

KoPaR

KOŁO NAUKOWE ROBOTYKÓW

Eagle 3D

Простой!

(proste)

Karol Sydor - styczeń 2008

Żeby wykonać trójwymiarowy rendering projektu płytki drukowanej, musimy ściągnąć z Internetu dwa programy:

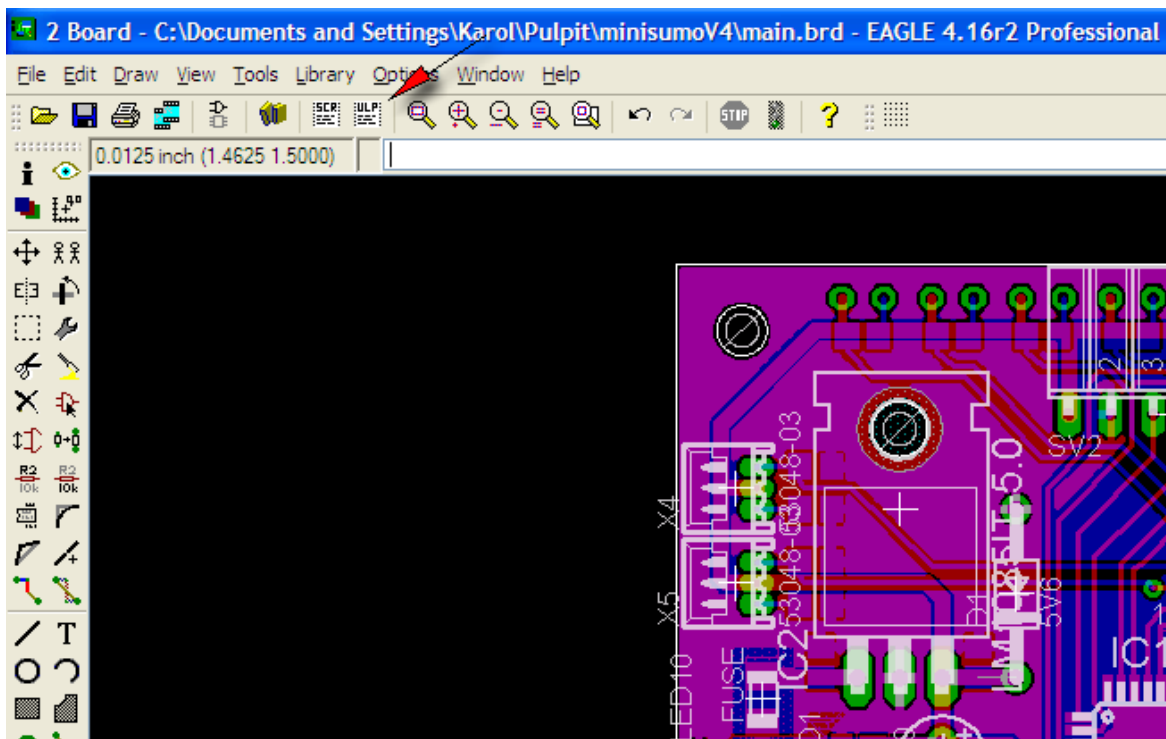
- Eagle3D – to darmowy zestaw skryptów, dostępny na stronie <http://www.matwei.de> (w chwili pisania artykułu) który należy zainstalować w podkatalogu ULP Eagla. Przykładowo:
C:\Program Files\EAGLE-4.16r2\ulp\Eagle3D\
- Pov-Ray – to darmowy program do renderowania, oparty o język skryptowy, dostępny na stronie <http://www.povray.org>

Następnie należy przekopiować zawartość folderu „povray” z katalogu Eagle3D, do podkatalogu „include” w katalogu Pov-Ray. Przykładowe ścieżki dostępu do tych folderów mogą wyglądać tak (zależne od wersji):

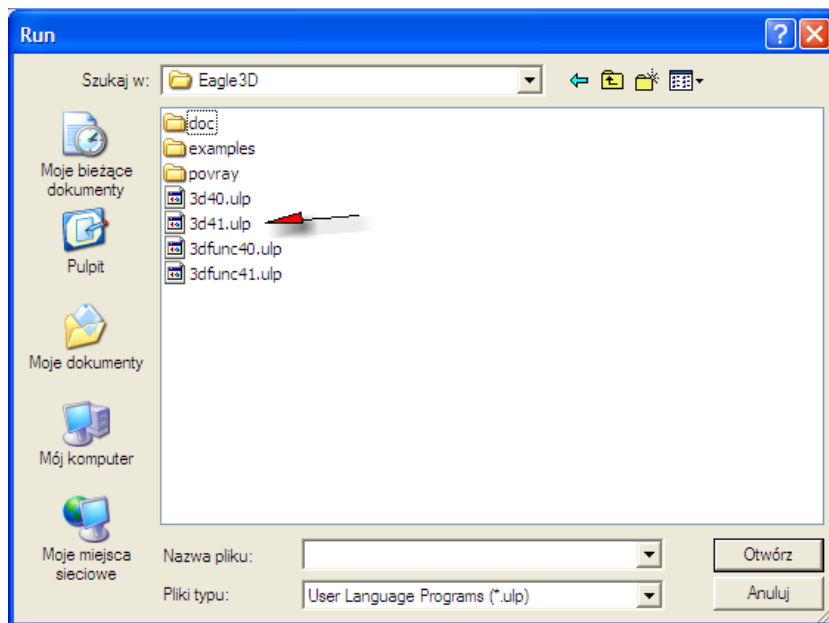
C:\Program Files\EAGLE-4.16r2\ulp\Eagle3D\povray
C:\Program Files\POV-Ray for Windows v3.6\include

Po zainstalowaniu programów możemy przystąpić do tworzenia wizualizacji.

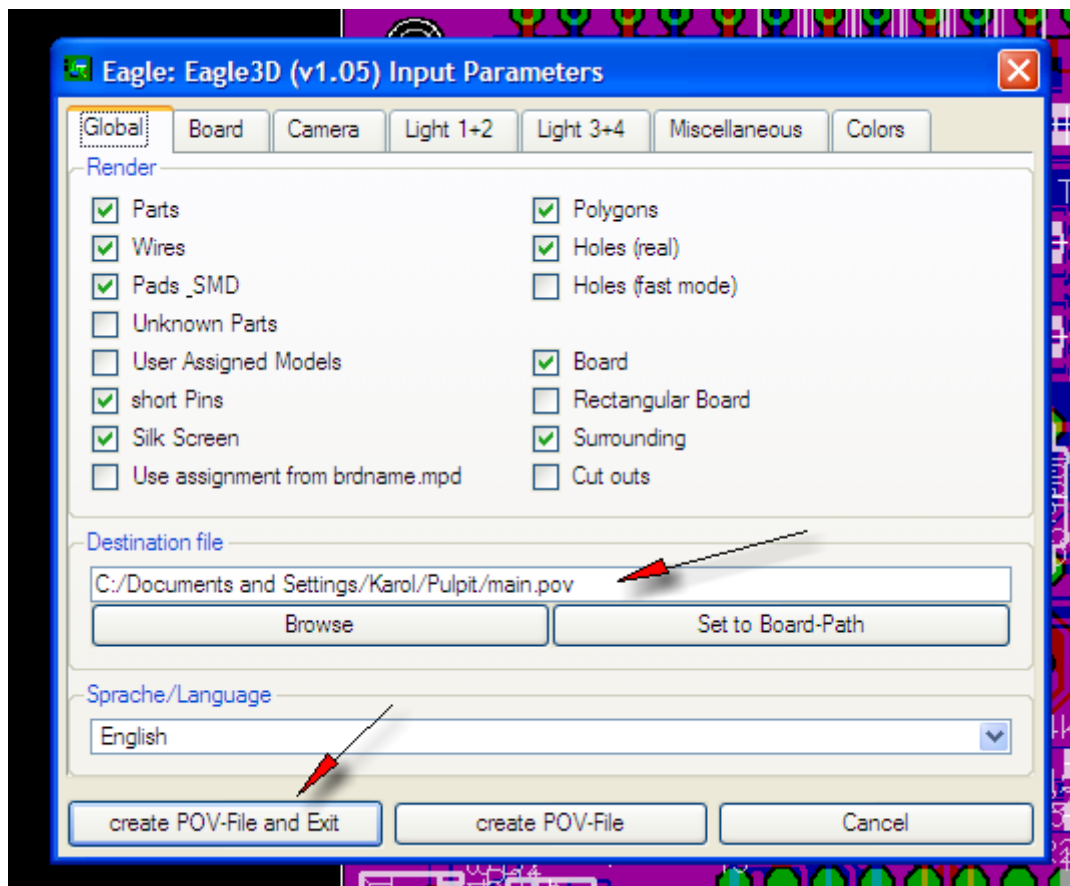
W tym celu otwieramy projekt płytki w eagle, a następnie klikamy na ikonkę ULP.



Wchodzimy do katalogu Eagle3D i otwieramy 3D41.ulp jeśli mamy Eagle w wersji 4.1x, bądź 3D40.ulp jeśli mamy wersję 4.09r2 (program nie działa z wcześniejszymi).

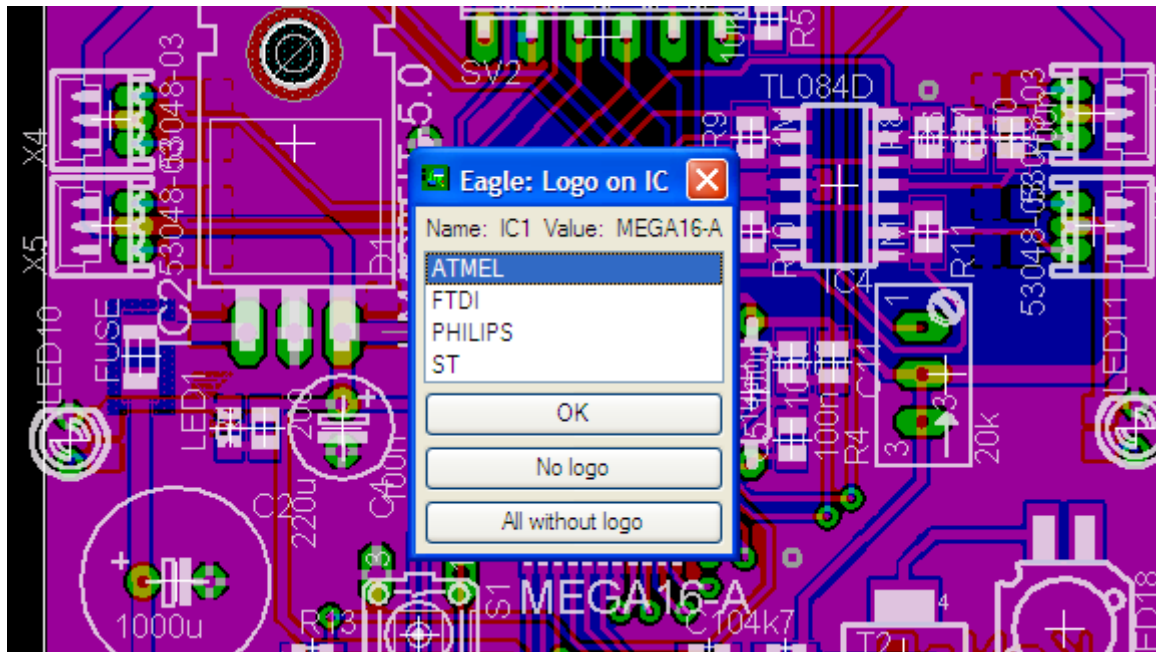


Ukaże nam się okno z kilkoma opcjami:

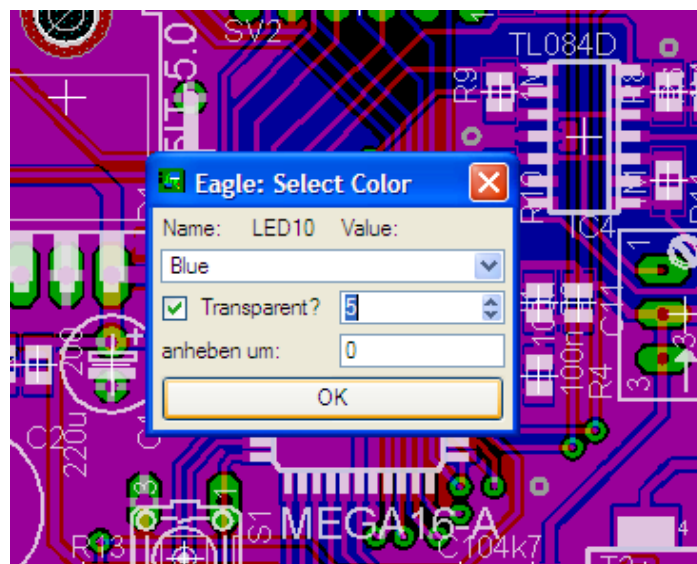


Należy ustawić odpowiednią ścieżkę do zapisu pliku wyjściowego (w formacie .pov) Możemy korzystając z zakładek zmienić orientację kamery, płytki, zmienić

parametry oświetlenia i wybrać kolory płytki. Ustawienia domyślne są optymalne. Klikamy „create POV-File and Exit”. Program rozpocznie generowanie skryptu. Może się okazać że program zapyta nas o pewne parametry płytki.

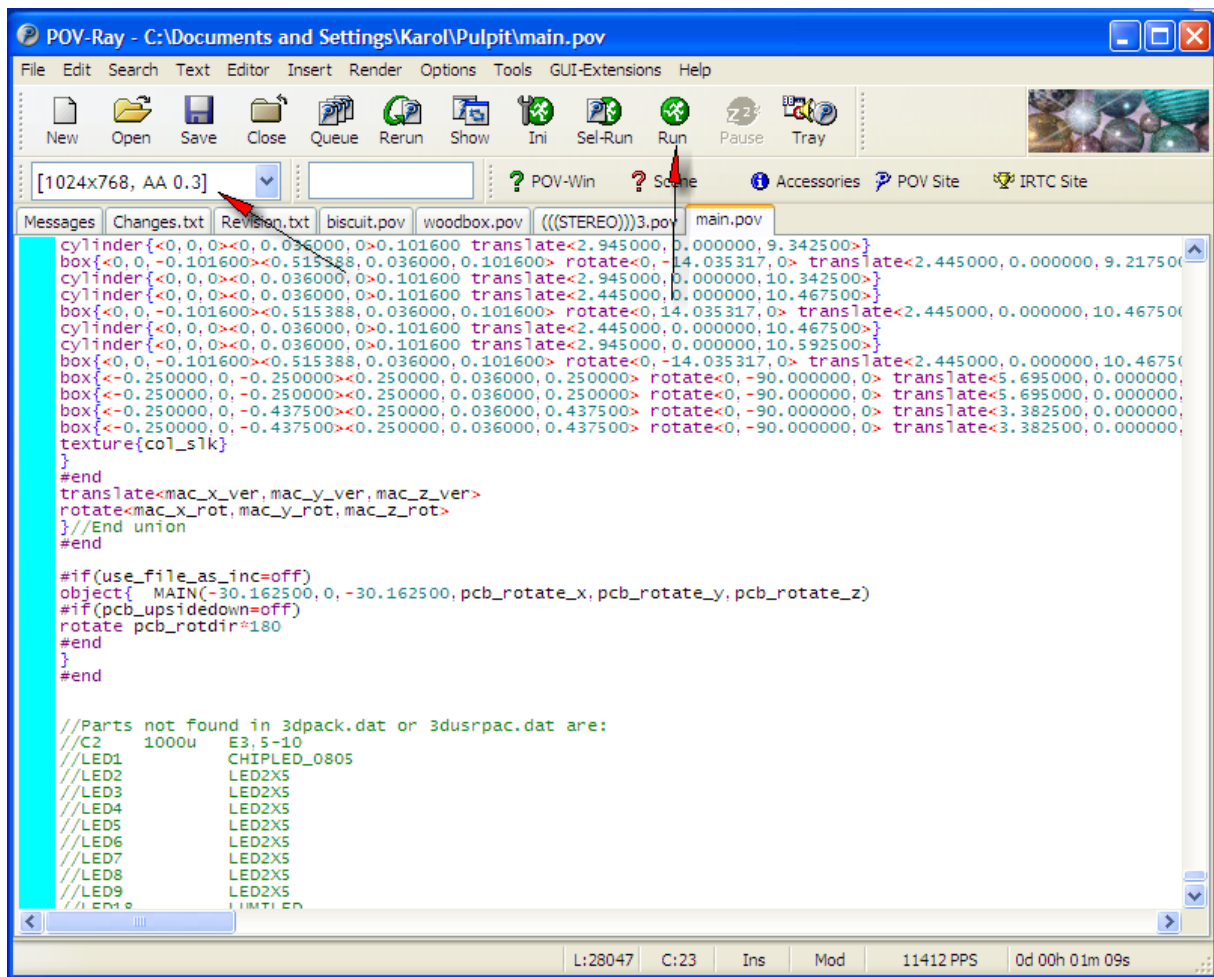


Tutaj pyta o to jakie logo ma się pojawić na elemencie IC1.



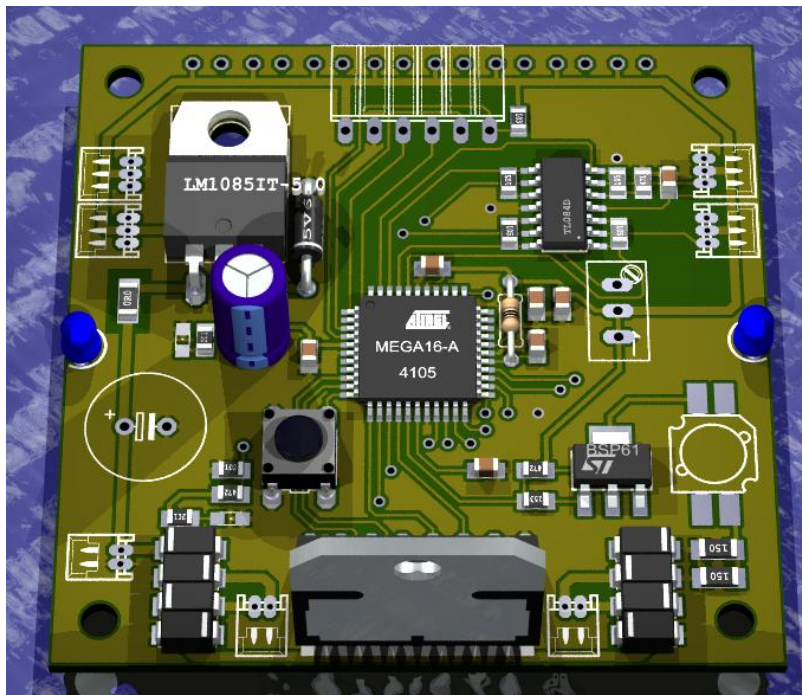
Ponieważ na płytce są diody LED, należy wybrać ich kolor, stopień przezroczystości (0- nieprzezroczysta, 10- przezroczysta, zabarwienie pozostaje) oraz wysokość nad płytką (1 jednostka to 1mm)

Gdy skrypt jest gotowy, uruchamiamy POV-Raya i otwieramy plik .pov wygenerowany przez Eagle3D.

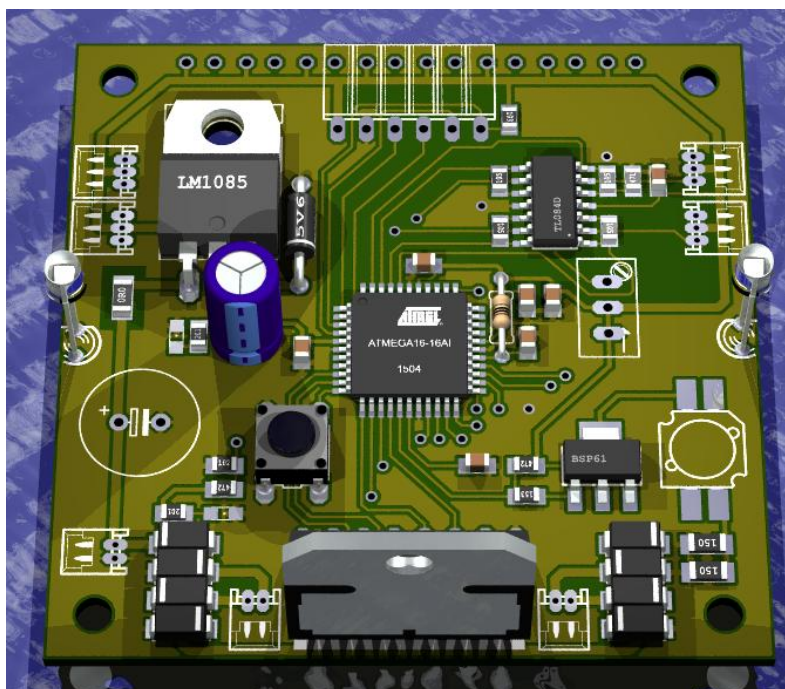


Ustawiamy porządaną rozdzielczość (im większa tym rendering trwa dłużej) i klikamy przycisk „Run”

Naszym oczom powinna ukazać się wyrenderowana płytko drukowana. Nie wszystkie elementy posiadają odpowiednie biblioteki 3D umożliwiające wyświetlenie, zamiast tych elementów będzie ich obrys na warstwie opisowej. Warto zauważyć że wartości rezystorów są odwzorowane poprawnie.

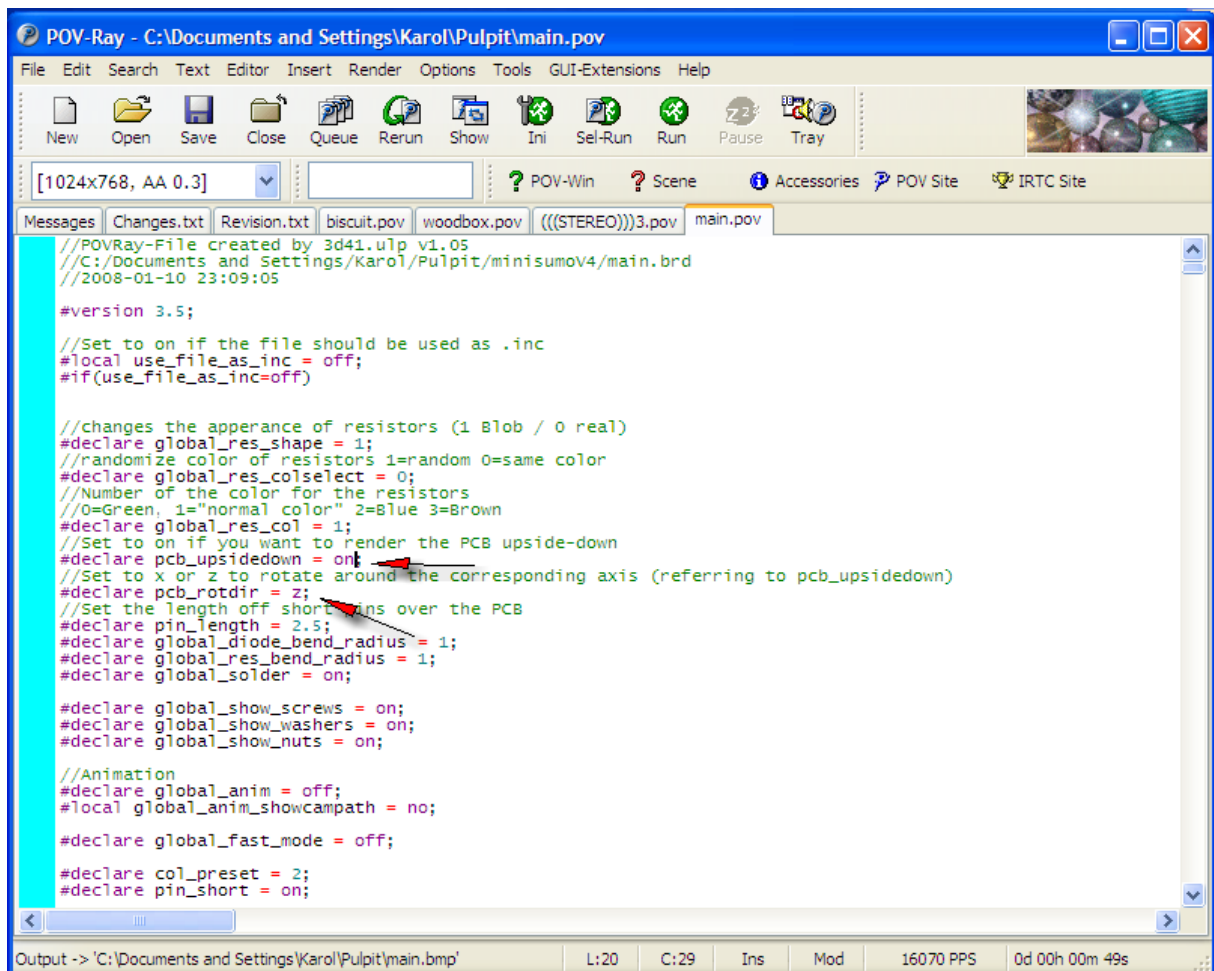


Jednak niektóre opisy są nieco „za duże” i nie mieszczą się na scalakach, podobnie jak logo na tranzystorze. Aby temu zaradzić należy zmienić nazwę w Eagle i wygenerować skrypt ponownie, jednak wystarczy zajrzeć w skrypt aby zauważyć że jego struktura jest dość prosta i wszelkich poprawek można dokonać bez edycji pliku .brd. Bardzo pomocne jest przy tym podświetlanie składni w edytorze pov-raya.



Poprawione opisy i zmiana parametrów LEDów poprawia znacznie wygląd wizualizacji. Program po wykonaniu renderingu automatycznie zapisuje plik z taką samą nazwą i rozszerzeniem .bmp, jest to standardowa bitmapa.

Aby zrobić wizualizację drugiej strony płytki, możemy uruchomić Eagle3D ponownie i dokonać rotacji płytki wzdłuż osi Z o 180, lub zmienić dwie linie w skrypcie i uzyskać ten sam efekt.



```
POV-Ray - C:\Documents and Settings\Karo\Pulpit\main.pov
File Edit Search Text Editor Insert Render Options Tools GUI-Extensions Help
New Open Save Close Queue Rerun Show Ini Sel-Run Run Pause Tray
[1024x768, AA 0.3]
POV-Ray File created by 3d41.ulp v1.05
C:\Documents and Settings\Karo\Pulpit\minisumov4\main.brd
2008-01-10 23:09:05

#version 3.5;

//set to on if the file should be used as .inc
#local use_file_as_inc = off;
#if(use_file_as_inc=off)

//changes the appearance of resistors (1 Blob / 0 real)
#declare global_res_shape = 1;
//randomize color of resistors 1=random 0=same color
#declare global_res_colselect = 0;
//Number of the color for the resistors
//0=Green, 1="normal color" 2=Blue 3=Brown
#declare global_res_col = 1;
//set to on if you want to render the PCB upside-down
#declare pcb_upsidedown = on;
//set to x or z to rotate around the corresponding axis (referring to pcb_upsidedown)
#declare pcb_rotmdir = z;
//set the length off short pins over the PCB
#declare pin_length = 2.5;
#declare global_diode_bend_radius = 1;
#declare global_res_bend_radius = 1;
#declare global_solder = on;

#declare global_show_screws = on;
#declare global_show_washers = on;
#declare global_show_nuts = on;

//Animation
#declare global_anim = off;
#local global_anim_showcampath = no;

#declare global_fast_mode = off;

#declare col_preset = 2;
#declare pin_short = on;
```

Należy zamienić parametr „pcb_upsidedown” na „on” oraz „pcb_rotmdir” na „z”.

Oto efekt:

