


Kurs obsługi EAGLE, część 7

Wszystkie elementy, z których składa się projekt są pobierane z bibliotek. Dzięki możliwości tworzenia własnych elementów bibliotecznych program EAGLE zyskuje na elastyczności. Można zaprojektować bowiem elementy niestandardowe oraz elementy wprowadzone niedawno na rynek.

W tym odcinku opisujemy sposób tworzenia nowych elementów bibliotecznych. Opis został przeprowadzony na podstawie konkretnego przykładu diody, której model został zaprojektowany w dwóch różnych obudowach.

Na początku przyjęliśmy, że diodę zaprojektujemy w dwóch różnych obudowach: przewlekanej oraz SMD. Musimy więc zaprojektować jeszcze jedną obudowę, tym razem przystosowaną do montażu powierzchniowego – SMD. Odpowiednikiem diody 1N4007 jest SM4007 w obudowie DO-213AB. Proces tworzenia obudowy SMD jest praktycznie identyczny jak przewlekanej. Jedyną różnicą jest to, że w miejsce padów użyjemy pól SMD.

Rozpoczynamy identycznie jak ze zwykłą obudową: klikamy na *Package*, po czym w okienku *New* wpisujemy DO213AB. Potwierdzamy przez *OK*, w następnym okienku potwierdzamy chęć utworzenia nowego elementu. Zostaje otwarte okno edytora, w którym to możemy rozpocząć już rysowanie naszej obudowy. W przypadku elementów SMD, których to obudowy mają zazwyczaj mniejsze wymiary, warto ustawić raster na 5 lub nawet na 2,5 mil. Po wydaniu komendy SMD (lub kliknięciu ikonki ) na pasku *Parameters Toolbar* mamy dostęp do elementów decydujących o kształcie pól SMD (**rys. 39**). Parametry te możemy zmienić przed położeniem pola, lub później przy pomocy komendy *Change*. Mamy wpływ na następujące elementy:

Layer – płaszczyzna, na której umieszczamy SMD. W przypadku elementów bibliotecznych, wybieramy zawsze stronę górną płytki: *1 Top* nawet, jeżeli element ma zostać umieszczony na stronie dolnej. W czasie projektowania płytki, jeżeli chcemy przełożyć element na drugą stronę, używamy polecenia *MIRROR*.



Rys. 39.

Po jego wydaniu wszystkie elementy skojarzone z obudową, znajdujące się na płaszczyznach rozpoczynających się na *t...* (21, 23, 25, 27...) zostają przełożone na odpowiadające im płaszczyzny rozpoczynające się na *b...* (22, 24, 26, 28...).



Size – wielkość pola. Możemy wybrać jedną z dostępnych, lub kliknąć na pole i wpisać z klawiatury potrzebną nam wartość. W przypadku naszej diody wpisujemy wartość 100x60 mil (2,54x1,524 mm).

Roundness – określa stopień zaokrąglenia rogów pól SMD. Najlepiej ustawić go na 0 % dzięki temu zaokrąglenie pól możemy określić globalnie w ustawieniach DRC.

Rotation – kąt, pod którym pole jest obrócone. Możemy wpisać praktycznie dowolną wartość z dokładnością do jednego miejsca po przecinku.

Położymy więc na płaszczyźnie *1-Top* dwa pola o wymiarach 100x60 mils i kącie 90 stopni. Jedno w miejscu o współrzędnych (-100,0), drugie (100,0). Kolejne czynności są identyczne jak przy tworzeniu obudowy elementu przewlekane, nie będą więc powtórnie opisane. Warto zaznaczyć, że w przypadku miniaturowych elementów SMD, może się okazać, iż na płytce nie ma już miejsca na narysowanie kształtu elementu. Należy wtedy umieścić go w całości na płaszczyźnie 51 *tDocu*. Widok gotowej obudowy przedstawiono na **rys. 40**.

Gdy uporaliśmy się już z obudowami należy stworzyć symbol, przedstawiający element na schemacie. Klikamy na

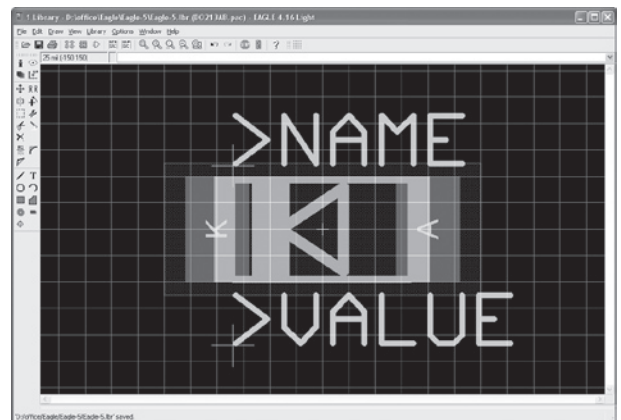
ikonkę *Symbol* , po czym w okienku *New* wpisujemy „Dioda”. Po potwierdzeniu otwiera się okno edytora symboli, jest ono praktycznie identyczne z edytorem schematów. Pierwszą czynnością jest ustawienie rastra. Jak wspomniano w czasie opisu edytora schematów, aby zapewnić pewne połączenie pomiędzy elementami, piny symboli w bibliotekach są rozmieszczone w rastrze 100 mils, i taki właśnie ustawimy. Następnie umieścimy końcówki naszej diody. Do tego celu służy polecenie *Pin* . Po jego aktywowaniu, na pasku *Parameters Toolbar* pojawiają się elementy dzięki którym możemy zmienić jego parametry (**rys. 41**). Jak w większości przypadków, w czasie pracy z programem Eagle parametry możemy zmienić przed położeniem pinu lub później przy pomocy polecenia *Change*.

Ikony realizują następujące zadania:

Orientation – pierwsze cztery, pozwalają na zmianę kierunku pinu. Alternatywnie można ją zmienić klikając prawym klawiszem myszy, w momencie, gdy pin jest podwieszony do kursora.

Function – określa sposób przedstawienia pinu. Do wyboru mamy: brak określonej funkcji, symbol negacji, wejście zegarowe, zanegowane wejście zegarowe.

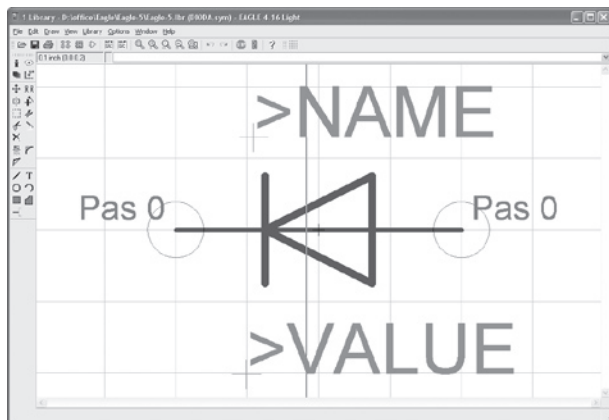
Length – określa długość pinu. Do wyboru mamy (0, 100, 200, 300) mils. Pin jest symbolizowany jako linia na płaszczyźnie 94 – *Symbols*. W czasie kreślenia sche-



Rys. 40.



Rys. 41.



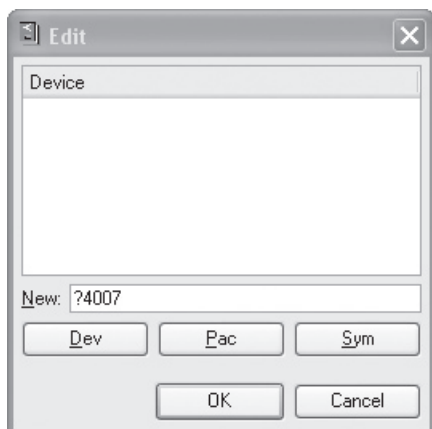
Rys. 42.

matu mamy możliwość sprawdzenia czy sygnał jest prawidłowo podłączony do pinu. Służy do tego komenda *Show*. Po jej uruchomieniu i kliknięciu na sygnał, zostaje on podświetlony wraz z podłączonymi do niego pinami. Jeżeli pin ma długość równą zero, to nie jest on podświetlany, z tegoż powodu zaleca się stosowanie długości różnej od zera.

Visible – Określa czy opis pinu, oraz nazwa końcówki obudowy mają być widoczne, czy też nie. Położenie, oraz wielkość (60 mils) tych opisów jest sztywno określone i nie można ich później zmienić. Do wyboru mamy cztery możliwości: oba opisy wyłączone, opis końcówki włączony, opis pinu włączony, oba włączone.

Direction – określa logiczny kierunek przepływu sygnałów. Parametr ten wpływa na przeprowadzany na schemacie test ERC. Przykładowo test ten wskaże błąd, jeżeli na jednym sygnale leżą jedynie same wejścia lub, gdy mamy połączonych ze sobą kilka wyjść.

Pozzczególne opcje mają następujące znaczenie:



Rys. 43.

NC – niepodłączone

In – wejście

Out – wyjście

I/O – wejście/wyjście

OC – wyjście typu otwarty kolektor

Hiz – wyjście o wysokiej impedancji

Pas – pasywne (końcówki rezystorów, kondensatorów itp.)

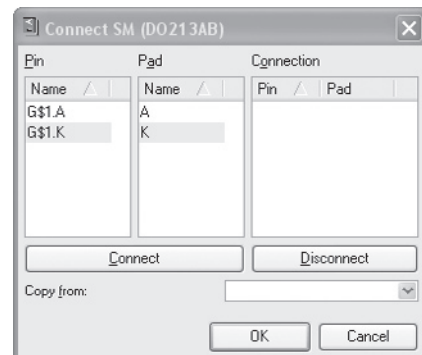
Pwr – wejścia zasilania

Sup – wyjście zasilania,

stosowane dla symboli masy, oraz zasilania.




Swaplevel – Jeżeli w czasie projektowania symbolu, kilku pinom nadamy ten sam *Swaplevel*, to będzie je można na schemacie zamieniać za pomocą komendy *Pinswap*. Gdy podamy wartość zero, to pinu nie uda się z żadnym innym zamienić. Przykładowo w wielowejsciowej bramce NAND, wszystkie wejścia mają identyczne funkcje, można je więc zamienić miejscami. W przypadku diody poszczególnych końcówek nie wolno zamieniać, dlatego też otrzymują one *Swaplevel* równy zero.

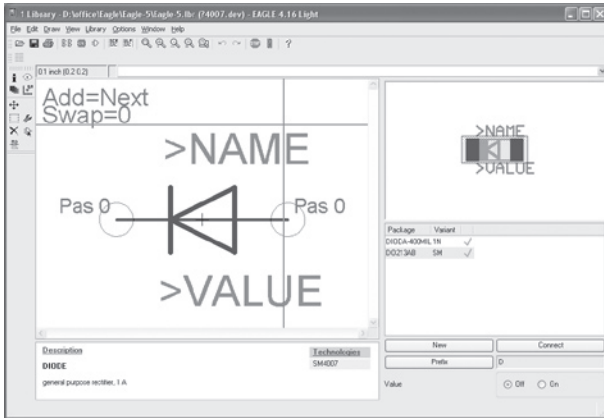
Położmy teraz dwa piny o następujących parametrach: *Function* – *none*, długość 100 mils, *Visible* – *off*, *Direction* – *Pas*, *Swaplevel* – 0. Pierwszy w punkcie (-200,0) mils i rotacji R0, drugi w punkcie (200,0) mils i rotacji R180. Następnie, przy pomocy funkcji *NAME* nadajemy im odpowiednie nazwy. I tak: lewemu pinowi nadajemy nazwę „K”, prawemu „A”, po czym poleceniem *WIRE* oraz innymi narzędziami graficznymi, kreślimy na płaszczyźnie 94 – *Symbols* właściwy kształt symbolu. Dla linii odchodzących od pinów, grubość ustawiamy na 6 mils, ponieważ taką samą wartość ma grubość linii symbolizująca pin. Resztę elementów kreślimy linią o grubości 10 mils. W celu wierniejszego odwzorowania symbolu możemy ustawić dokładniejszy raster. Kolejną czynnością jest wpisanie tekstów, symbolizujących nazwę oraz wartość danego elementu. Powinny one zostać umieszczone w pobliżu elementu. W tym celu na płaszczyźnie 95 – *Names* wpisujemy *tekst* >NAME, a na płaszczyźnie 96 – *Values* *tekst* >VALUE. Oba przy pomocy czcionki o wielkości 70 mils.



Rys. 44.

Na tym zakończyliśmy tworzenie symbolu. Gotowy, przykładowy symbol przedstawiono na **rys. 42**.

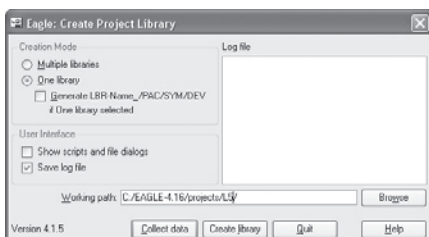
Pozostało nam jeszcze tylko podłączenie symbolu do obudowy, czyli stworzenie elementu wynikowego – *Device*. Dokonujemy tego klikając na ikonę  lub wybierając z menu głównego *Library/Device...* W nowo otwartym okienku (**rys. 43**) w polu *New* wpisujemy nazwę naszego elementu. Powinna ona brzmieć „?4007”. Znak zapytania symbolizuje przedrostek, który zmienia się automatycznie w zależności od obudowy. Dla diody SMD nazwa przyjmie postać SM4007, a dla diody w obudowie przewlekanej postać 1N4007. Po przyciśnięciu *OK* zostaniemy zapytani czy chcemy utworzyć nowy symbol: „*Create new device?4007*”. Potwierdzamy klikając na *Yes*. Po tych zabiegach zostaje otwarte okno edytora *Device*. Pierwszą czynnością jest wstawienie wcześniej przez nas utworzonego symbolu diody. Służy temu polecenie *Add* lub ikonka  leżąca na pasku po lewej stronie ekranu. Z nowo otwartego okienka wybieramy potrzebny nam symbol (DIODA), po czym umieszczamy go tak, aby jego środek znajdował się w punkcie o współrzędnych (0,0). Punkt ten jest późniejszym punktem zaczepienia dla elementu na schemacie. Parametry *Addlevel* oraz *SwapLevel* są domyślnie ustawione na odpowiednio *next* oraz 0. W przypadku elementów składających się tylko z jednego symbolu, zaleca się pozostawienie tychże domyślnych wartości. Kolejnym elementem w oknie edytora jest komenda *Name* . Ma ona znaczenie jedynie w przypadku elementów składających się z większej liczby symboli. W naszym przypadku, jednego tylko symbolu, nie ma ona znaczenia, gdyż nazwa którą nadamy symbolowi i tak nie pojawi



Rys. 45.

się później na schemacie. W takim przypadku producent zaleca pozostawienie nazwy wygenerowanej automatycznie (G\$1).

Po prawej stronie okna, u dołu, znajduje się przycisk *New* służy on do dołączenia do elementu jednej z wcześniej zdefiniowanych obudów. Po jego przyciśnięciu zostaje otwarte nowe okienko, w którym możemy wybrać interesujący nas wariant obudowy. Wybierzmy DO213AB, po czym w okienku *Variant Name* wpisujemy SM. Jeżeli tworzymy element tylko z jednym wariantem obudowy, okienko to możemy zostawić puste. Potwierdzamy klikając OK. W ten sam sposób dołączymy również drugą obudowę (DIODA-400MIL), otrzyma ona jednak nazwę 1N. Pod *New* znajduje się kolejny przycisk *Prefix*. Pozwala on na określenie przedrostka nazwy danego elementu. W czasie dokładania elementów do schematu, są one automatycznie numerowane z wykorzystaniem prefiksu, który możemy sztywno określić. W naszym przypadku, po przyciśnięciu pola *Prefix* w nowo otwartym okienku wpisujemy D, po czym potwierdzamy przez OK. Kolejnym parametrem, który możemy zmienić jest *Value*. Jeżeli zaznaczymy *On*, wartość naszego elementu będziemy mogli dowolnie zmieniać na schemacie. Jest to przydatne dla rezystorów, kondensatorów oraz innych tego typu elementów. Dla półprzewodników nale-



Rys. 46.

ży zastosować wartość *off*. Powoduje to przypisanie elementowi na schemacie nazwy zgodnej z *Device-name* uzupełnionej o rozszerzenia z wariantu obudowy oraz technologii.

Kolejnym przyciskiem jest *Connect* służy on do połączenia odpowiednich pinów symbolu z odpowiedzającymi im padami obudowy. Po jego przyciśnięciu zostaje utworzone okno, w którym dokonujemy właściwych połączeń (rys. 44). W lewym oknie (*Pin*) zaznaczamy G\$1.A, po czym w środkowym (*Pad*) zaznaczamy A. Po przyciśnięciu pola *Connect* zaznaczony pin oraz pad zostaje przeniesiony jako para do okna prawego (*Connection*). Klikając na pola *Name*, *Pin*, *Pad* możemy zmienić kolejność, w której są wyświetlane poszczególne elementy. Następnie wprowadzamy kolejne (w naszym przypadku ostatnie) połączenie pomiędzy G\$1.K, a K. Jeżeli wkradł się nam błąd możemy niewłaściwe połączenie rozłączyć zaznaczając je, po czym klikając na przycisk *Disconnect*. Jeżeli oba połączenia są gotowe i nie mamy żadnego błędu, to możemy zamknąć okno klikając na OK. Opisaną procedurę należy powtórzyć dla drugiego wariantu obudowy. W oknie *Connect* jest wtedy aktywna funkcja *Copy from* pozwala ona na skopionanie połączeń z innej wersji obudowy.

Ostatnią już czynnością jest podanie opisu naszego elementu. Na schemacie, w czasie szukania elementów, program przeszukuje również tenże opis. Opis ten wprowadzamy w oknie powstałym po wydaniu komendy *Description* lub klikając to pole w dolnej, lewej stronie okna edytora. Tekst możemy formatować w sposób opisany wcześniej (*Rich-Text*). Okno z gotowym elementem przedstawiono na rys. 45.

Aby naszą bibliotekę można było użyć przy kreśleniu schematu oraz (lub) płytki, należy ją wcześniej dołączyć do projektu. W tym celu w jednym z powyższych edytorów użyjemy komendy *Use*. Możemy również w menu głównym wskazać *Library/Use...* Następnie należy wybrać potrzebny nam plik.

Jeżeli biblioteka została dołączona już wcześniej, a my dokonaliśmy w niej zmian lub dodaliśmy do niej nowe elementy. To aby zmiany te były widoczne w projekcie musimy bibliotekę tą zaktualizować. Służy temu polecenie *Update*. Polecenie to możemy również wskazać w menu głównym *Library/Update...*

W katalogu ULP oraz na stronie internetowej producenta: www.Cadsoft.de można znaleźć sporą liczbę programików ULP, które wspomagają tworzenie bibliotek. Między innymi można tam znaleźć parę wersji programiku *exp-project-lbr*. Wersja dołączona standardowo do programu nosi nazwę *exp-project-lbr.ulp*. Programikiem tym możemy eksportować do bibliotek elementy ze schematu lub projektu płytki. Jeżeli zostanie on uruchomiony z edytora płytki, powstanie biblioteka zawierająca jedynie obudowy elementów. Jeżeli uruchomimy go w edytorze schematów, powstanie kompletna biblioteka zawierająca obudowy, symbole oraz połączenia pomiędzy nimi – *Devices*. Okno główne programiku przedstawiono na rys. 46. Jest ono wyposażone w parę przełączników dzięki którym możemy zmienić główne parametry. Przede wszystkim możemy określić czy wszystkie elementy mają zostać umieszczone w pojedynczej bibliotece, czy też każdy w osobnej. Następnie możemy określić czy nazwy elementów w nowej bibliotece mają być rozszerzone o nazwy biblioteki z których one pierwotnie pochodzą, czy też nie. Pracę programiku rozpoczynamy klikając na przycisk *Collect data* po czym następuje zebranie informacji o wszystkich elementach znajdujących się w projekcie. Czas tego procesu jest uzależniony od wielkości projektu oraz od szybkości naszego komputera. Gdy wszystkie dane zostaną zebrane klikamy na przycisk *Create library*. Zostaje otwarte okno edytora bibliotek, po czym następuje import wszystkich elementów. Po zakończeniu biblioteka jest zapisywana automatycznie w katalogu, w którym znajduje się pierwotny projekt. Przyjmuje również jego nazwę z rozszerzeniem **.lbr*.

Na tym kończymy kolejny odcinek naszego kursu. Za miesiąc opiszemy tworzenie dokumentacji niezbędnej przy produkcji. Między innymi opiszemy tworzenie plików Gerber za pomocą procesora CAM.
inż. Henryk Wiczorek
henrykwiczorek@gmx.net