeli 6, do pliku `/boot/config.txt` dopisano konfigurację parametru `dpi_output_format`, przypisując mu wyliczoną wartość 458773:

```
dpi_output_format=458773
```

Finalna postać pliku `/boot/config.txt` została przedstawiona na listingu 2.

Po zakończonej konfiguracji programowej można przystąpić do wykonania połączeń sprzętowych. Na bazie informacji zawartych na rysunku 3 oraz w tabeli 3 (dla trybu pracy RGB666), wykonujemy zestaw połączeń w sposób pokazany na rysunku 4.

Po włączeniu zasilania moduł projektora wyświetli wyłącznie domyślny ekran startowy (obraz ten może zostać nadpisany grafiką zdefiniowaną przez użytkownika). Aby wyświetlić obraz generowany przez komputer Raspberry Pi, należy nawiązać za pomocą magistrali I<sup>2</sup>C komunikację z układem kontrolera DLPC2601, a następnie do rejestru 0x0B (rejestr *Input Source Selection*) wpisać wartość 0, która jako źródło sygnału wybiera interfejs równoległy RGB. Dostępność kontrolera DLPC2601 (o adresie sprzętowym 0x1B) na magistrali I<sup>2</sup>C możemy sprawdzić poprzez wykonanie skanowania za pomocą narzędzia `i2cdetect`, np.:

```
~# i2cdetect -y 0
      0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
50:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
70:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
```

Wybór źródła sygnału i ustawienie rozdzielczości wejściowej:

```
i2cset -y 0 0x1b 0x0b 0x00 0x00
0x00 0x00 i
i2cset -y 0 0x1b 0x0c 0x00 0x00
0x00 0x07 i
```

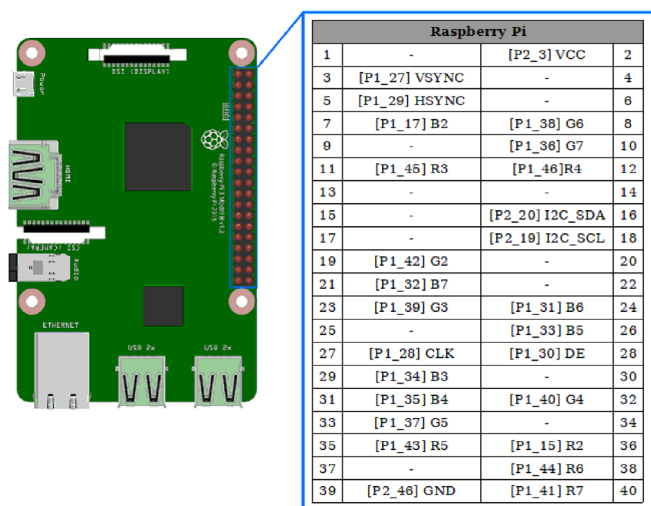
Powyższe komendy można umieścić np. w pliku `/etc/rc.local`, tak aby wybór źródła sygnału odbywał się automatycznie po włączeniu Raspberry Pi.

Kompletny zbiór rejestrów i komend umożliwiających m.in. skalowanie i obracanie obrazu, wybór źródła sygnału, wyświetlanie obrazów testowych, został kompleksowo opisany w dokumencie *DLPC2601 and DLPC2607 Software Programmer's Guide* [2].

Przykład działania modułu DLP LightCrafter Display 2000 podłączonego do komputera jednopłytkowego Raspberry Pi został przedstawiony na filmie: <http://bit.ly/2MwuieY>.

Lukasz Skalski  
contact@lukasz-skalski.com

Bibliografia:  
<http://bit.ly/2wm8Gqv>, <http://bit.ly/2w4z6hi>



Rysunek 4. Schemat połączeń pomiędzy Raspberry Pi 3 a modułem LightCrafter Display 2000