The following code is an example of the style used throughout the simulator in where code allowed a quicker or precise alternative to block diagram implementation. The code below takes into account motor constants, speed and angular position to generate estimates of the BEMF produced by the motor at that instance. Input u1 provides the position of the rotor and calculates the BEMF for each phase [y1,y2,y3 being a,b,c respectably] based the total BEMF that is calculated using block diagram methods and fed into the code as input u2.

```
%{ Michael Thomas Ratcliffe m.ratcliife@lancaster.ac.uk

 This program is free software: you can redistribute it and/or
modify it under the terms of the GNU General Public License as
published by the Free Software Foundation, either version 3 of the
License, or(at your option) any later version. This program is
distributed in the hope that it will be useful but WITHOUT ANY
WARRANTY; without even the implied warranty of MERCHANTABILITY or
FITNESS FOR A PARTICULAR PURPOSE.  See the GNU General Public
License for more details:

<http://www.gnu.org/licenses/>. %}



function [y1,y2,y3] = BEMF(u1,u2) %%Declaring inputs and outputs

    if 0<=u1 && u1<60           y1=u2; %%Top of trapezoid
      y3=(((6*u2)/180)*(60-(u1)))-u2;
      y2=-u2;

    else if 60<=u1 && u1<120
y1=u2;
y3=-u2;
y2=((((6*u2)/180))*((u1)-(120)))+u2;

    else if 120<=u1 && u1<180
y1=(((6*u2)/180)*(180-(u1)))-u2;
y3=-u2;
y2=u2;


    else if 180<=u1 && u1<240
y1=-u2;
y3=((((6*u2)/180))*((u1)-(240)))+u2;
y2=u2;


    else if 240<=u1 && u1<300
y1=-u2;
y3=u2;
y2=(((6*u2)/180)*(300-(u1)))-u2;
```

```
    else if 300<=u1 && u1<361
y1=((((6*u2)/180))*((u1)-(360)))+u2;
y3=u2;
y2=-u2;


  else           %% This statement is included for error checking
y1=1000000000;
y2=1000000000;
y3=1000000000;

        end
        end
        end
        end
        end


    end
```