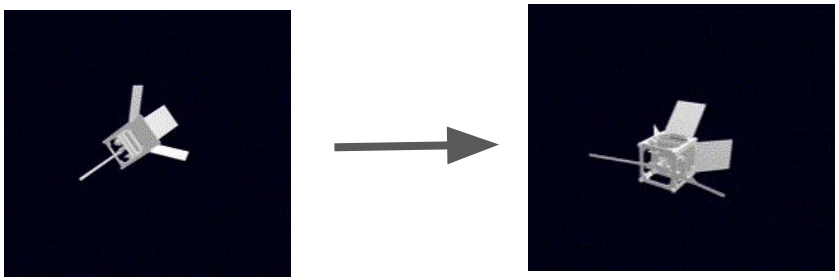
Weekly Quad Chart - 8/11

Updates



Blockers and Requirements

Blockers

- Jetson Orin for power consumption of EKF, MEKF on different modes and frequencies
- Wire AWG selection and material for Helmholtz cage

Requirements

-

Weekly Results and Plan

Weekly Results

- Studied MEKF course resources and papers
- Started solving combined-estimation problem for attitude and orbit determination
- Leveraging Brahe for the final sim. The python library did not support drag computation. Added local atmospheric density calculation using the NRLMSISE00 atmosphere model.
- Created the visualization for a detumbling satellite using the sim.

Next week

- Improve on orbit estimation by adding more complexity
- Debug MEKF and study more resources
- Continue solving combined estimation problem for attitude and orbit determination
- Continue working on CAD design for the cage
- Implement SGP4 Orbit propagation utilizing TLE data.

Interface dependencies

Avionics:

- Fprime development on Pycubed; Communication with Jetson for estimation

Mechanical:

-

Vision:

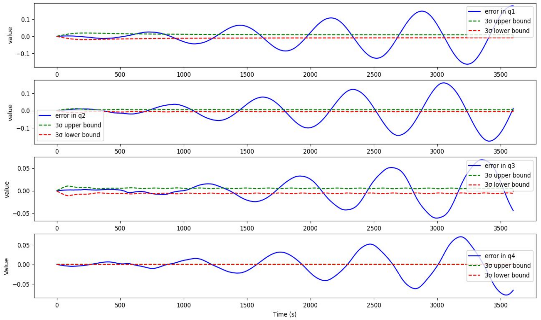
- Discuss any major changes or updates to output of vision system

COMOPS:

- Communication protocol & Commands

Weekly Quad Chart - 1/11

Updates



Incorrect MEKF error plots : error builds

```yaml
sc_properties:
  inertia_rows:
    - [1.959e-4, 2016.333e-9, 269.176e-9]     # kg·m²
    - [2016.333e-9, 1.999e-4, 2318.659e-9]    # kg·m²
    - [269.176e-9, 2318.659e-9, 1.064e-4]     # kg·m²
  mass:  177.808e-3                           # kg
  area:  0.01                                 # m²
  drag_coeff:    1.5                          # ()

gravity:
  spherical_harmonic_gravity_bool: true       # ()
  gravity_order:  6                           # ()
  gravity_degree: 6                           # ()

environment:
  drag_bool:               true               # ()
  srp_bool:                true               # ()
  gravity_gradient_bool:   true               # ()

initial_orbital_elements:
  sma:  6.9281363e6                           # m
  ecc:  0.00069140                            # ()
  inc:  1.7209103504296768                    # rad      (98.6009°)
  raan: 2.22778014514761                      # rad      (127.6424°)
  argp: 1.6058739541014786                    # rad      (92.0098°)
  M:    4.680781067631073                     # rad      (268.189°)

initial_attitude_conditions:
  q0: [0,0,0,1.0]                             # ()       (ᴺqᴮ)
  w0: [-2.41, 3.7092, -1.62]                  # (rad/s)  (ᴺωᴮ, expr. B)

propagator_parameters:
  epoch: "2018-12-01 16:22:19.0 GPS"          # Epoch (SatelliteDynamics)
  dt_orbital:      10.0                       # sec
  dt_attitude:     0.05                       # sec
  dt_controller:   0.1                        # sec
  time_final: 0.1                             # days

sensor_properties:
  gyro:
    rotate_std_deg: 0.0
    noise_std_degps: 0.22
    initial_bias_deg: 2.0
    bias_noise_std: 1e-2
```

Sample config file for sim

Blockers and Requirements

Blockers

- Jetson Orin for power consumption of EKF, MEKF on different modes and frequencies
- Wire AWG selection and material for Helmholtz cage

Requirements

- Crash course by Zac on MEKF

Weekly Results and Plan

Weekly Results

- Studied MEKF course resources and papers
- Implemented MEKF
- Compute benchmarking on Raspberry Pi
- SW Eng work for simulator
  - Defined modular structure for sim. Came up with yaml config file based approach for simulation configuration,initialization and runs.
  - Implemented config parser script and created sample config file.

Next week

- Improve on orbit estimation by adding more complexity
- Debug MEKF and study more resources
- Generate simulated ground truth data for MEKF with angular velocity dynamics
- Power analysis on Jetson
- Continue finalize the cage simulation by adding wire awg information

Interface dependencies

Avionics:

- Fprime development on Pycubed; Communication with Jetson for estimation

Mechanical:

-

Vision:

- Discuss any major changes or updates to output of vision system

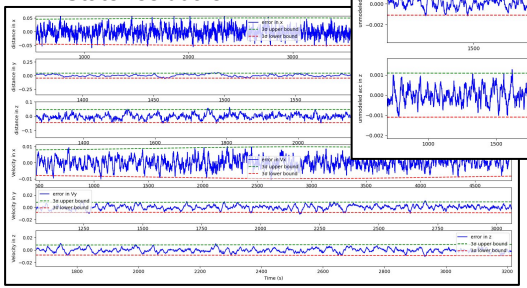
COMOPS:

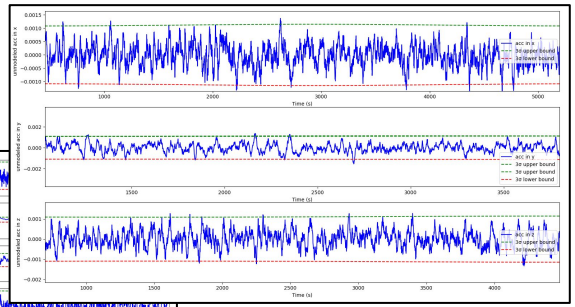
- Communication protocol & Commands

Weekly Quad Chart - 25/10

Updates

State residuals





Unmodeled acceleration residuals

Blockers and Requirements

Blockers

-

Requirements

- Crash course by Zac on MEKF
- Pycubed board to start twiddling with software frameworks

Weekly Results and Plan

Weekly Results

- Implemented EKF for orbit estimation following version 1 method
- Implemented EKF for orbit estimation with unmodelled acceleration in states
- Started the design of HIL devices for magnetometer, magnetorquer and sun sensor
- Explored slew maneuvers using magnetorquers.

Next week

- Improve on orbit estimation
- Implement simple MEKF

Interface dependencies

Avionics:

- Finalize computer framework and sensor selections

Mechanical:

-

Vision:

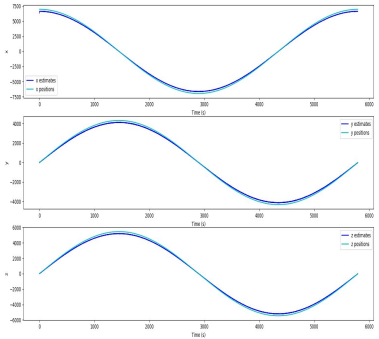
- Discuss any major changes or updates to output of vision system

COMOPS:

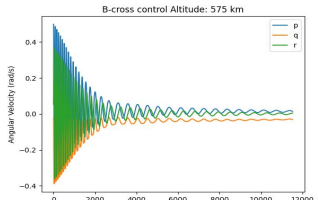
- Communication protocol & Commands

Weekly Quad Chart - 11/10

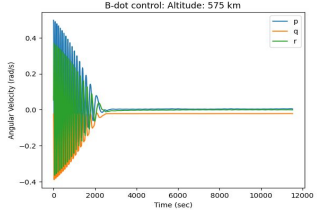
Updates



Kalman filter for orbit estimation



B-cross control for detumbling



B-dot control for detumbling

Blockers and Requirements

Blockers

-

Requirements

- Crash course by Zac on MEKF
- Pycubed board to start twiddling with software frameworks

Weekly Results and Plan

- Implemented Kalman filter for orbit estimation
- Implemented b-cross control for detumbling. Ran some experiments to compare the performance of b-dot and b-cross control
- Started listing risks and potential mitigations
- Fprime compatible with PyCubed board

Next week

- Implement simple MEKF
- Formally compare the performance of the detumbling control techniques in a Monte carlo sim.
- Continue risk analysis
- Basic workflow/state machine using Fprime on PyCubed
- Start evaluating options for estimation architecture with vision inputs
- Keep implementing more risk analysis

Interface dependencies

Avionics:

- Finalize computer framework and sensor selections

Mechanical:

-

Vision:

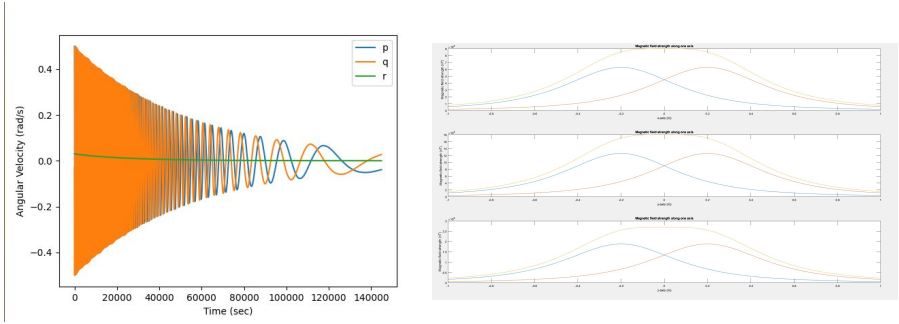
- Discuss any major changes or updates to output of vision system

COMOPS:

- Communication protocol & Commands

Weekly Quad Chart - 4/10

Updates



Implemented a detumbling simulation with a BDot controller.
Magnetic field in 3-axis Helmholtz coils

Blockers and Requirements

Blockers

-

Requirements

- Crash course by Zac on MEKF
- Pycubed board to start twiddling with software frameworks

Weekly Results and Plan

Weekly Results

- Solved wabhas problem with CVX, SVD, q-method and gauss newton method
- Implemented a detumbling simulation with a BDot controller. Sim models translational and rotational kinematics and dynamics for the cubesat.
- Finished the Helmholtz coil's matlab simulation, which can give the magnetic field on 3-axis

Next week

- Implement simple MEKF
- Add sensor noise, bias, sensor filtering for sim

Interface dependencies

Avionics:

- Finalize computer framework and sensor selections

Mechanical:

-

Vision:

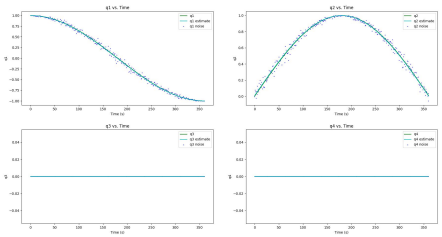
- Discuss any major changes or updates to output of vision system

COMOPS:

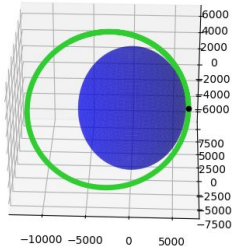
- Communication protocol & Commands

Weekly Quad Chart - 27/09

Updates



Simple attitude determination in quaternions



Simulating orbit propagation with two-body dynamics

Blockers and Requirements

Blockers

-

Requirements

- Crash course by Zac on MEKF
- Pycubed board to start twiddling with software frameworks

Weekly Results and Plan

Weekly Results

- Implemented simple EKF for attitude determination
- Studied and implemented quaternion operations in python required for MEKF
- Trade study between CircuitPython vs FPrime
- Simulated orbit propagation with two-body dynamics
- Studied HIL testing for magnetometer, magnetic torquer, and IMU

Next week

- Study attitude determination course notes
- Implement first version of attitude estimator
- Start playing around with chosen architecture
- Start designing the Helmholtz cage for HIL simulation

Interface dependencies

Avionics:

- Finalize computer framework and sensor selections

Mechanical:

-

Vision:

- Discuss any major changes or updates to output of vision system

COMOPS:

- Communication protocol & Commands

Weekly Quad Chart - 20/09

Updates

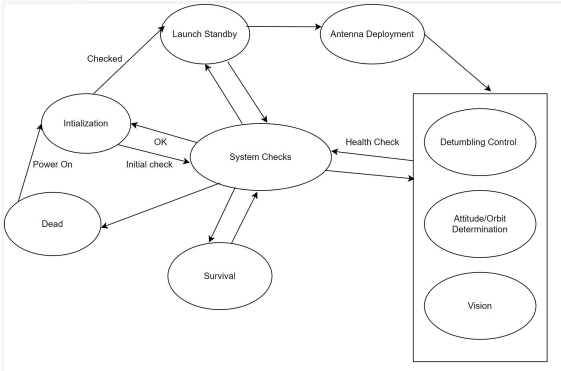


Figure: Basic state machine diagram

Blockers and Requirements

Blockers

-

Requirements

- Recorded satellite sensor(sun,magnetometer) measurements
- Crash Course by Zac

Weekly Results and Plan

Weekly result

- Studied resources on quaternions and MEKF
- Initiated development on attitude estimation with basic quaternion operations
- Developed basic state machine
- First version of simulation for satellite position based on Newton's law of gravity and RK4

Next week

- Implement simple attitude estimator
- Progress on subsystem interfaces
- Meeting with Comms for functional partitioning and protocol
- Meeting with Avionics for hardware choices
- Iterate on State machine design and simulation

Interface dependencies

Avionics:

- Software framework/ baseline computer system
- Kernel functionality
- Sensors & Actuators drivers

Mechanical:

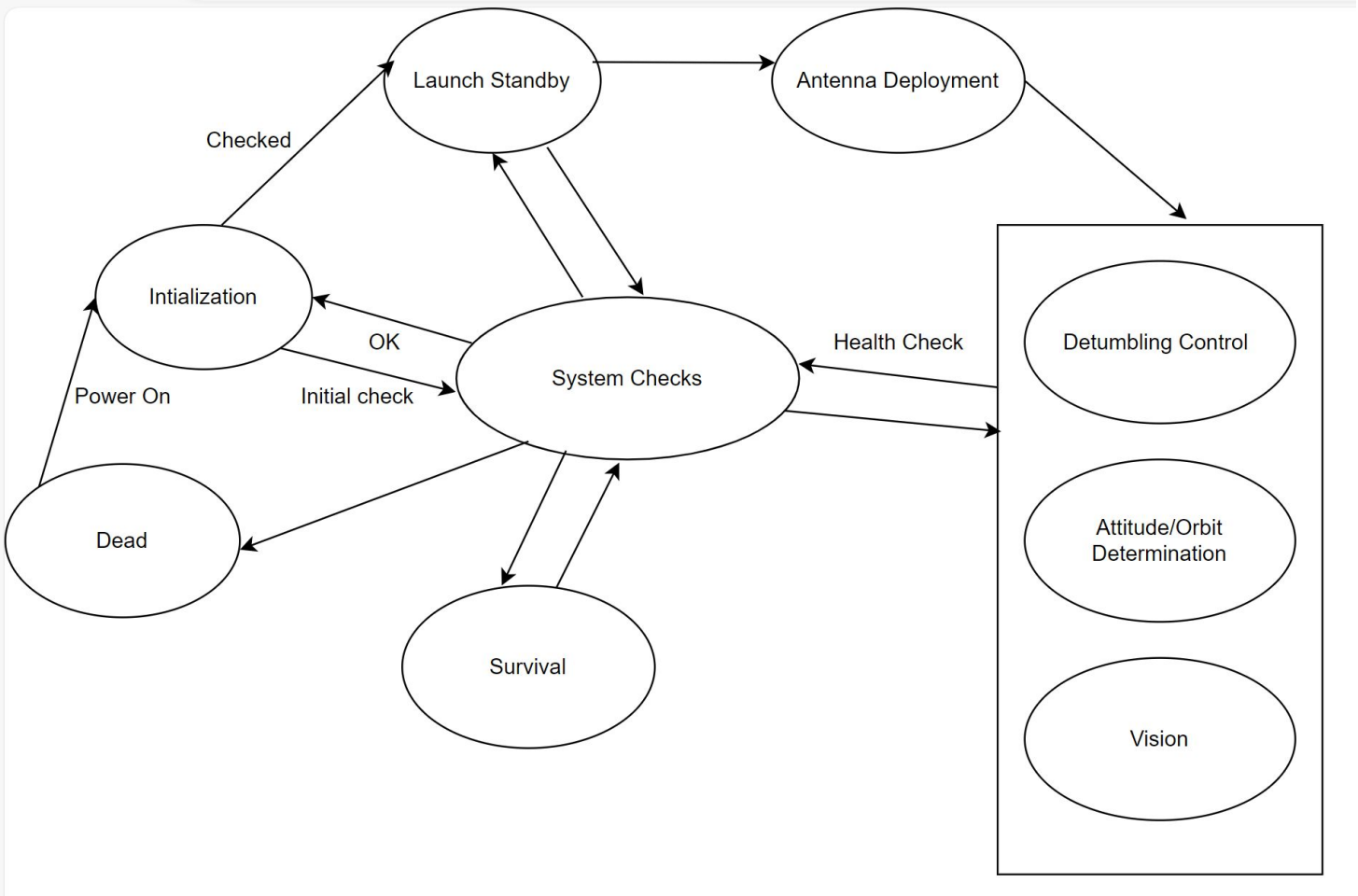
- Mass estimate
- Moment of inertia
- Mechanical layout

Vision:

- Discuss any major changes or updates to output of vision system

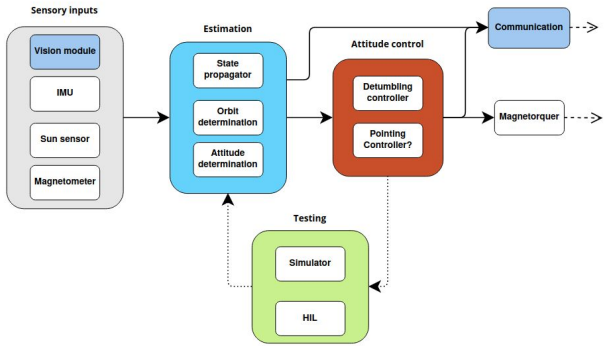
COMOPS:

- Communication protocol & Commands


Launch Standby
Antenna Deployment
Checked
Intialization
OK
Initial check
Power On
System Checks
Health Check
Detumbling Control
Dead
Attitude/Orbit
Determination
Survival
Vision

Weekly Quad Chart - 12/09

Updates



- Developed first draft of attitude and orbit determination architecture

Blockers and Requirements

Blockers

- Coming up with specific numbers in the requirements (sensors)
- Lack of background in orbital and attitude dynamics (simulation,estimation,attitude control)

Requirements

- None

Weekly Results and Plan

Weekly result

- Refined the level 2 requirements
- Developed initial block diagram for estimation
- Studied material about Kalman filter, spacecraft attitude determination and control system
- Preliminary actuator and sensor selection

Next week

- Develop milestone chart
- Study resources given by Zac on attitude determination and (M)EKF
- Start development of simple EKFs
- Simulation rigid-body dynamics (2 body and attitude)
- General flight software architecture
- Design first draft of cyber physical architecture for estimation module
- Create first draft of software design document for attitude control

Interface dependencies

Avionics:

- Software framework/ baseline computer system
- Power budget?
- Sensors & Actuators drivers

Mechanical:

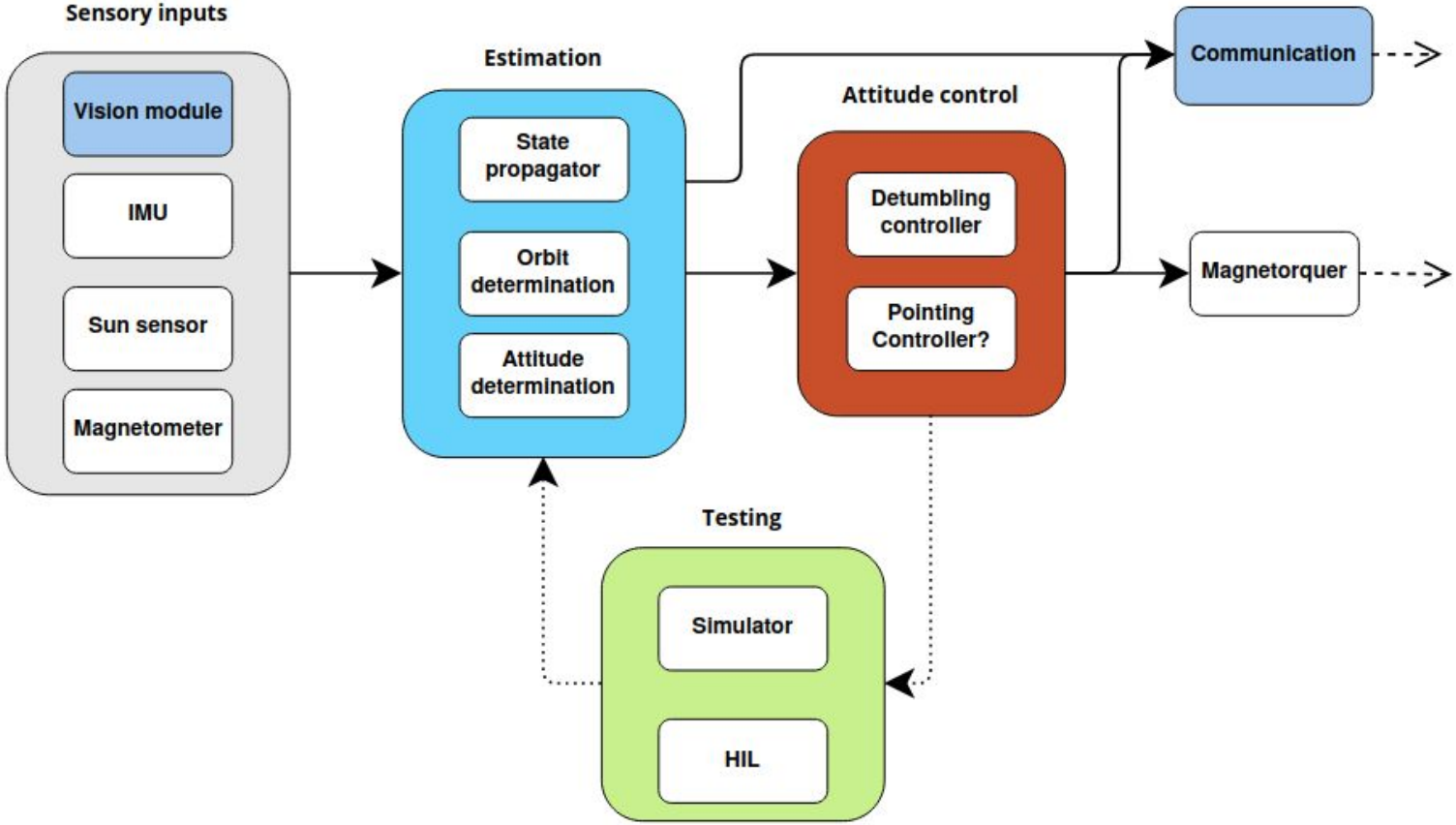
- Mass estimate
- Moment of inertia
- Mechanical layout

Vision:

- Vision system output to finalise orbit and attitude determination design architecture. Set up meeting this week

COMOPS:

- Type of antenna and influence on pointing requirement


Sensory inputs
Vision module
IMU
Sun sensor
Magnetometer
Estimation
State propagator
Orbit determination
Attitude determination
Attitude control
Detumbling controller
Pointing Controller?
Communication
Magnetorquer
Testing
Simulator
HIL