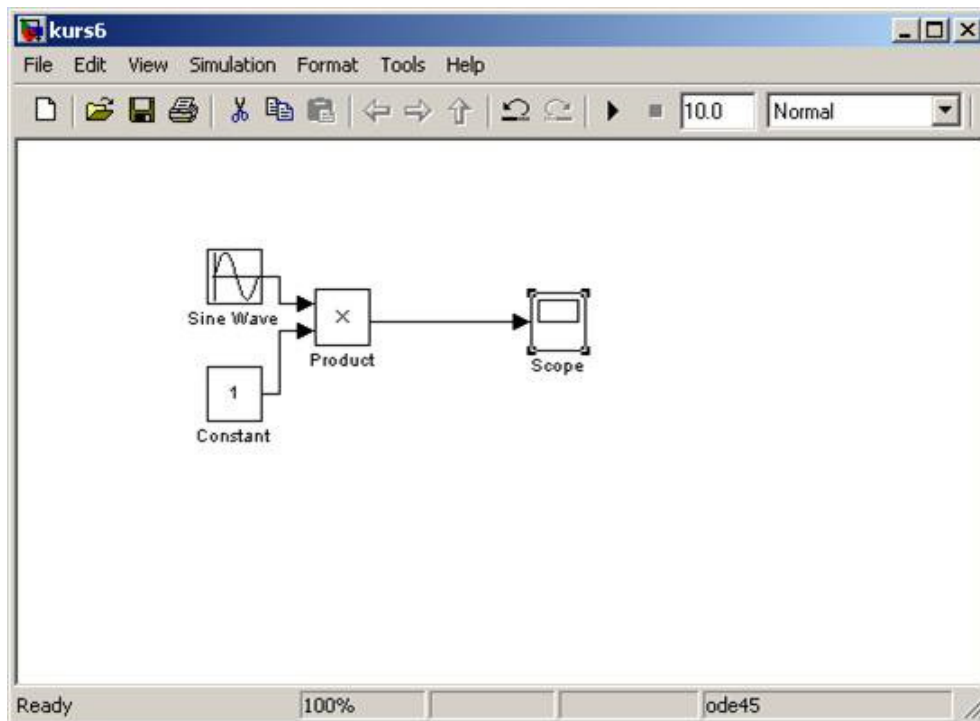


SIMULINK 6

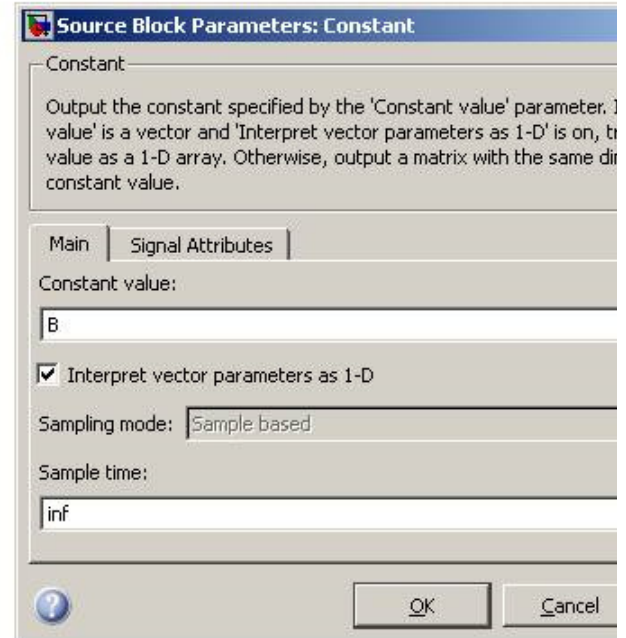
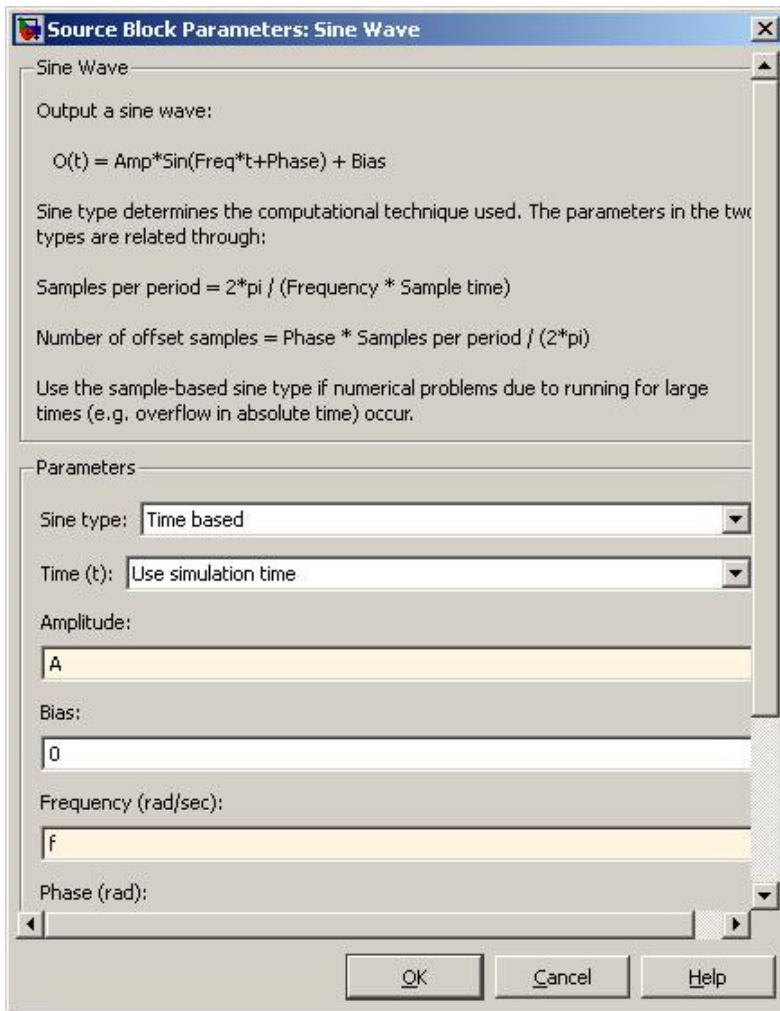
Więcej o podsystemach

W tym paragrafie powiemy kilka słów więcej o podsystemach, które można tworzyć w Simulinku. Zaczniemy od prostego modelu złożonego ze źródła sinusoidalnego, stałej, bloku mnożącego i bloku rysującego połączonych w następujący sposób:



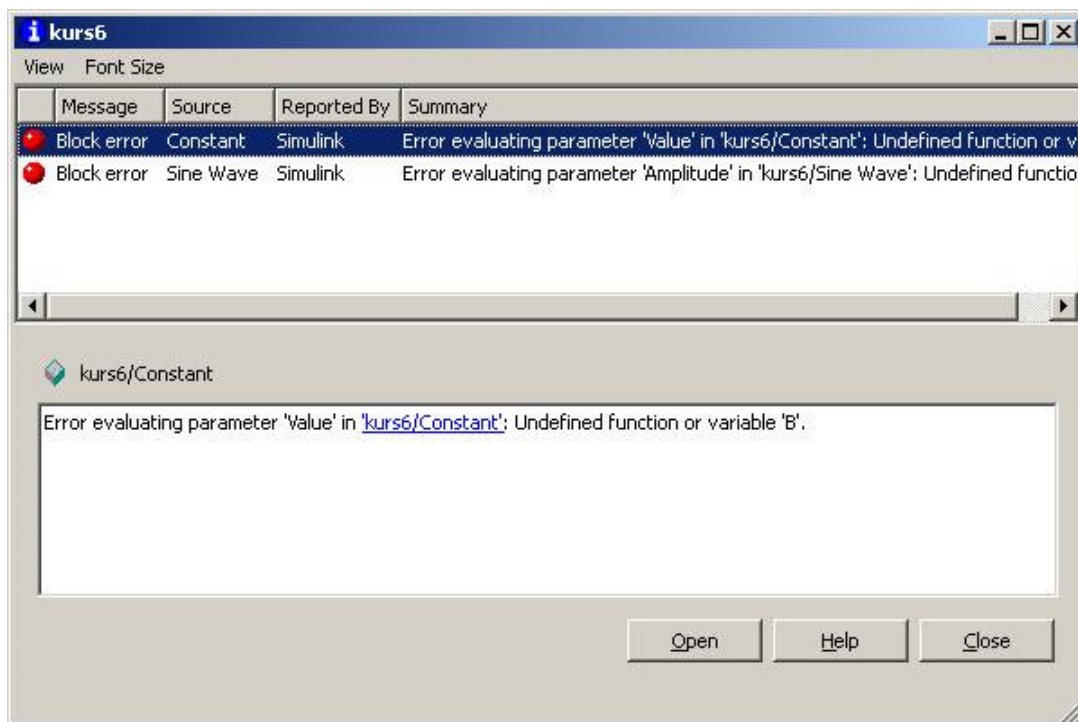
Rysunek 1: Model wyjściowy

Zanim utworzymy nasz podsystem zmienimy właściwości bloku *Sine Wave* i *Constant* w następujący sposób:

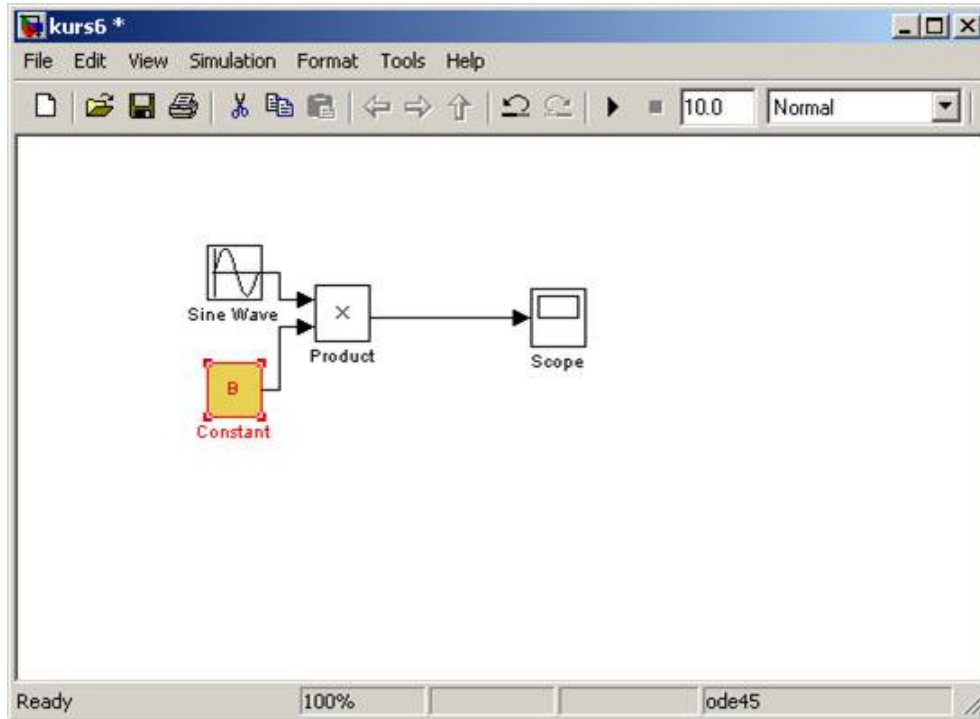


Rysunek 2: Ustawianie parametrów bloków

I nie robiąc nic więcej uruchommy symulację. Zakończy się to następującym błędem:



Simulink ostrzega nas, że zmienna B jest nie zdefiniowana w żadnym miejscu, podobnie użyte przez nas zmienne A i f . Dodatkowo blok, którego dotyczy wybrany błąd zmienia kolor aby można było go łatwiej znaleźć:



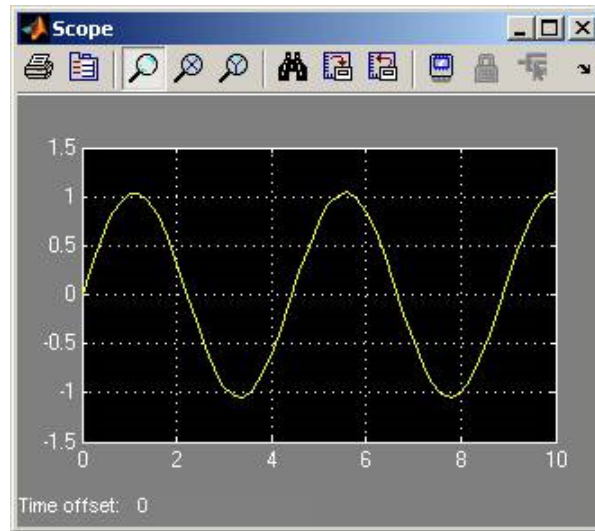
Rysunek 4: Dodatkowo Simulink podświetla blok odpowiedzialny za błąd

Pojawia się pytanie gdzie mamy zdefiniować te zmienne? Odpowiedzią jest przestrzeń robocza Matlab, ponieważ Simulink w pierwszej kolejności przeszukuje aktualny workspace aby znaleźć wywoływane przez nas zmienne. Zdefiniujmy zatem A , B i f w *command window*:

```
Command Window
>> A=1.1
A =
    1.1000
>> B=.95
B =
    0.9500
>> f=sqrt(2)
f =
    1.4142
fx >> |
```

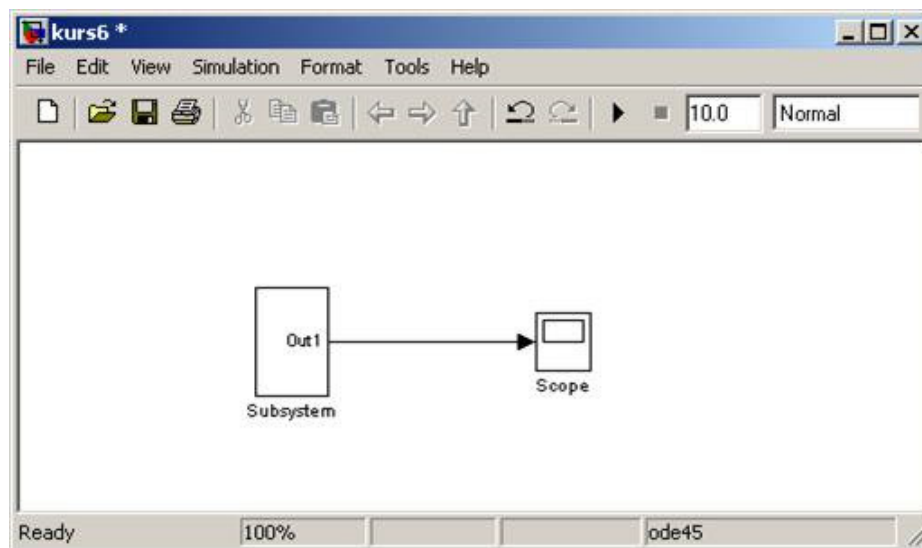
Rysunek 5: Definiowanie zmiennych w workspace Matlab

... i uruchommy nasz model. W wyniku dostaniemy sinusoidę o wybranych parametrach:



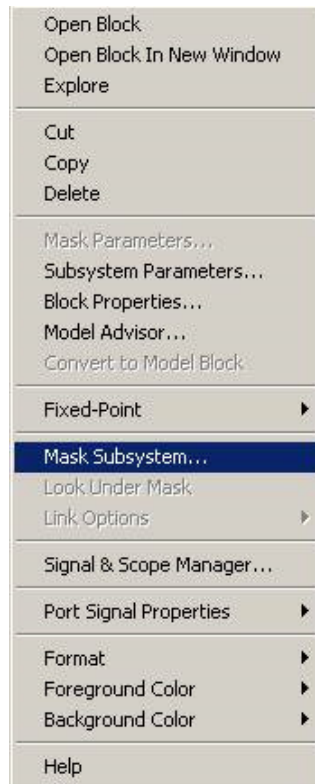
Rysunek 6: Wynik symulacji dla powyżej ustawionych parametrów

Utwórzmy teraz system, który nie będzie miał żadnego wejścia, ale będzie posiadał jedno wyjście i będzie on wykonywał operację mnożenia sinusoidy przez stałą, da to nam bardzo prosty w wyglądzie model:



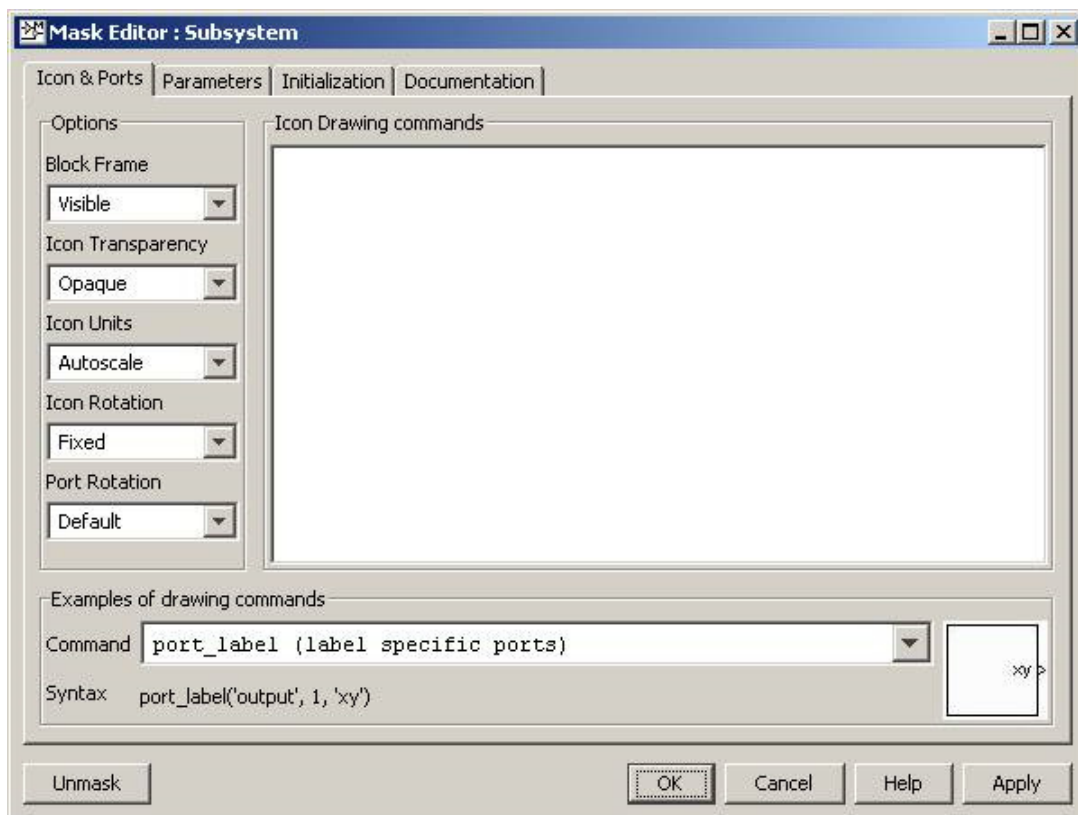
Rysunek 7: Utworzenie podsystemu

Teraz za pomocą menu kontekstowego wybierzmy opcję Mask Subsystem:



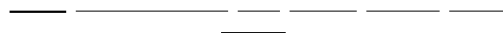
Rysunek 8: MASKOWANIE PODSYSTEMU

Wywoła to następujące okno, w którym możemy zmieniać właściwości naszego systemu, takie jak ikona, liczba portów, parametry, czy dodać dokumentację:



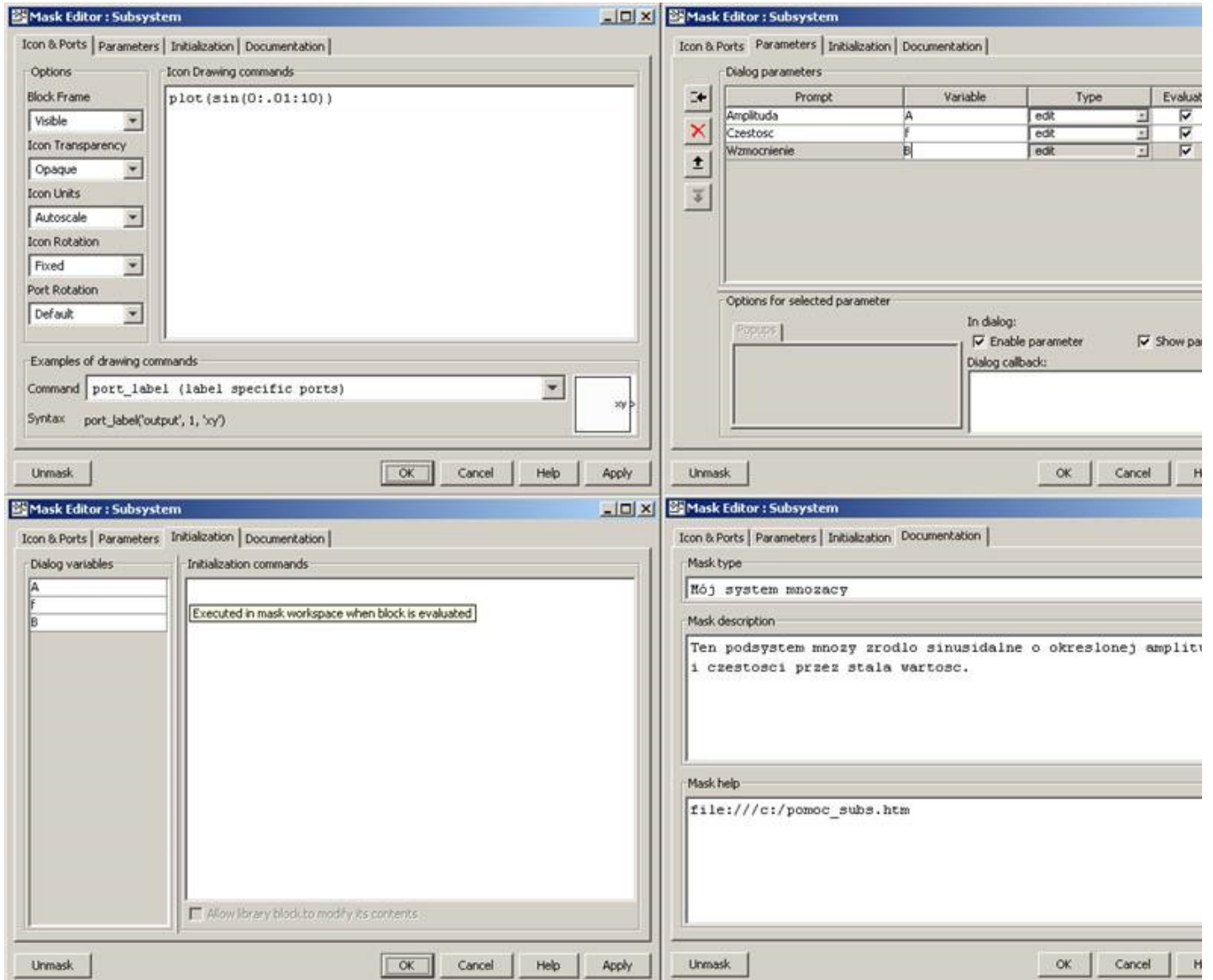
Rysunek 9: Maskowanie podsytsemu - okno preferencji

Składa się ono z następujących zakładek:



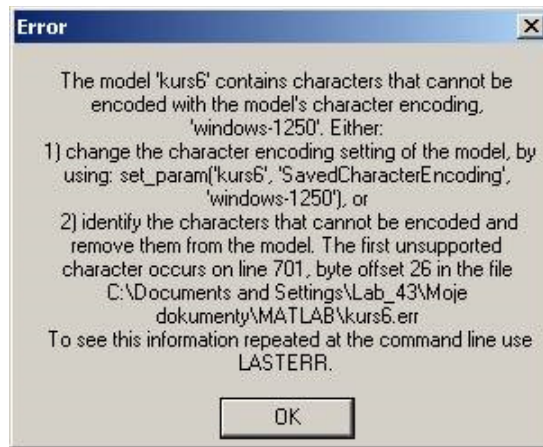
- Parameters – tu możemy ustawić parametry bloku. Gdy włączymy maskowanie, nasz subsystem zachowuje się jak funkcja w matlabie i posiada własny workspace, zatem musimy ustawić parametry A , B i f
- Initialization – tu możemy dodać dowolne komendy wywoływane w momencie startu bloku
- Documentation – jeśli nasz model zaczyna być skomplikowany powinniśmy zacząć do niego dokumentację, ta zakładka pozwala na proste jej tworzenie

Przykładowe wartości dla naszego modelu mogą wyglądać tak jak poniżej:



Rysunek 10: Zakładki okna maskowania podsystemu i przykładowe wartości

Należy pamiętać aby nie stosować polskich znaków, ponieważ może to spowodować pojawienie się błędu tego rodzaju:



Rysunek 11: Błąd spowodowany wystąpieniem polskiego znaku

W oknie *Mask help* możemy wpisać adresu pliku *.htm z pomocą. Np. Plik o kodzie źródłowym:

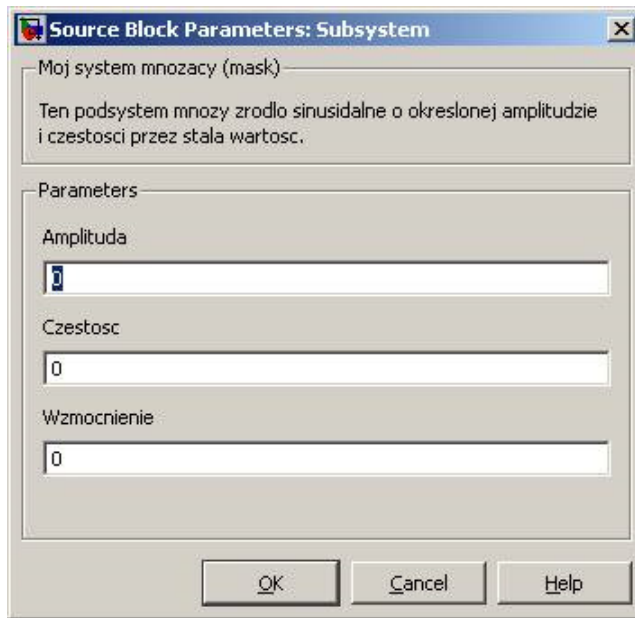
```
<html>
<head>
<meta http-equiv=Content-Type content="text/html; charset=windows-1250">
</head>
<body lang=PL style='tab-interval:35.4pt'>
<div class=Section1>
<h1><span class=SpellE>Subsystem</span> mnożący - pomoc</h1>
<p class=MsoNormal>Tu może się znajdować informacja opisująca dokładne
działanie naszego podsystemu.</p>
</div>
</body>
</html>
```

Wyświetli w pomocy do programu następującą informację:



Rysunek 12: Pomoc do podsystemu stworzona przez Użytkownika

Po wybraniu wszystkich opcji i zatwierdzeniu nasz model z nowym subsystemem jest gotowy do użycia, wystarczy jeszcze tylko zmienić domyślne wartości parametrów poprzez dwukrotne kliknięcie w blok:



Rysunek 13: Wybór parametrów podsystemu

... i uruchomić symulację :)

[Spis wszystkich lekcji.](#)

2010 – Grzegorz Knor