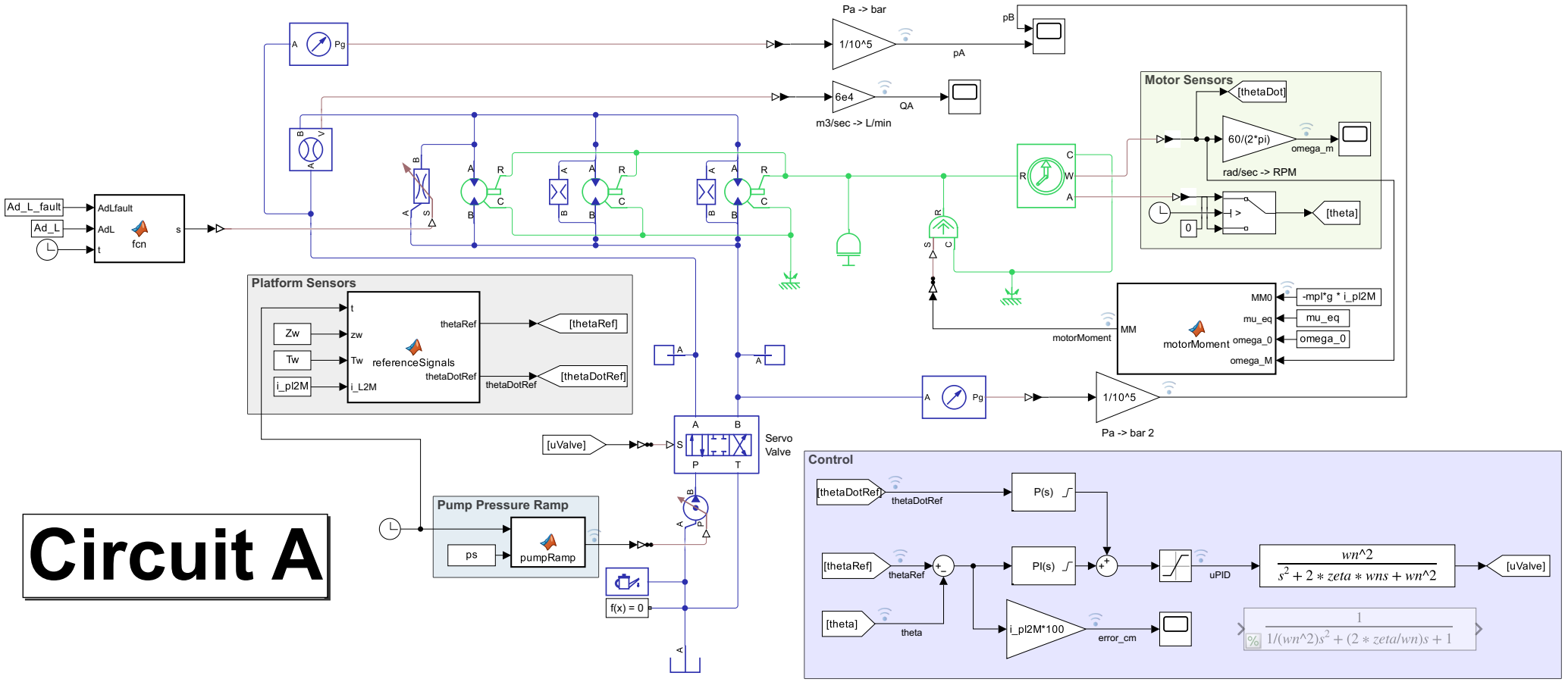
Circuit A
Ad_L_fault
Ad_L
t
AdLfault
AdL
fcn
s
Pg
Pa -> bar
1/10^5
pA
pB
6e4
QA
m3/sec -> L/min
Motor Sensors
[thetaDot]
60/(2*pi)
omega_m
rad/sec -> RPM
[theta]
0
Platform Sensors
Zw
Tw
i_pl2M
referenceSignals
thetaRef
thetaDotRef
[thetaRef]
[thetaDotRef]
motorMoment
MM
MM0
mu_eq
omega_0
omega_M
-mpl*g * i_pl2M
Pa -> bar 2
[uValve]
Servo Valve
Pump Pressure Ramp
ps
pumpRamp
f(x) = 0
Control
P(s)
PI(s)
uPID
$\frac{wn^2}{s^2 + 2 * zeta * wns + wn^2}$
i_pl2M*100
error_cm
$\frac{1}{1/(wn^2)s^2 + (2 * zeta/wn)s + 1}$

```
function s = fcn(AdLfault, AdL, t)

% if t<= 30
    s = AdL/AdLfault;
% else
%     s = 1;
% end
```

```
function [thetaRef,thetaDotRef] = referenceSignals(t,zw, Tw, i_L2M)

% Reference values for platform
zRef = zw * sin(2*pi/Tw * t); % [m]
zDotRef = 2*pi/Tw * zw * cos(2*pi/Tw * t); % [m/s]

% Reference values for motor
thetaRef = zRef/i_L2M; % [rad]
thetaDotRef = zDotRef/i_L2M; % [rad/sec]

% Envelope Functions
tEnv = 2*Tw; % [sec]
uEnv = t/tEnv; % [-]

if t < tEnv
    xEnv    = 3*uEnv^2    - 2*uEnv^3;
    xDotEnv = 6*uEnv/tEnv - 6*uEnv^2/tEnv;
else
    xEnv    = 1;
    xDotEnv = 0;
end

% Encorporating cubic polynomial into reference signals
thetaRef    = xEnv*thetaRef;
thetaDotRef = xEnv*thetaDotRef + xDotEnv*thetaRef;
```

```
function MM = motorMoment(MM0, mu_eq, omega_0, omega_M)

MM = MM0 * ( 1 + mu_eq * tanh(omega_M/omega_0) ); % [Nm]
```

```
function p = pumpRamp(t, pPump)

T = 0.2; % [sec]
tau = t/T; % [-]

if tau <= 1
    p = pPump*(3*tau^2 - 2*tau^3);
else
    p = pPump;
end
```