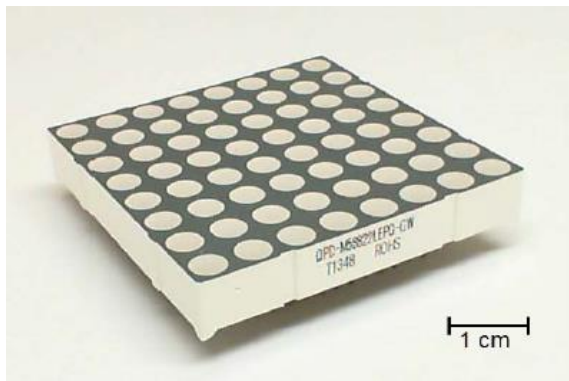


29. Matryca - LED

(LED Matrix)

Wykonaj Projekt: Matryca LED.



Rysunek 29-1: Matryca LED, z EmbeddedSystemsKit dla NiMyRio.

Matryca LED, pokazana na Rysunku 29-1; to regularnie rozłożone LED w rastrze 8 x 8. Zastosowano zielone i czerwone LED, w każdym oczku *połu* w kształcie koła, tej siatki wmontowano zieloną i czerwoną LED. Ta matryca LED ma połączone wiersze i kolumny podobnie jak klawiatura – jej *styki* skojarzone z przyciskami, z Rozdziału 25. Podobny jest, zatem schemat okablowania matrycy LED i klawiatury. Składa się on z dwóch linii kolumn - jedna linia dla każdej LED oraz wspólnej linii wiersza.

Cele nauczania: po starannym wykonaniu zalecanych w tym rozdziale działań z pewnością potrafisz:

- 1) Opisać schemat matrycy LED,
- 2) Wyjaśnić technikę wyświetlania rastra do wyświetlania dowolnych wzorów,
- 3) Opisać potrzebę przemian aktywacji zielonych i czerwonych LED,
- 4) Zaprojektować sekwencje ciekawych animacji.

29-1. Pokazy

Wykonaj kolejne czynności: wiodące do pokazu prawidłowego działania wykonanego interfejsu: matryca LED-NiMyRio.

Wybierz ze zbioru elementów StarterKit dla NiMyRio, następujące składniki interfejsu:

- Matryca LED,

<http://www.sparkfun.com/datasheets/Components/SanYoung-Medium-RG.pdf>

- UPM Uniwersalną Płytkę Montażową (2 szt.)
- PP Przewody Połączeniowe M-F (24 szt.)

Zbuduj obwód interfejsu: Zapoznaj się ze schematem przedstawionym na Rysunku 29-2, zalecanym układem połączeń i przedstawionym na Rysunku 29-3; układem wyprowadzeń końcówek wymaganego złącza.

PORADA: Skorzystaj z kodu kolorów do oznaczania rezystorów, dla odróżniania kabli połączeń DIO – matryca LED. Na przykład: czarny (0) dla B/DIO0, brązowy (1) dla B/DIO1, i tak dalej.

Obwód interfejsu wymaga ośmiu połączeń do złącza A NiMyRioMXP i szesnastu połączeń do złącza B (Rysunek A-1):

- 1) Wiersz 1 (1) → A/DIO0 (pin 11)
- 2) Wiersz 2 (2) → A/DIO1 (pin 13)
- 3) Wiersz 3 (3) → A/DIO2 (pin 15)
- 4) Wiersz 4 (4) → A/DIO3 (pin 17)
- 5) Wiersz 5 (21) → A/DIO4 (pin 19)
- 6) Wiersz 6 (22) → A/DIO5 (pin 21)
- 7) Wiersz 7 (23) → A/DIO6 (pin 23)
- 8) Wiersz 8 (24) → A/DIO7 (pin 25)
- 9) Kolumna 1 zielony (5) → B/DIO0 (pin 11)
- 10) Kolumna 2 zielony (6) → B/DIO1 (pin 13)
- 11) Kolumna 3 zielony (7) → B/DIO2 (pin 15)
- 12) Kolumna 4 zielony (8) → B/DIO3 (pin 17)
- 13) Kolumna 5 zielony (9) → B/DIO4 (pin 19)
- 14) Kolumna 6 zielony (10) → B/DIO5 (pin 21)
- 15) Kolumna 7 zielony (11) → B/DIO6 (pin 23)
- 16) Kolumna 8 zielony (12) → B/DIO7 (pin 25)
- 17) Kolumna 1 czerwony (20) → B/DIO8 (pin 27)
- 18) Kolumna 2 czerwony (19) → B/DIO9 (pin 29)
- 19) Kolumna 3 czerwony (18) → B/DIO10 (pin 31)
- 20) Kolumna 4 czerwony (17) → B/DIO11 (pin 18)
- 21) Kolumna 5 czerwony (16) → B/DIO12 (pin 22)
- 22) Kolumna 6 czerwony (15) → B/DIO13 (pin 26)
- 23) Kolumna 7 czerwony (14) → B/DIO14 (pin 32)
- 24) Kolumna 8 czerwony (13) → B/DIO15 (pin 34)

WSKAZÓWKA: Za krótkie przewody połączeniowe? Można podłączyć podzbiór tablicy, na przykład, od 1 do 4 wierszy i kolumn 1 do 4 na zielonym ekranie, który wymaga jedynie tylko ośmiu przewodów połączeniowych.

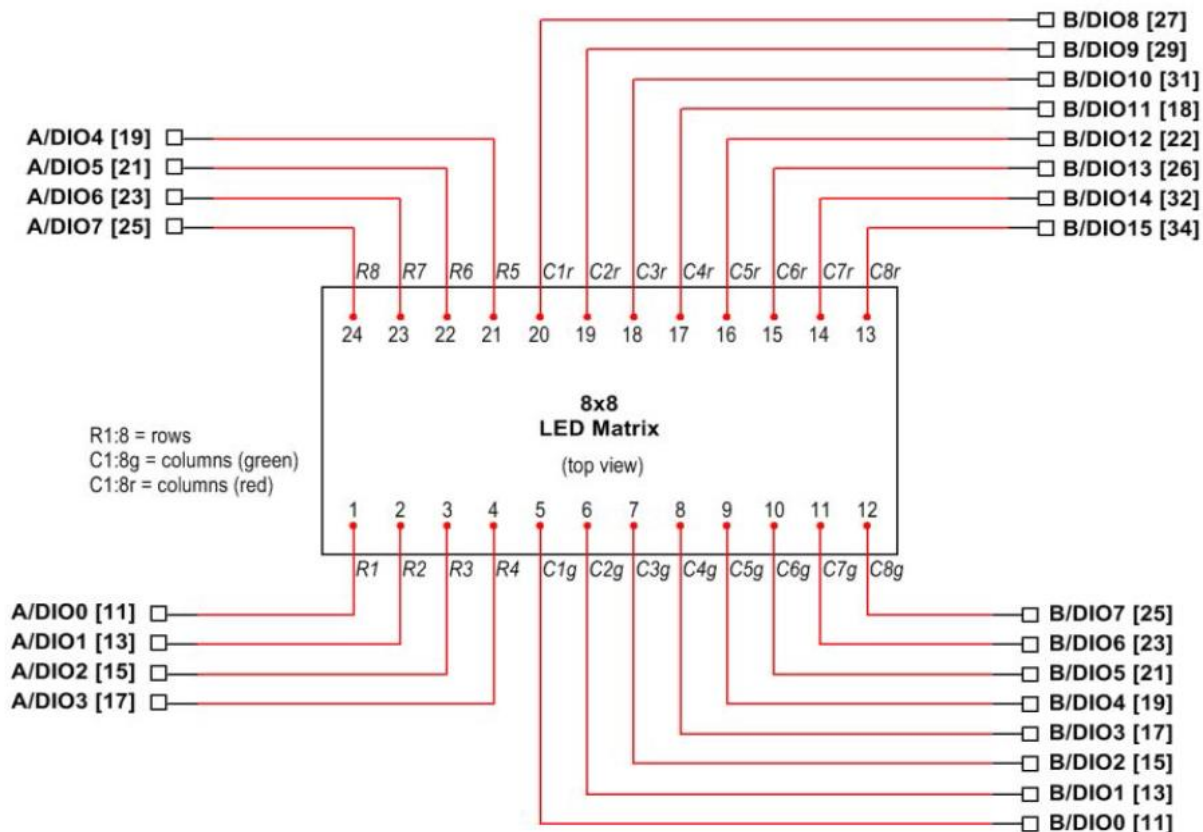
Uruchom pokaz VI:

- **Pobierz:** <http://www.ni.com/academic/mrio/project-guide-vis.zip>, jeśli tego nie zrobiłeś wcześniej, to rozpakuj pobraną zawartość w dogodnej lokalizacji swojego komputera.
- **Otwórz Projekt:** *Discrete LED demo.hproj*; zawarty w podkatalogu: *Discrete LED demo*,

- **Rozwiń przycisk hierarchii:** (znak plus), dla obiektu *myRIO*, następnie podwójnym kliknięciem otwórz: *Main.vi*.
- **Upewnij się, że:** *NiMyRio* jest podłączone do komputera.
- **Uruchom VI:** klikając przycisk: *Run* na pasku narzędzi lub naciskając kombinację klawiszy: $\langle \text{Ctrl} + \text{R} \rangle$.

każdy wiersz i wszystkie kolumny, upewnij się, że okablowanie jest prawidłowe!

Powtórz to samo, wyzerowując elementy z czerwonej tablicy 2-D. Zielone i czerwone wskaźniki po wystereowaniu tablic pokazują stan 8 x 8 2-D tablic logicznych, jako jedną 64-bitową, w wartościach szesnastkowych.



Rysunek 29-2: Układ pokazowy Projekt: Matryca LED; schemat ideowy, proponowane połączenia 24 linii I/O do końcówek złącza A i B NiMyRio.

- **Spodziewaj się okna:** *Deployment Process* (Proces wdrażania) w nim przed startem VI, zobaczysz, w jaki sposób Projekt kompiluje i instaluje (pliki do pobrania) do *NiMyRio*.

UWAGA: Możesz chcieć wybrać opcję:

Close on successful completion,

(Zamknij po ukończeniu),

opcja ta wymusi na VI start automatyczny.

Oczekiwane rezultaty: VI rozpocznie prace w trybie *AutomaticPattern*, wyświetlacz powinien pokazać przetworzoną zbieżną kwadratową animację, odświeżającą zapis, co pół sekundy. Przesuń suwak wskaźnika czasu wzoru pętli, aby dobrać optymalnie szybkość animacji. Czy zauważasz coś niezwykłego dla czasów pętli poniżej 100 ms?

Kliknij przełącznik wzoru *AutomaticPattern*, a następnie kliknij przycisk na zielonej tablicy sterowania 2-D. Powiniście zobaczyć odpowiednio zieloną LED, włączoną na matrycy. Poświęć chwilę, aby sprawdzić

Wybierz: Edit | Reinitialize Values to Default *ponowne inicjalizowanie wartości domyślnych*, aby *wyzerować* wszystkie 64 przyciski tablicy. Następnie kliknij prawy dolny przycisk. Zwróć uwagę, że odnosi się on do najmniej znaczącego bitu wartości 64-bitowej. Kliknij przycisk lewy górny; jest to najbardziej znaczący bit.

Kliknij więcej przycisków, aby określić kolejność pozostałych bitów. Te 64-bitowe kody wzorów zapewniają wygodny sposób tworzenia własnych wzorów wyświetlania i mogą być wykorzystywane do animacji.

Kliknij przycisk: *Stop* lub wybierz z klawiatury komputera przycisk $\langle \text{Esc} \rangle$, aby zatrzymać VI i zresetować *NiMyRio*; *reset* spowoduje powrót *NiMyRio* do trybu początkowego, czyli ustawień początkowych. W stanie *reset*, do pamięci układu nie muszą być wpisane same zera lub same jedynki w rejestrach, *reset* - to powrót układu do stanu początkowego.

Wskazówki dotyczące rozwiązywania problemów: nie widzisz oczekiwanych rezultatów? Potwierdź prawdziwość poniższych zdarzeń:

- LED wskazująca poprawność zasilania w NiMyRio świeci jaskrawym światłem,
- Przycisk *Run*, na pasku narzędzi jest czarny, co oznacza, że **VI** jest w *RunMode* - trybie pracy,
- Że okablowanie jest wykonane prawidłowo, a styki zapewniają dobry kontakt elektryczny,

29.2. Teoria interfejsu

Obwód interfejsu:

W każdym okrągłym polu matrycy LED, o rozmiarach 8 x 8 pól, zawiera zieloną i czerwoną LED. Katody każdej pary LED łączą się w tym samym przewodzie,

pewnia indywidualne sterowanie dwóch LED w każdym punkcie (polu). Aktywacja obu LED może dać kolor mieszany: *żółty*.

Uważnie przestudiuj wideo:

LEDMatrix Theory (09:50)

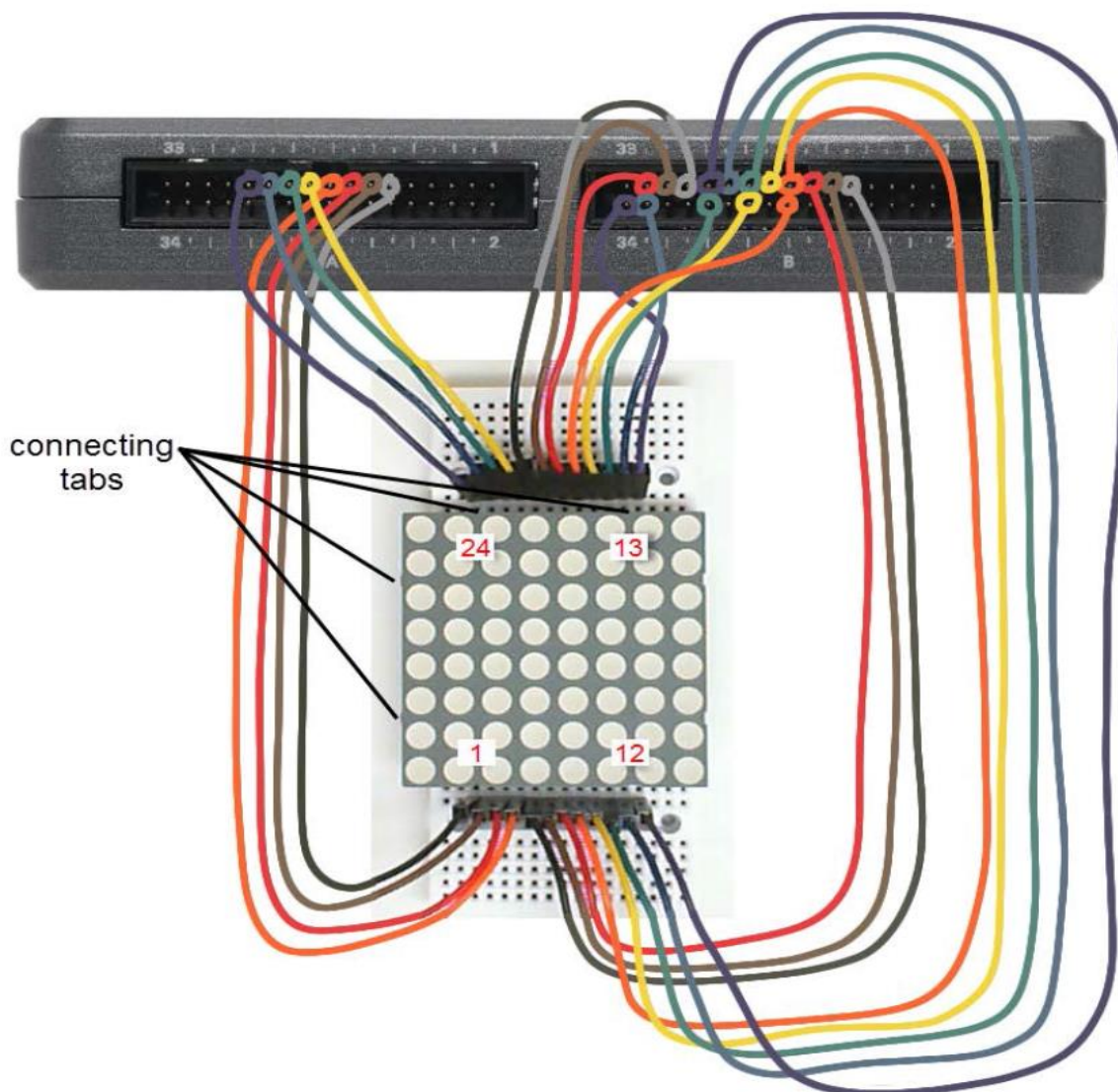
<http://youtu.be/vsBjZBLdeNc>

NiMyRio Project Essential Guide

LED Matrix

- 8 x 8 matrix, 64 red & green LEDs
- Wiring diagram
- Electrical Interface
- Raster display technique

Poznasz schematy połączeń elektrycznych, dowiesz się jak uzyskać dostęp do wszystkich LED, poznasz re-



Rysunek 29-3: Układ pokazowy Projekt: Matryca LED, propozycja połączeń matrycy LED - 24 I/O do złączy A i B NiMyRio.

wiersza. Anody wszystkich zielonych LED w analogicznym przewodzie pionowym kolumny. Anody czerwonych LED połączone są w podobny sposób, co za-

guly łączenia LED, problemy złączy elektrycznych (zwłaszcza znaczne niedopasowanie *napięcia/prądu* pomiędzy czerwonymi i zielonymi LED) oraz rastrową technikę wyświetlania

UWAGA: łączenie techniki opisanej w tym rozdziale nie wymaga żadnych dodatkowych elementów, a także pomaga zrozumieć, jak ustawić wyświetlanie *rastra* (zwanego również wyświetlaczem multipleksowanym). Jednak 24 DIO jest wykorzystanych, to ponad połowa z dostępnych 40 DIO, a także wiąże udostępnione zasoby, takie jak: **SPI**, **I²C-bus PWM** i enkoder, ponadto skanowanie kodu rastra wymaga znacznych nakładów mocy procesora.

Zastanów się, może skorzystasz ze sterownika wyświetlacza MAX7219 LED (patrz: 29.5. Więcej informacji..., sekcja końcowa tego rozdziału), aby całkowicie odciążyć NiMyRio szczegółowym sterowaniem matrycy LED; po prostu można w ten sposób przenieść potrzebne wzorce graficzne poprzez magistralę SPI do sterownika. Ta technika staje się konieczna i aplikacje powinny z niej korzystać, zwłaszcza, gdy trzeba obsłużyć więcej matryc LED, by budować większe wyświetlacze.

Programowanie LabView:

Uważnie przestuduj wideo:

Digital Output low-level subVI (04:52)

<http://youtu.be/WvvnInG3ffqY>

NiMyRio Project Essential Guide

Digital Output subVI

- Control one or more digital outputs with the low-level Digital Output sub VIs

Przykłady wykorzystania subVIs, do obsługi bezpośredniej tablic logicznych wyjść cyfrowych, jako sterowników matrycy LED,

29.3. Podstawowe modyfikacje

Uważnie przestuduj wideo:

LedMatrix Walk-Through (12:13)

<http://youtu.be/Bqq63sKwQKE>

NiMyRio Project Essential Guide

LED Matrix Demo

- Walk-Through the

Led Matrix Demo” LabView Project

Poznasz podstawy projektowania *układów demo LED Matrix*, spróbuj włączyć poniższe zmiany do *Main.vi*:

- 1) Zakończ pętlę wzorca: *AutoPatternGenerator* i zaktualizuj pętlę główną, dodaj kodową tablicę wzorów dla czerwonych LED.
- 2) Utwórz zestaw wzorów zakodowanych: 64-bitowych tablicy 1-D *AutoPatternGenerator*. Może się okazać, że wygodniej zmienić stałe do sterowania

na *FrontPanel*. Możesz także użyć *ArraySize* z *Programming | ArraySubpalette* zamiast stałej 4, aby Twój kod wstawił dowolną liczbę 64-bitową.

- 3) Pomyśl o sposobie mapowania jednego lub więcej parametrów, takich jak wbudowane złącze akcelerometru lub prezentacja analogowych wartości napięć wejściowych na wyświetlaczu. Na przykład, można zrobić osiem wykresów słupkowych i pokazać osiem analogowych napięć wejściowych, czy można mapować wyjścia Y z akcelerometru na siatce kartezjańskiej? Docelowe mapowanie jest 2-D Boolean tablicy zmiennych globalnych.
- 4) Dodaj obraz wskaźnika zmiany czasu w pętli czasowej: przeciągnij w dół *RightDataNode* w prawym górnym wnętrzu pętli czasu wystaw *NextLoopIterationTiming* terminal (pojawi się w *dt*), a następnie utwórz formant *FrontPanel*, aby interaktywnie ustawić czas pętli. Ustaw dolny limit na 1 ms, aby uniknąć wybierania 0 ms (to może zatrzymać aplikację).
- 5) Obserwuj efekty fluktuacji na wyświetlaczu: zmiany czasowe pętli, do pętli *while standard* (kliknij prawym przyciskiem myszy na ramce pętli i wybierz *ReplaceWithWhileLoop*, a następnie dodaj 1 ms opóźnienia. Należy zauważyć, że intensywność wyświetlania ma niewielkie losowe migotania (pulsacje), ponieważ procesor w czasie rzeczywistym ma teraz większą swobodę przetwarzania zadania w tle. Z drugiej strony pętla czasowa zapewnia teraz jej dokładny czas.

29.4. Pomysły integracji Projektu

Wicie, jak stosować matrycę LED, możecie zintegrować Projekt w bardziej złożone systemy, na przykład:

- Poziomica cyfrowa (56); *Digital Bubble Level* (56);
- Zegar NTP (42); *NTP Clock* (42);

29.5. Więcej informacji...

8x8 LED Matrix Datasheet by SparkFun~

Dane techniczne dla 8x8 matrycy LED:

<http://www.sparkfun.com/datasheets/Components/SanYoung-Medium-RG.pdf>

MAX7219 LED Display Driver with SPI Interface by Maxim Integrated~

Zawiera tranzystory mocy i karty graficzne rastrowe, aby całkowicie odciążyć sterowanie każdej LED. Wystarczy wysłać wzory grafik poprzez szeregową magistralę SPI:

<http://www.maximintegrated.com/MAX7219>

NOTATKI: