AER821 LAB 1

Question 1

Earth–Moon system setup

* We treat Earth and Moon as orbiting their barycenter (common center of mass).
* At time t = 0:  
  + Earth starts on the –x axis (slightly offset, since it’s heavier).
  + Moon starts on the +x axis (farther from the barycenter).
* Both move in counter-clockwise (CCW) circular orbits around the barycenter.

Spacecraft starting position

* We place the spacecraft on the +x side of Earth (so “to the right” of Earth at t=0).
* Its distance from Earth’s center is Earth’s radius + 35,786 km, which is a geostationary orbit altitude (GEO).
* At this altitude, a satellite’s orbital period matches Earth’s rotation (24 hours).

Spacecraft starting velocity

* The spacecraft is given the circular orbital speed it needs to stay bound to Earth.
* The velocity is pointed in the +y direction, so it orbits Earth counter-clockwise, just like satellites do in equatorial orbits.

Dynamics in the code

* At each step, the code calculates the positions of Earth and Moon (using cos/sin with angular velocity Ω).
* It then computes the gravitational pull on the spacecraft from both Earth and Moon.
* The spacecraft’s trajectory is integrated over one lunar cycle (~27 days for a Sidereal month).

What the plot shows (Inertial/Barycenter frame):

* Earth (blue) makes a tiny circle around the barycenter.
* Moon (black) makes a large circle around the barycenter.
* The spacecraft (red) is in orbit around Earth, but since Earth itself moves around the barycenter, the red orbit gets “carried along.”
* That’s why the red path resembles a stack of loops.

Simulink

The core of the model is the MATLAB Function block (accel), which calculates the spacecraft’s acceleration from Earth’s and Moon’s gravity.

Integrators (1/s blocks):

* The acceleration is integrated once to get velocity.
* The velocity is integrated again to get position.

Initial conditions:

* Position integrators start at the spacecraft’s initial coordinates (r0x, r0y, r0z).
* Velocity integrators start at the spacecraft’s initial velocity (v0x, v0y, v0z).

A Mux block groups x, y, z into a single position vector, which the accel block uses.

The Clock block provides simulation time t to the accel block (needed for Earth and Moon positions).

To Workspace blocks (x\_sc, y\_sc) save the simulation results (position vs. time) into MATLAB so we can plot them.