

Czujnik przyśpieszenia (2)

Układ odczytu i wyświetlania

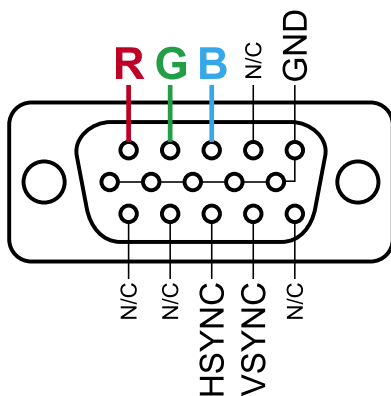


Newton, formułując drugą zasadę dynamiki określał zmianę prędkości w czasie „zmianę ruchu” (mutationem motus). Obecnie używamy pojęcia przyśpieszenie i mamy odpowiednie przyrządy do jego pomiaru. Akcelerometry, bo o nich mowa, są elementami stosowanymi w systemach nawigacji lotniczej, ale też w urządzeniach powszechnego użytku, jak telefony komórkowe czy aparaty fotograficzne. W artykule opisano sposób odczytu informacji z akcelerometru o zmierzonym przyśpieszeniu. W pierwszej części przedstawiono realizację interfejsu do komunikacji pomiędzy akcelerometrem Freescale MMA7455L a układem FPGA. W drugiej pokazano, jak wygenerować sygnały sterujące dla monitora VGA, aby wyświetlić na nim odczytane wartości przyśpieszenia.

Interfejs VGA

W standardowym monitorze CRT obraz wyświetla się za pomocą wielu punktów nazywanych pikselami. Piksel składa się z trzech punktów luminoforu o kolorach świecenia czerwonym, zielonym i niebieskim. Gdy odpowiedni punkt zostanie pobudzony strumieniem elektronów, zaczyna emitować światło. Strumień elektronów przesuwa się od lewej strony ekranu do prawej, linijka po linijce, od góry do dołu, i gdy trafia w kolejne piksele powoduje wyświetlanie obrazu na ekranie.

Na **rysunku 10** zamieszczono widok wtyczki kabla monitora VGA z rozmieszczeniem jego sygnałów wejściowych: synchronizacji poziomej HSYNC, synchronizacji pionowej VSYN oraz składowych koloru czerwonego R, zielonego G, niebieskiego B. HSYNC i VSYN sterują prędkością przesuwania się strumienia elektronów, natomiast sygnały R, G, B określają konfigurację wiązek składających się na ów strumień.

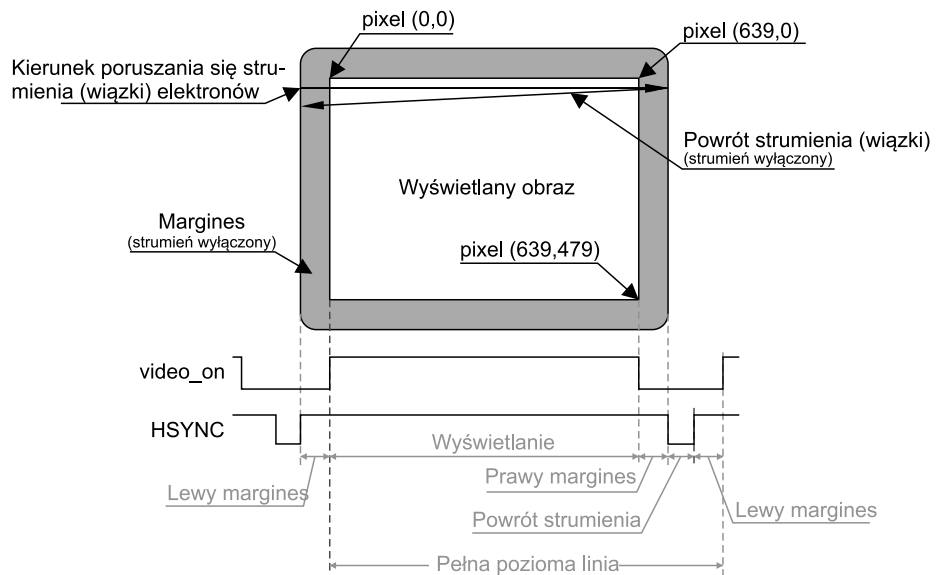


Rysunek 10. Widok rozmieszczenia sygnałów na doprowadzeniach wtyku VGA

Synchronizacja obrazu

Na **rysunku 11** pokazano drogę wiązki elektronów i współzależność sygnałów podczas wyświetlania jednej linijki obrazu. Na ekranie są dwa obszary: obszar z wyświetlanym obrazem oraz margines. Gdy wiązka elektronów przejdzie przez margines, to strumień powinien być wyłączony, co jest kontrolowane przez sygnał *video_on*. Sygnał *video_on* jest sterowany licznikiem, który zlicza piksele podczas przejścia strumienia elektronów od lewej strony ekranu do prawej.

Przy rozdzielczości obrazu 640×480 pikseli, rozdzielczość pozioma wynosząca 640 pikseli oznacza, że część ekranu wyświetlająca obraz będzie podzielona na 640 kolumn. Szerokość lewego marginesu przy tej rozdzielczości ekranu wynosi 44, piksele a prawego 20. Ważny jest też



Rysunek 11. Schemat wyświetlania poziomej linii, synchronizacja pozioma

Dodatkowe materiały na CD i FTP:
<ftp://ep.com.pl>, user: 17855, pass: 4s406qj2
 • pierwsza część kursu

czas przejścia strumienia do nowej linijki (czas powrotu na rys. 11), który odpowiada wyświetleniu 96 pikseli. Można obliczyć, że na długość (w pikselach) jednej pełnej linii ekranu składają się wartości z czterech obszarów, które w sumie dają $640+44+20+96=800$ pikseli.

W chwili ustawienia sygnału *HSYNC* na poziomie wysokim, strumień elektronów zaczyna się przesuwać od lewej strony ekranu do prawej, z krokiem równym czasowi wyświetlania pojedynczego piksela. Licznik zaczyna zliczać impulsy zegara taktującego. Gdy licznik osiągnie wartość określoną jako lewy margines, sygnał *video_on* jest ustawiany na poziomie wysokim, działo elektronowe zaczyna emitować strumień elektronów i następuje wyświetlanie obrazu. Gdy licznik osiągnie wartość $640+44=684$, oznacza to koniec obszaru wyświetlania obrazu i początek prawego marginesu. Sygnał *video_on* zostaje wyzerowany i jest wyłączony strumień elektronów. Po przejściu jeszcze 20 pikseli, czyli dojdzie wiązki do końca ekranu, sygnał *HSYNC* jest zerowany, a wiązka elektronów przechodzi do początku nowej linijki. Gdy licznik osiągnie wartość 800, to jest zerowany i cały proces wyświetlania jednego wiersza rozpoczyna się od początku.

Na rys. 11 pokazano nie tylko prawy i lewy margines, ale również górny i dolny, który jest związany z sygnałem *VSYNC*. Sygnał ten odpo-

Listing 3. Fragment podprogramu synchronizacji poziomej obrazu

```

//////// Rejestry
always @ (posedge CLK_50MHZ, posedge reset)
  if (reset) begin
    hlicznik_reg <= 0;
    hsync_reg <= 1'b0;
  end
  else begin
    hlicznik_reg <= hlicznik_next;
    hsync_reg <= hsync_next;
  end
//////// Blok kontroli licznika
always @ * begin
  if (zegar) // Wykryty impuls zegara powoduje
    if (h_koniec) // jeżeli licznik pełen, to jego wyzerowanie
      hlicznik_next = 0;
    else // jeśli nie to zwiększenie o 1
      hlicznik_next = hlicznik_reg + 1;
    else // w każdym innym wypadku zachowanie wartości
      hlicznik_next = hlicznik_reg;
end
//////// Sygnały sterujące przepełnieniem licznika oraz synchronizacją poziomą
assign h_koniec = (hlicznik_reg == 799); // doszliśmy do końca ekranu
// obszar pomiędzy 640 + 20(rozdzielczość + p. margines) a 640 + 96 + 20 - 1(rozdzielczość + powrót działa + p. margines)
assign hsync_next = (hlicznik_reg >= 660) && (hlicznik_reg <= 755); // impuls synchronizacji poziomej
. . .

//////// Wyprowadzenie na zewnątrz sygnałów i liczników
assign HSYNC = hsync_reg;
assign VSYNC = vsync_reg;
// Gdy znajdujemy się w obszarze aktywnego wyświetlania obrazu
assign video_on = (hlicznik_reg <= ROZDZ_POZIOMA) && (vlicznik_reg <= ROZDZ_PIONOWA);
assign piksel_X = hlicznik_reg;
assign piksel_Y = vlicznik_reg;

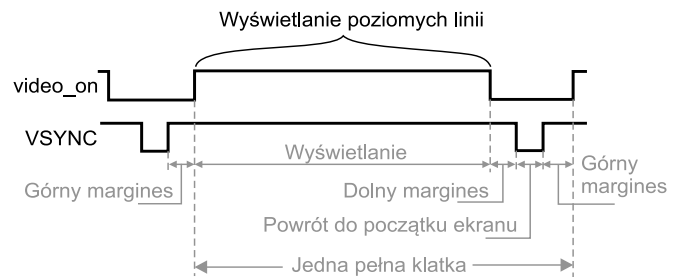
```

wiada za synchronizację pionową i również steruje sygnałem *video_on*. Podobnie jak przy wyświetlaniu pojedynczej linii poziomej, również przy wyświetlaniu całej klatki obrazu składającej się z wielu linii, sygnał *video_on* uruchamia strumień elektronów tylko w obszarze aktywnego wyświetlania obrazu.

W chwili, gdy sygnał *VSYNC* zostanie ustalony na poziomie wysokim, wiązka elektronów jest przemieszczana linijką po linijce od góry ekranu w dół (rysunek 12). Każda zakończona linia powoduje inkrementację licznika zliczającego ich liczbę. Przy założonej rozdzielczości (640×480 pikseli) obszar wyświetlania jest podzielony na 480 linii. Górny margines ma 33 linie, co oznacza, że gdy licznik osiągnie tę wartość, sygnał *video_on* załączy strumień elektronów i rozpocznie się wyświetlanie obrazu. Gdy wiązka dojdzie do dolnej granicy obszaru wyświetlania, to licznik osiągnie wartość 33+480=513. Wówczas sygnał *VSYNC* jest zerowany, a *video_on* wyłącza strumień elektronów. Po kolejnych 10 liniach, stanowiących dolny margines, następuje powrót wiązki elektronowej do początku ekranu, co czasowo odpowiada wyświetleniu 2 linii. Cykl się powtarza, i w taki sam sposób jest wyświetlana kolejna klatka obrazu. W procesie synchronizacji obrazu są stosowane dwa sygnały: *HSYNC* oraz *VSYNC*, które wspólnie określają przemieszczanie wiązki elektronowej przez cały ekran, dlatego są przesyłane bezpośrednio do monitora. Na schematach

oraz w opisie procesu wyświetlania obrazu zastosowano pewien skrót myślowy, a mianowicie, że sygnałem *video_on* sterują *HSYNC* oraz *VSYNC*. W rzeczywistości wszystkim sterują dwa liczniki. Obydwa zliczają impulsy zegara, a charakterystyki czasowe sygnałów synchronizacji na potrzeby projektu zostały odpowiednio przesunięte dla łatwiejszego sterowania ekranem. Fragment podprogramu (w języku Verilog) odpowiedzialnego za synchronizację poziomą umieszczono na **listingu 3**.

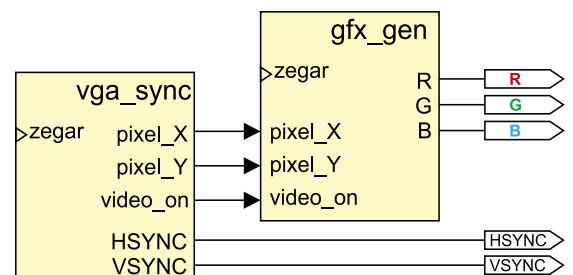
Gdy licznik synchronizacji poziomej (*hlicznik_reg*) ma zawartość w przedziale od 660 do 755 (przedział wartości przeznaczony na przejście wiązki do początku nowej linijki), następuje przesłanie impulsu synchronizacji poziomej za pomocą *hsync*. Analogicznie działa synchronizacja pionowa i licznik przeznaczony dla sygnału *vsync*. Gdy licznik synchronizacji poziomej ma zawartość w zakresie od 0 do 640, a licznik synchronizacji pionowej pomiędzy 0 a 480, sygnał *video_on* jest ustawiany na poziom wysoki, co powoduje włączenie strumienia



Rysunek 12. Schemat wyświetlania 1 klatki obrazu, synchronizacja pionowa

elektronów. Ten sygnał oraz liczniki synchronizujące są używane w module implementowanym w FPGA, wyświetlającym wyniki pomiaru przyspieszenia na ekranie monitora.

Moduł synchronizujący obraz na monitorze powinien być taktowany sygnałem zegarowym o częstotliwości dostosowanej do trybu, w którym ma pracować monitor. W wypadku rozdzielczości 640×480, jak wspomniano wcześniej,



Rysunek 13. Schemat kontrolera VGA

Listing 4. Fragment opisu układu odpowiedzialnego za rysowanie marginesu o szerokości 10 pikseli

```

wire margines;
assign margines = ((piksel_X<=10) || (640-10<=piksel_X) || // Deklaracja sygnału margines
                  (piksel_Y<=10) || (480-10<=piksel_Y)); // Definicja reguły

always @ * begin // Sterowanie wyjściami R, G, B
  if (~video_on) // przy użyciu sygnału margines
    {R,G,B} <= 3'b0; // oraz video_on
  else if (margines)
    {R,G,B} <= 3'b100;
  else
    {R,G,B} <= 3'b0;
end

```

Uwaga!

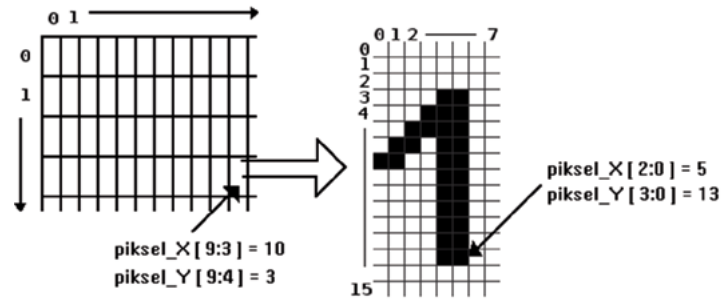
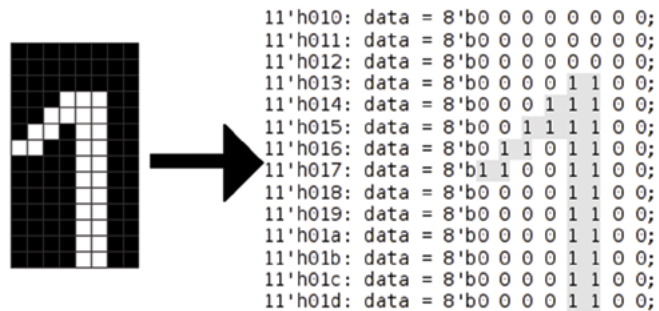
Szerokości marginesów w różnych monitorach mogą się różnić, należy więc poeksperymentować ze swoimi ustawieniami gdyby standardowe nie przyniosły spodziewanego efektu.

linia pozioma ma długość 800 pikseli, natomiast cały ekran jest podzielony na 525 takich linii. Przy częstotliwości wyświetlania obrazu wynoszącej 60 Hz, w czasie jednej sekundy ekran jest odświeżany 60 razy (60 klatek/s). Wymażając te trzy liczby przez siebie otrzymamy liczbę pikseli wyświetlanych w czasie 1 sekundy. Na jej podstawie można określić częstotliwość sygnału taktującego *vga_sync* zastosowanego w module: $800 \times 525 \times 60 = 25200000$. Liczba ta po zaokrągleniu daje częstotliwość sygnału zegarowego wynoszącą 25 MHz.

Inicjowanie wyświetlania pikseli

Jak już wspomniano, moduł synchronizujący *vga_sync* generuje nie tylko sygnały wysyłane do monitora, ale również odpowiednie sygnały sterujące do modułu pozycjonującego grafikę *gfx_gen*. Schemat połączenia tych dwóch modułów pokazano na **rysunku 13**.

Sygnały *R*, *G*, *B* sterują strumieniem elektronów i w zależności od kombinacji ich wartości piksel świeci się w określonym kolorze. Ustawienie wszystkich sygnałów na poziom wysoki spowoduje świecenie się piksela w kolorze białym, natomiast pozostawienie ich wyzerowanych odpowiada kolorowi czarnemu. Inne kolory uzyskuje się poprzez odpowiednią kombinację wartości tych sygnałów. Sygnały *R*, *G*, *B* są generowane w module *gfx_gen* dla każdego piksela. Moduł ten, poprzez sygnały *piksel_X* oraz *piksel_Y*, otrzymuje informacje, o pozycji wiązki elektronowej i w zależności od atrybutów wyświetlanego znaku dobiera odpowiednie wartości sygnałów *R*, *G*, *B*. Sygnały *piksel_X* oraz *piksel_Y* są wyprowadzonymi z modułu *vga_sync* zawartościami liczników synchronizacji poziomej i pionowej. Dla przykładu narysujmy dookoła ekranu czerwony margines, tzn. ramkę o szerokości 10 pikseli.

**Rysunek 14. Adres pola oraz pojedynczego piksela****Rysunek 15. Fragment tablicy opisującej wzór znaku „1”**

Rysowanie tej ramki jest realizowane przez układ, którego opis umieszczono na **listingu 4**.

Najpierw zadeklarowano sygnał sterujący wyświetlaniem marginesu *wire margins*. Przy lokalizacji wiązki elektronowej na współrzędnych:

- pozioma (*piksel_X*) mniejsza od 10 lub większa od 63,
- pionowa (*piksel_Y*) mniejsza od 10 lub większa od 470,

sygnał sterujący będzie na poziomie wysokim. W bloku *always* ustalone są reguły załączania poszczególnych wiązek strumienia elektronów (wyjścia *R*, *G*, *B*). Sygnał *margins* powoduje włączenie strumienia elektronów powodującego wyświetlanie koloru czerwonego. Gdy wiązka znajduje się poza obszarem zdefiniowanym w deklaracji sygnału *margins*, to mogą być zmieniane wartości sygnałów *R*, *G*, *B* sterujące wszystkimi wiązkami.

Wyświetlanie znaków

Aby wyświetlić znaki w odpowiednich miejscach ekranu, należy podzielić go na określoną

liczbę obszarów, a każdy z nich przeznaczyć do wyświetlania elementu z wzorca. Współrzędna danego obszaru odczytuje się z tablicy – wzorca – określającej, które piksele mają świecić się, a które pozostać zgaszone.

Aby wyświetlać znaki alfanumeryczne, obraz na ekranie o rozdzielczości 640×480 pikseli podzielmy na 80 kolumn o szerokości 8 pikseli oraz 30 linii o wysokości 16 pikseli. Uzyskuje się w ten sposób ekran składający się z 2400 prostokątnych pól o wymiarach 8×16 pikseli. Aby jednoznacznie określić pole, w którym chcemy wyświetlić znak, należy użyć sygnałów *piksel_X* oraz *piksel_Y*. Obydwa sygnały są 9-bitowe. Znając rozdzielczość pól ekranu, możemy wyodrębnić z sygnałów *piksel_X* oraz *piksel_Y* sygnały określające kolumny oraz wiersze, w których znajduje się to pole, a także informację o świeceniu (barwie) poszczególnych pikseli w danym polu. Przy szerokości znaku 8 pikseli potrzebujemy 3 bitów dla zaadresowania współrzędnej poziomej piksela danego znaku. Pozostałe 7 bitów określa numer kolumny, w której ma być wy-

Listing 5. Opis układu wyświetlania znaku

```

assign wzorzec = (piksel_X[9:3]== 7'd00 & piksel_Y[8:4]==5'd00) ? 6'h01:// Cyfra „1” w lewym górnym rogu (ze wzorca)
                6'h16; // w każdym innym miejscu ekranu czarne pole (również ze wzorca)
assign adres_linia = piksel_Y[3:0]; // Adres aktualnie wyświetlanego wiersza
assign adres_wzorca = {wzorzec,adres_linia}; // Złożony adres wzorca
assign kolumna = piksel_X[2:0]; // Numer wyświetlanej kolumny
// Moduł tablicy wzorców
tab_wzorcow wzorcet (
    .clk(clk),
    .addr(adres_wzorca), // Adres wzorca żadanego znaku
    .data(wzorzec_linia) // Dane o 8 kolejnych stanach wiązki elektronów
);
assign text_on = wzorzec_linia[7-kolumna]; // Ustawienie wiązki w odpowiedni stan (odczytany ze wzorca)

```

Listing 6. Wyświetlanie wskazania akcelerometru (fragment opisu układu)

```

assign wzorzec =
(piksel_X[9:3]== 7'd1 & piksel_Y[9:4]==6'd1) ? 6'h10: // Do rejestru trafia adres znaku „X”
(piksel_X[9:3]== 7'd2 & piksel_Y[9:4]==6'd1) ? 6'h14: // Adres znaku „:”
(piksel_X[9:3]== 7'd3 & piksel_Y[9:4]==6'd1) ? Wartosc_X[7:4]: // 4 starsze bity (0 na rys. 17)
(piksel_X[9:3]== 7'd4 & piksel_Y[9:4]==6'd1) ? Wartosc_X[3:0]: // 4 młodsze bity (3 na rys. 17)
6'h16;

```

X:03

Rysunek 16. Przykładowy rezultat

światlany znak. Dla wysokości znaku 16 pikseli, należy użyć 4 bitów adresujących linię w znaku oraz 6 bitów na adres linii. Na **rysunku 14** pokazano fragment ekranu podzielonego na pola jednakowej wielkości.

Należy zauważyć, że 7 bardziej znaczących bitów sygnału *piksel_X* oraz 6 bardziej znaczących bitów sygnału *piksel_Y* jest używanych do zaadresowania pola. Następnie, w obrębie wybranego pola, współrzędne pikseli są określane pozostałymi bitami z obydwu sygnałów. Aby wyświetlić np. cyfrę „1” w lewym górnym rogu ekranu, gdy sygnały *piksel_X* oraz *piksel_Y* mają wartość tego pola, musimy zastosować bity pikseli (3 mniej znaczące bity *piksel_X* oraz 4 mniej znaczące bity *piksel_Y*) do odczytania z tablicy wartości, jak należy wysterować działa elektronowe. Tablica zawierająca wzorce poszczególnych znaków jest macierzą składającą się z 8 kolumn. Liczba wierszy w tej macierzy jest wielokrotnością liczby 16 i jest zależna od liczby znaków, która się w niej znajduje. Fragment takiej tablicy, będący wzorcem znaku „1”, pokazano na **rysunku 15**.

W każdym wierszu i każdej kolumnie tablicy są zawarte wartości 0 lub 1, które sterują stru-

mieniem działa elektronowego (katody). Do wyświetlenia danego wzorca z tablicy, należy użyć aktualnej wartości sygnałów *piksel_X* i *piksel_Y* oraz adresu wzorca, który chcemy wyświetlić. Adres wzorca składa się z 6 bitów, które wraz z sygnałem *piksel_Y* określają wiersz w tablicy wzorców, z którego następnie, na podstawie sygnału *piksel_X*, są odczytywane wartości wzoru znaku określające stany wiązki elektronów. Jeżeli wartość bitu odczytanego z tablicy wynosi 0, to strumień elektronów będzie wyłączony, a jeżeli 1, to załączony.

Procedura z **listingu 5** spowoduje wyświetlenie znaku znajdującego się w tablicy wzorców pod adresem zapisanym szesnastkowo jako 6'h01. Wartość ta, połączona z numerem wiersza, zostaje użyta do odczytania wzorca odpowiadającego aktualnie wyświetlanej linii. Rejestr *wzorzec_linia* zawiera 8 bitów (jeden wiersz pola), spośród których na podstawie części sygnału *piksel_X* adresującej piksel w polu, wybierany jest bit, aktywujący działa elektronowe. Gdy współrzędne *piksel_X* oraz *piksel_Y* adresujące pole będą poza obszarem pola znajdującego się w lewym górnym rogu, jako wzorzec do wyświetlania ustawiane jest puste pole. W rezultacie otrzymamy czarny ekran z wyświetloną cyfrą „1” w pierwszym polu (numer 0,0).

Aby wyświetlić wskazania akcelerometru, wystarczy do rejestru *wzorzec* „wpisać” zawartość rejestru np. *Wartosc_X*, pamiętając jednak,

że *Wartosc_X* ma 8 bitów, co oznacza liczbę dwucyfrową. Należy więc zarezerwować pola na ekranie: jedno dla 4 bardziej znaczących bitów, drugie dla 4 mniej znaczących. Na **listingu 6** przedstawiono fragment opisu układu umieszczającego w lewym górnym rogu (linijka nr 1, pola 1, 2, 3 oraz 4) informację o odczytanej wartości przyspieszenia w osi X, co pokazano na **rysunku 16**.

Podsumowanie

Jak to czasami bywa z różnymi urządzeniami, w trakcie eksploatacji okazało się, że wystąpiły problemy przy dołączeniu opisywanego układu do projektora multimedialnego. Okazało się, że on nie wyświetla żadnego obrazu. Było to zastanawiające, ponieważ zwykły monitor nie miał żadnych problemów z rozpoznaniem sygnałów sterujących i wyświetleniem prawidłowego obrazu. Jak się okazało, problemy ze współpracą z rzutnikiem wynikały stąd, że fragment obrazu był wyświetlany w obszarze marginesu, w którym działa elektronowe powinno być wyłączone. W monitorze nie było problemu, ale elektronika sterująca projektorem nie radziła sobie z takim sygnałem. Po drobnej zmianie szerokości marginesów i obszaru aktywnego wyświetlania rozwiązano problem.

Christyan Ruminowicz
Piotr Pietrzyk

R	E	K	L	A	M	A
						
<h2>NAJWIĘKSZY PRODUCENT MODUŁÓW LASEROWYCH W POLSCE OFERUJE:</h2>						
<p>Całe spektrum wskaźników laserowych:</p> <ul style="list-style-type: none"> - proste pointery, - przemysłowe generatory linii, generatory krzyża, - profesjonalne laserowe systemy pomiarowe <p>Realizujemy zamówienia niestandardowe; tworząc rozwiązania wg indywidualnych potrzeb klienta.</p>						
<p>Oprócz produkcji modułów zajmujemy się sprzedażą całej gamy produktów z zakresu optoelektroniki:</p> <ul style="list-style-type: none"> - diody laserowe, diody LED, soczewki, płytki szklane, - elementy optomechaniczne, filtry, okulary ochronne, - okna i kurtyny ochronne, sterowniki laserowe, - spektrometry, oświetlacze IR, moduły Peltiera, - wizualizatory podczerwieni. 						
<p>NOWOŚĆ w naszej ofercie:</p> <h3>Laserowy Czujnik Położenia</h3> <p>wyróżnienie jury podczas Międzynarodowych Targów Optoelektroniki i Fotoniki</p> 						
 <p>ul. Zwoleńska 43/43a, 04 - 761 Warszawa tel. 022 615 73 71, 022 615 64 31 info@semicon.com.pl, www.semicon.com.pl</p>						