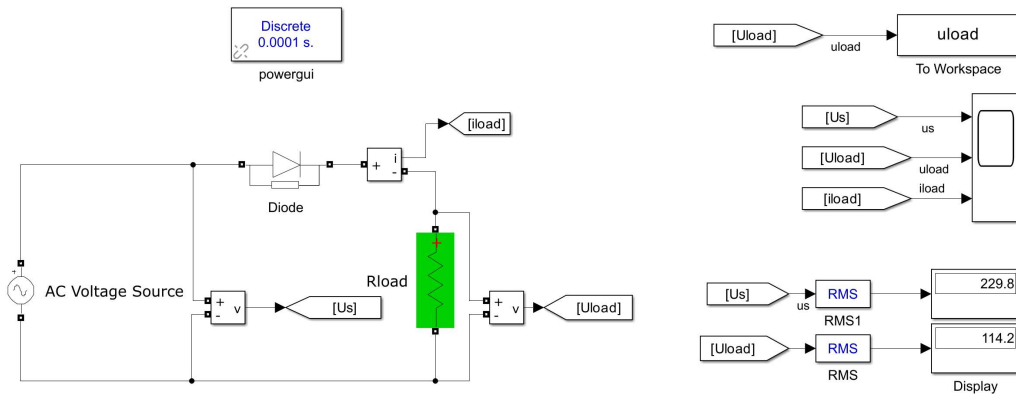


### ĆWICZENIE 3 „Prostownik diodowy jednofazowy”

#### 3.1 Budowa modelu jednofazowego prostownika diodowego zasilanego z sieci n.n.

Utworzyć nowy model w Simulinku i korzystając z dostępnych elementów bibliotecznych zbudować układ jak na rys 3.1. Przyjąć wartość  $R_{load}=2$ .

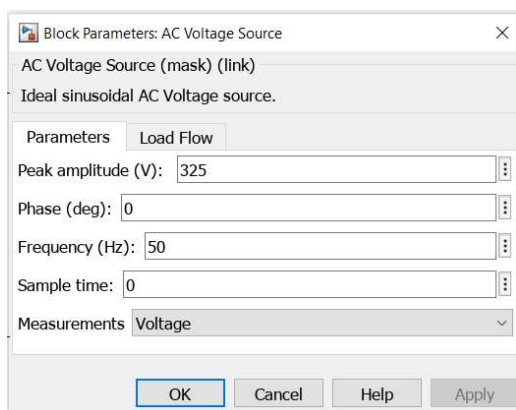


Rysunek 3.1

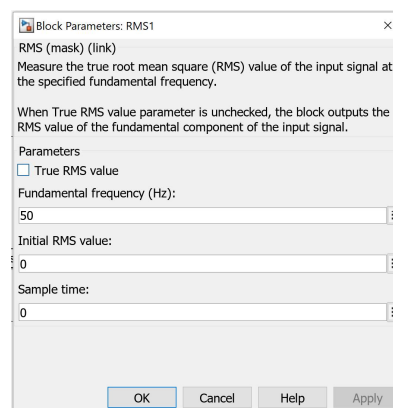
a) Ustawić parametry symulacji zgodnie z tabelą poniżej

<i>Solver type:</i>	Variable-step	Metoda całkowania
<i>Solver:</i>	<b>Ode23tb</b>	Typ metody
<i>Max Step size:</i>	<b>100e-6</b>	Maksymalny krok całkowania
<i>Min Step size:</i>	auto	Minimalny krok całkowania
<i>Initial step:</i>	auto	
<i>Stop time</i>	100e-3	Czas zakończenia symulacji

b) Ustawienia elementów



AC Voltage Source



RMS



*Uwaga: element z biblioteki  
Simscape/Electrical/Specialized Power  
Systems/Sensors and Measurements*

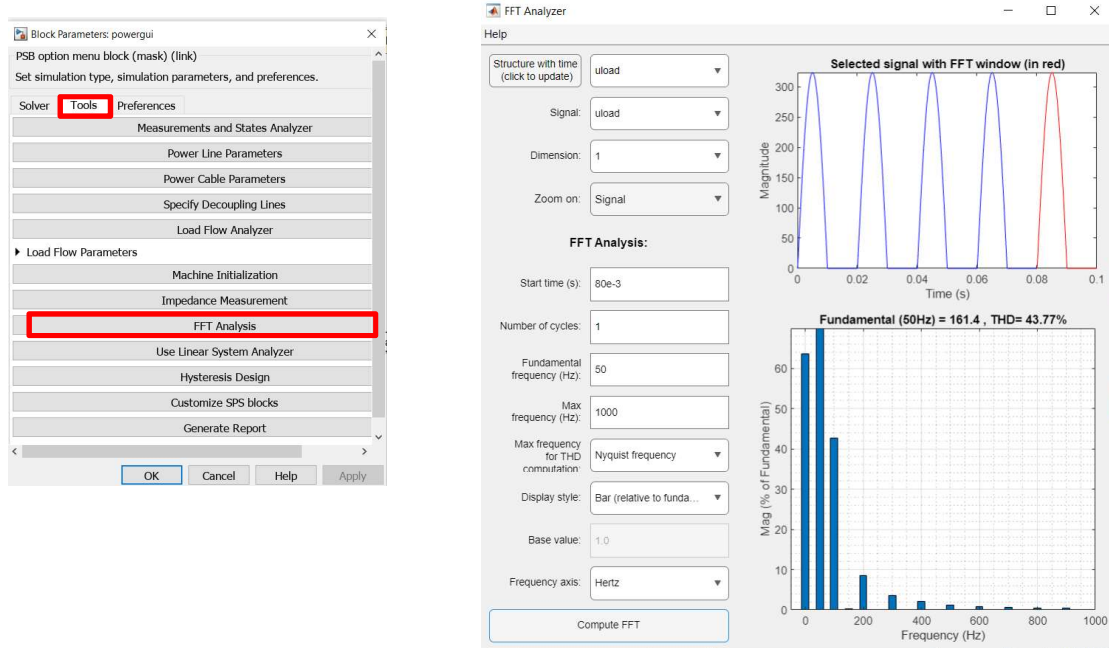
Przeprowadzić symulację w celu weryfikacji poprawności działania modelu

### ĆWICZENIE 3 „Prostownik diodowy jednopółwkowy”

#### 3.2 Pomiar podstawowych parametrów jakości energii elektrycznej

##### 3.2.1 Wykorzystanie elementu *Powergui* do pomiarów jakości energii elektrycznej

Klikając na element *Powergui*, wybierając zakładkę Tools, można między innymi, dokonać analizy FFT sygnałów dostępnych przestrzenie roboczej Matlaba (zapisanych jako *Structure with Time*).



Rysunek 3.2

**Zadanie 3.2.1** Wykorzystując element *Powergui* (ustawienia zgodnie z Rys. 3.2) dokonać analizy FFT sygnału napięcia obciążenia na prostowniku jednopółwkowym. Na podstawie wyników określić składowe harmoniczne widoczne w widmie (do 1kHz).

##### 3.2.2 Wykorzystanie funkcji *power\_fftscope* do wyliczania parametrów jakości.

```
%% KROK 1: Utworzenie struktury dla funkcji power_fftscope dla zapisanej w
workspace przykładowej zmiennej iload (w formacie Structure with time)
FFTDATA = power_fftscope(Uload); %
%% KROK 2: Ustawienie parametrów analizy - szukaj w HELP do funkcji
FFTDATA.fundamental=50; % ustawienie podstawowej harmonicznej sygnału (w Hz)
FFTDATA.startTime=80e-3;
FFTDATA.cycles=1;
% .....
%
%% KROK 3: ponowne przeliczenie funkcji
FFTDATA = power_fftscope(FFTDATA);
%% KROK 4: Wydobycie i obróbka uzyskanych wyników
FFT=FFTDATA.mag/FFTDATA.magFundamental*100;
figure()
bar(FFTDATA.freq,FFT);
grid on;
```

**Zadanie 3.2.2** Wykorzystując funkcję *power\_fftscope* (posiłkując się powyższy przykładem oraz Dokumentacją HELP dla funkcji) dokonać analizy FFT sygnału napięcia obciążenia i przedstawić go w formie wykresu słupkowego. Porównać uzyskane wyniki z wynikami z zadania 3.2.1.