# Design and Control of the BlueFoot Platform: A Multi-terrain Quadruped Robot

Brian Cairl
Dr. Farshad Khorrami

May 13, 2015

# Overview

- Introduction
- Hardware and Design
- Software Architecture
- System Modeling
- Gait and Stability Control
- Navigation Control
- Concluding Remarks

# Introduction

# Quadruped Robotics

- Inspired by notable quadruped robotic systems from the past decade:
  - LittleDog (Boston Dynamics)
  - BigDog (Boston Dynamics)
  - Tekken (Kyoto Institute of Technology)
  - Kolt (Stanford University)
  - HyQ (Istituto Italiano di Tecnologia)

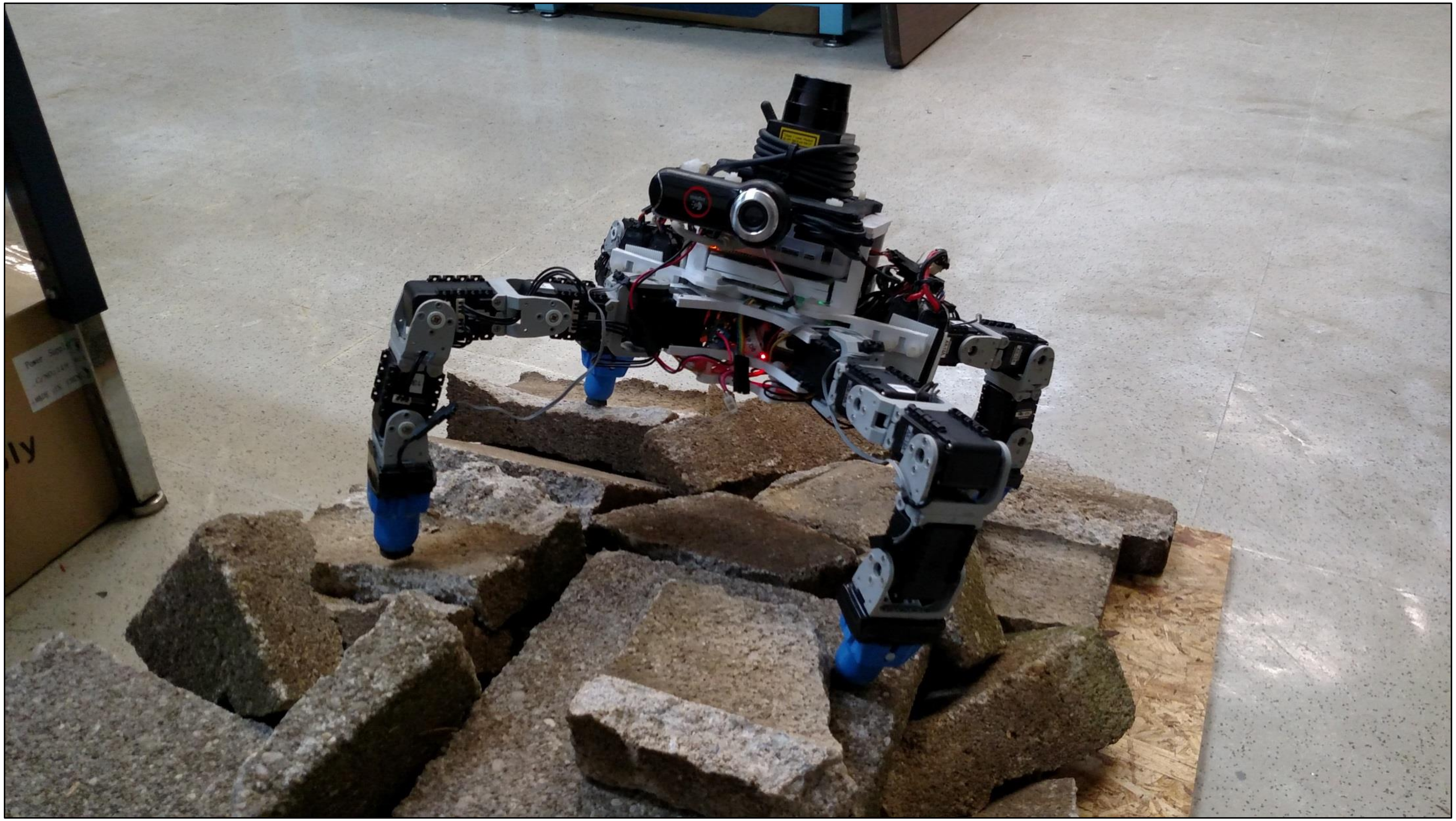
# Quadruped Robotics

- Well known advantages over wheeled robots, mainly in adaptability to terrain
- At the follows expenses:
  - Higher power consumption (more actuators)
  - Lower payload capacity (higher actuator loading)

# Quadruped Robotics

- Applications:
  - Gaiting design/research
  - Rough-terrain navigation/planning
  - Disaster recovery
  - Search and rescue
  - Environmental mapping

# The BlueFoot Platform



# The BlueFoot Platform, cont.

- Small-scale quadruped robot
- 16 actuated degrees of freedom
- High Dexterity, which lends itself to:
  - Stabilization and repositioning on variable terrain
  - Large range of trunk articulation
  - Ability to overcome raised/uneven terrain

# Overview of Control Strategies

- Gaiting and Stability
  - Central Pattern Generator (CPG)-based gait control
  - Zero-Moment Point (ZMP) body posture control
  - Virtual-Force foothold controller
  - NARX-Neural Network trunk-leveling controller

# Overview of Control Strategies, cont.

- Navigation Control:
  - Potential-Fields/Visual-Servoing
  - Surface Reconstruction for rough terrain navigation
    - Composing 3D point clouds from 2D LIDAR scans
    - Height-map Surface Representation
    - Surface (Normal) Estimation from 3D point clouds
  - Rough terrain planning using Height-maps

# System Design

# Design Overview

- Modularity between body segments
  - Robot designed with 3 main sections

- 3D printed structural components
  - Lightweight
  - Rapid design iterations
  - Facilitates modular design

# Core Devices

- AutoPilot
  - TM4C MCU (80 MHz) and RM48 MCU (220 MHz)
  - Inertial Measurement Unit (IMU)
    - 3-axis accelerometer (x2)
    - 3-axis gyroscope
  - Magnetometer
  - Pressure/Humidity Sensor

# Core Devices, cont.

- ODROID-XU computer
  - 1.6 Ghz, quad-core processor
  - 2 Gb of RAM
  - WiFi Antenna
- Logitech9000 Web Camera
  - 30 fps
  - 1280 by 720 max resolution

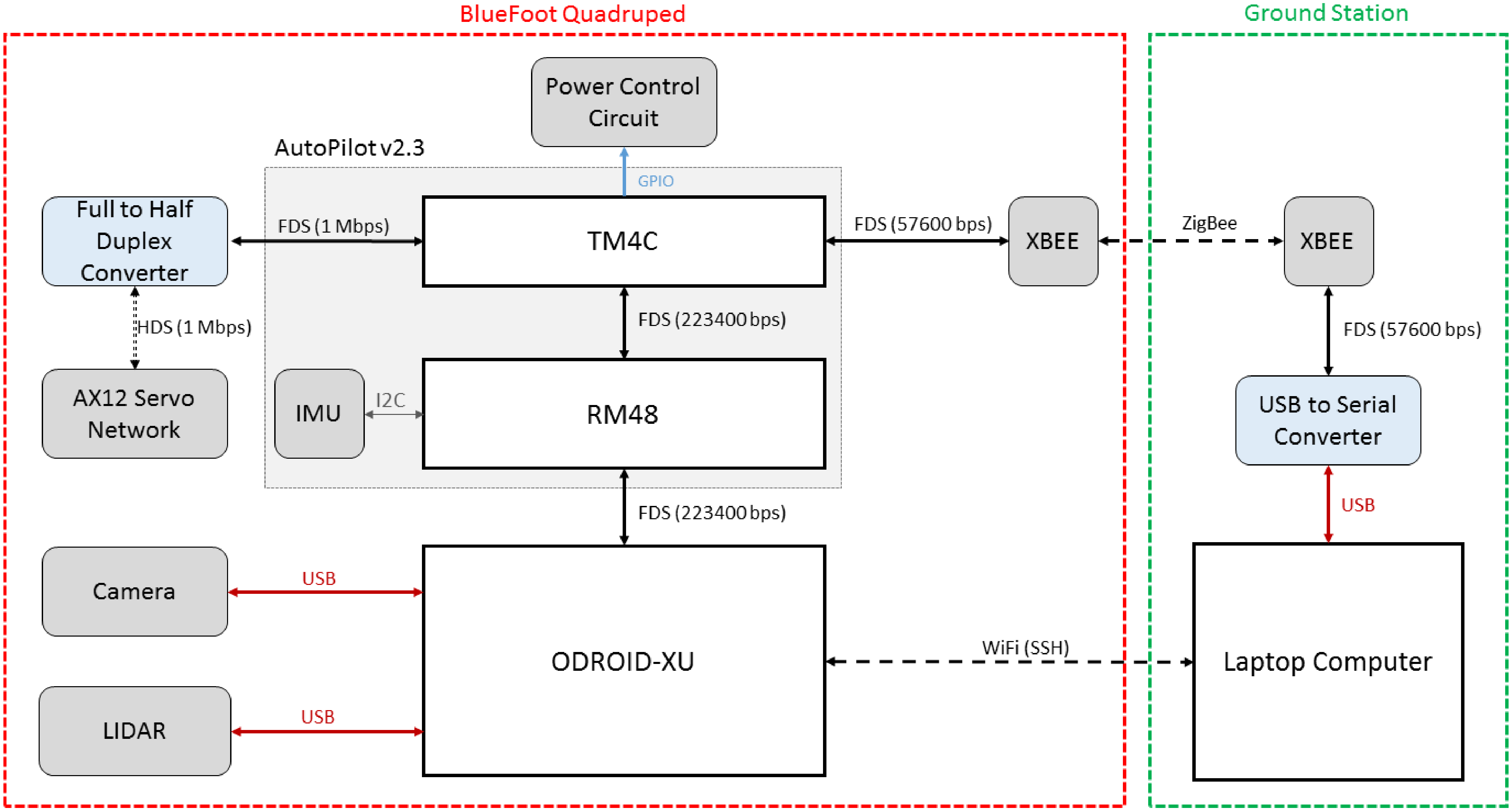
# Core Devices, cont.

- Hokuyo-URG LIDAR
  - 0.3822 degree scan resolution (628 points per scan)
  - 5.6 meter range
- Dynamixel AX12 Smart Servos (x16)
  - Position, Velocity, Torque-Loading Feedback
  - 1 Mbps data transfer rate
  - High-gain proportional controller

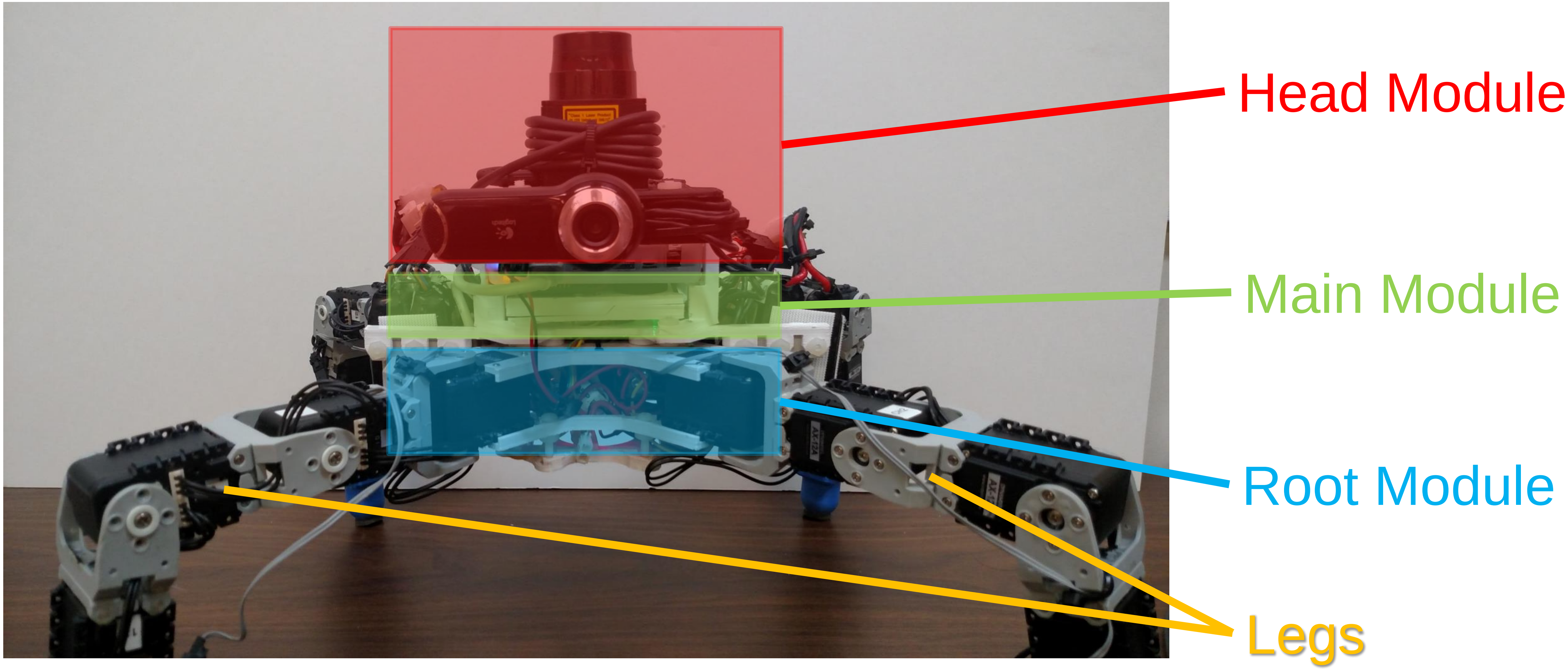
# Core Devices, cont.

- Binary foot-contact sensors
- XBEE Wireless Radio
  - 27 meter outdoor range
  - 115 kbps max data transfer rate
  - 57.6 kbps nominal data transfer rate

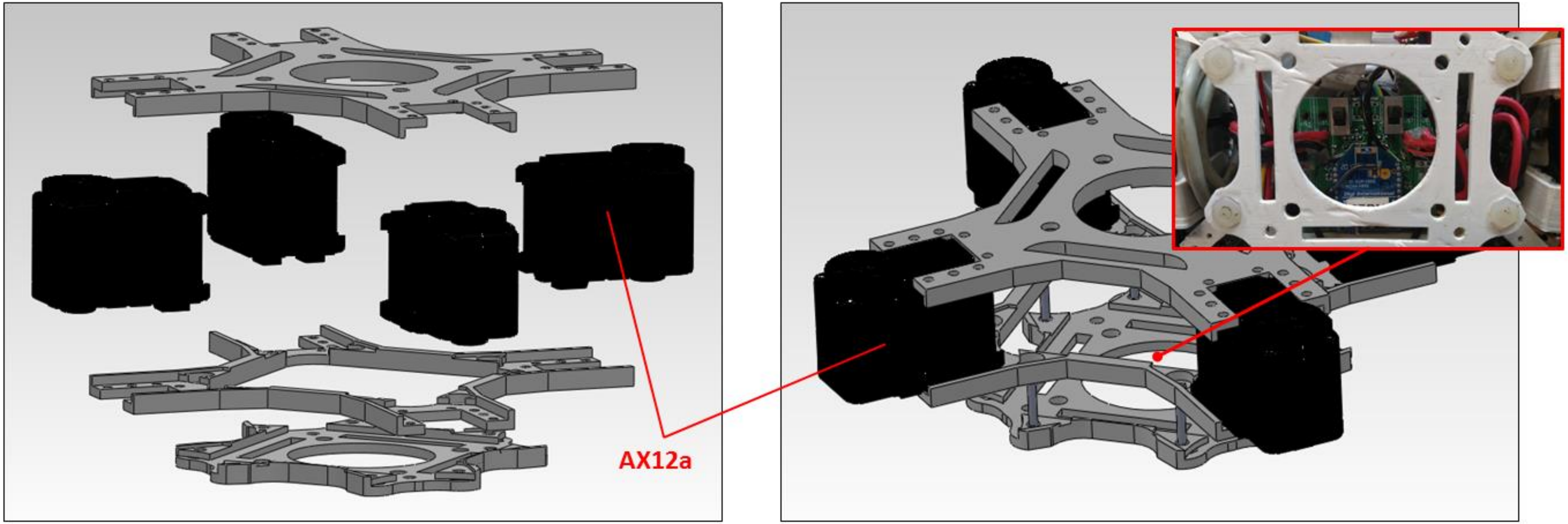
# Device Interfacing



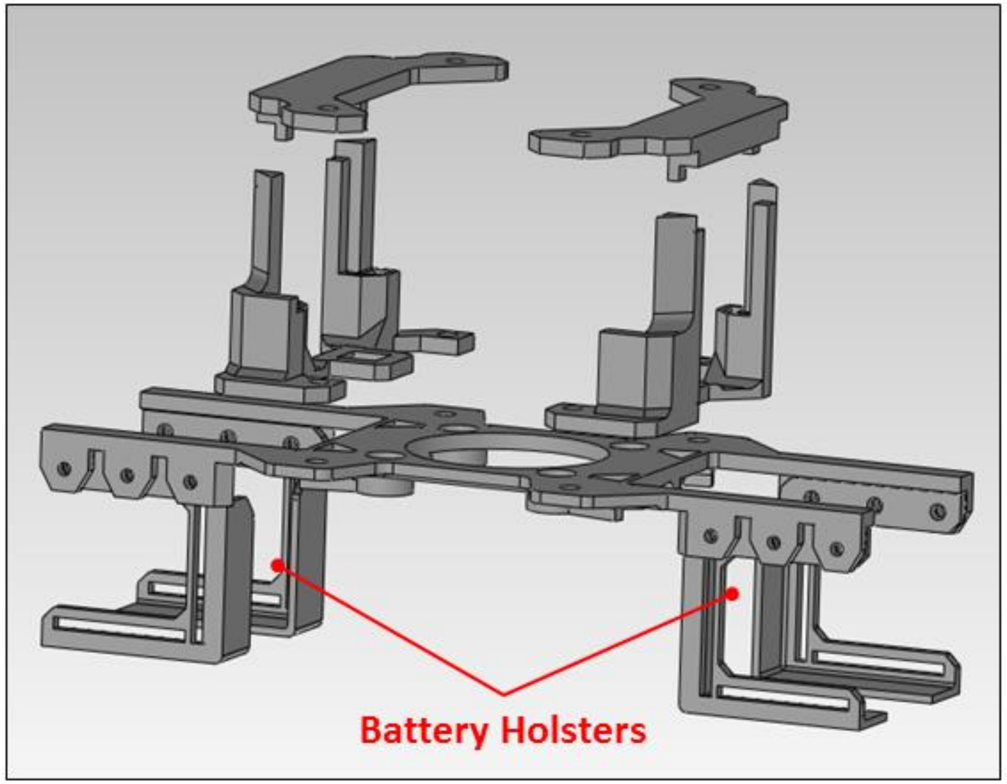
# Structural Layout

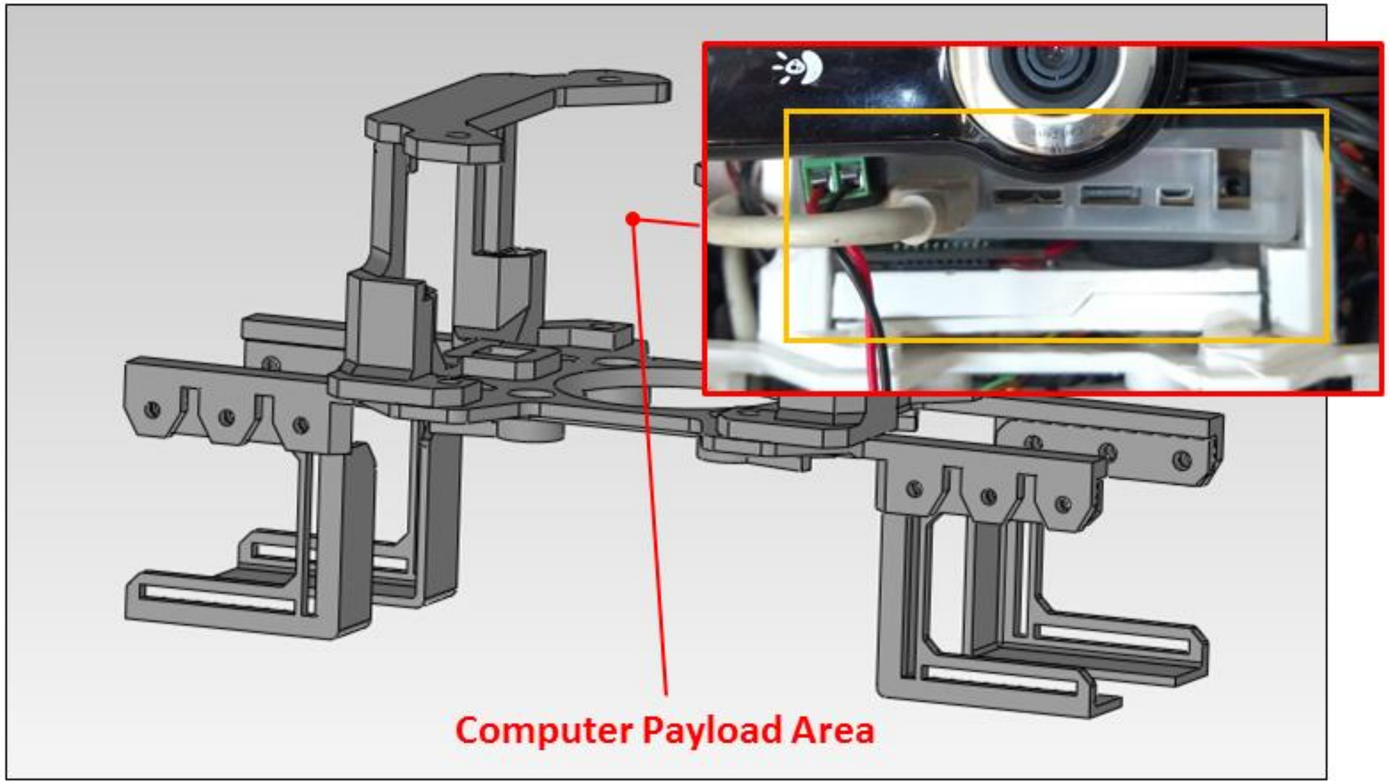


# Lower Body (*Root Module*)

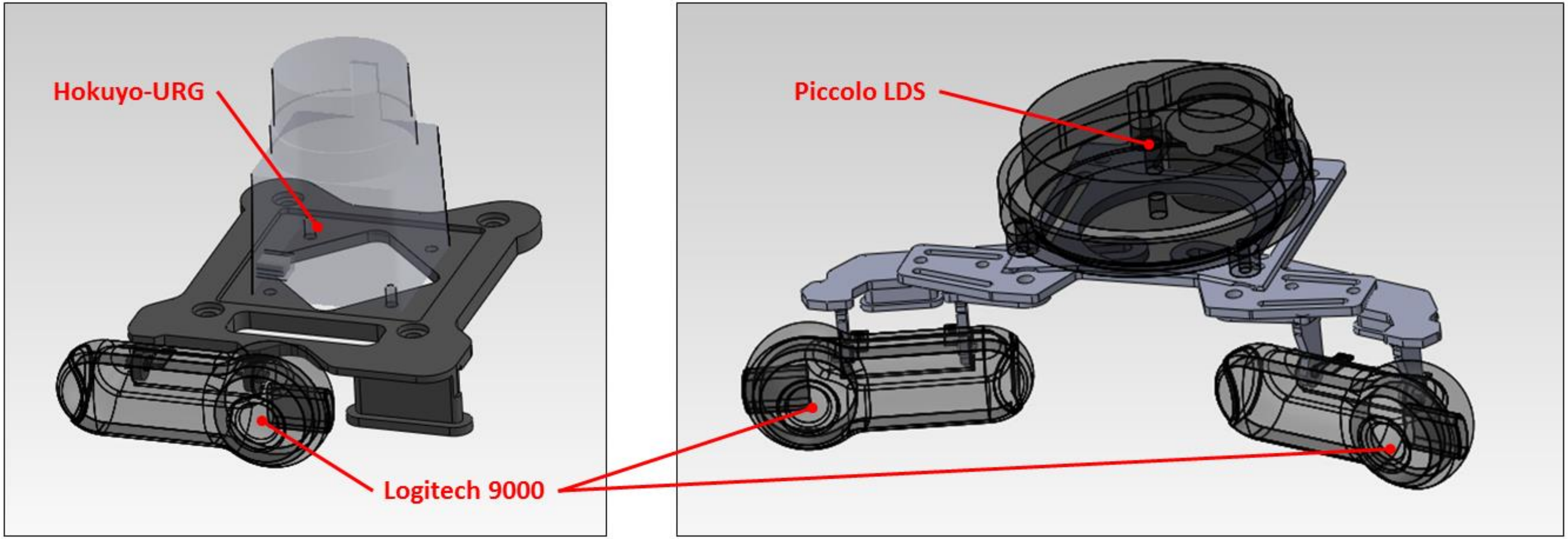


# Mid Body (*Main Module*)

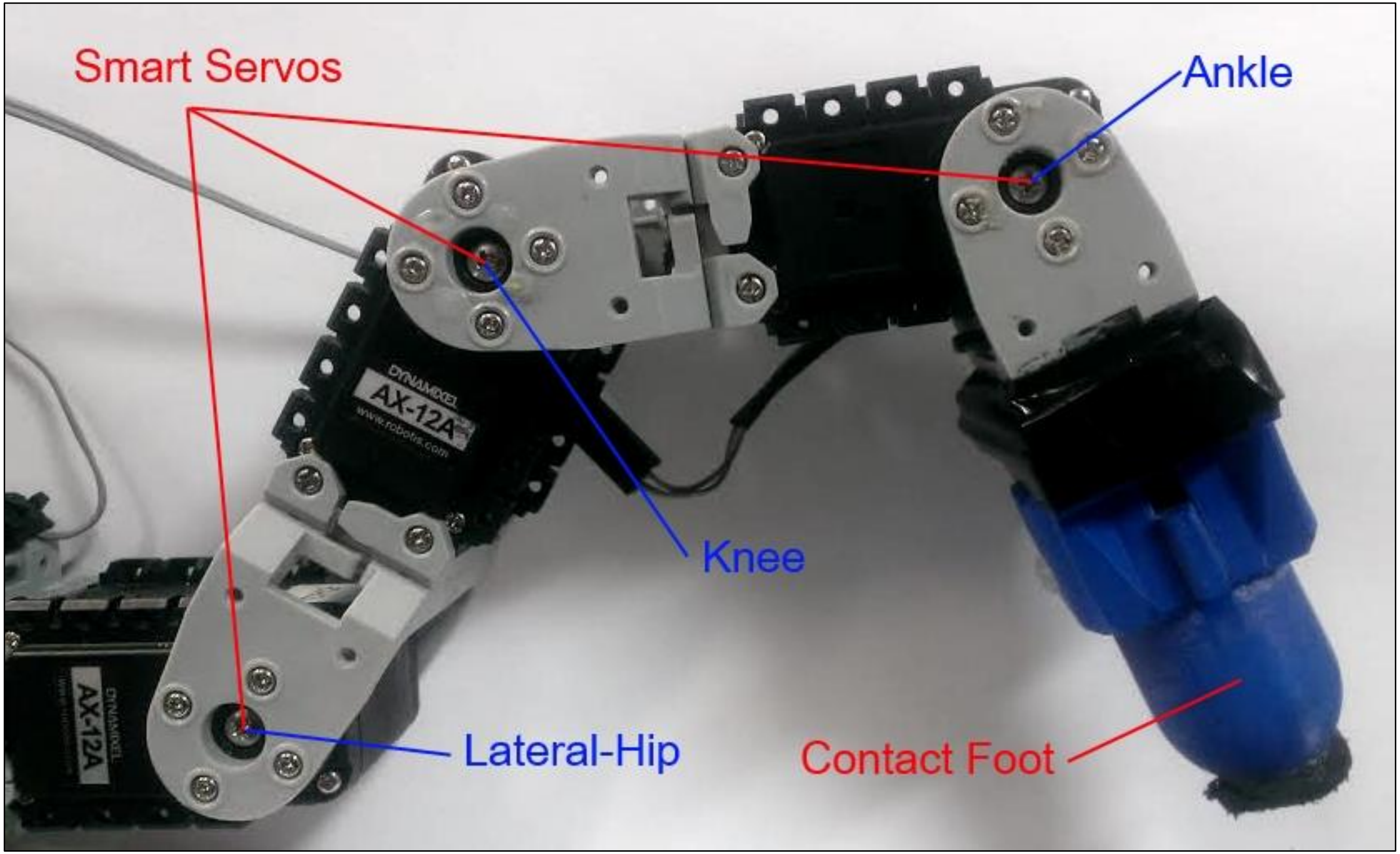


# Upper Body (*Head Module*)

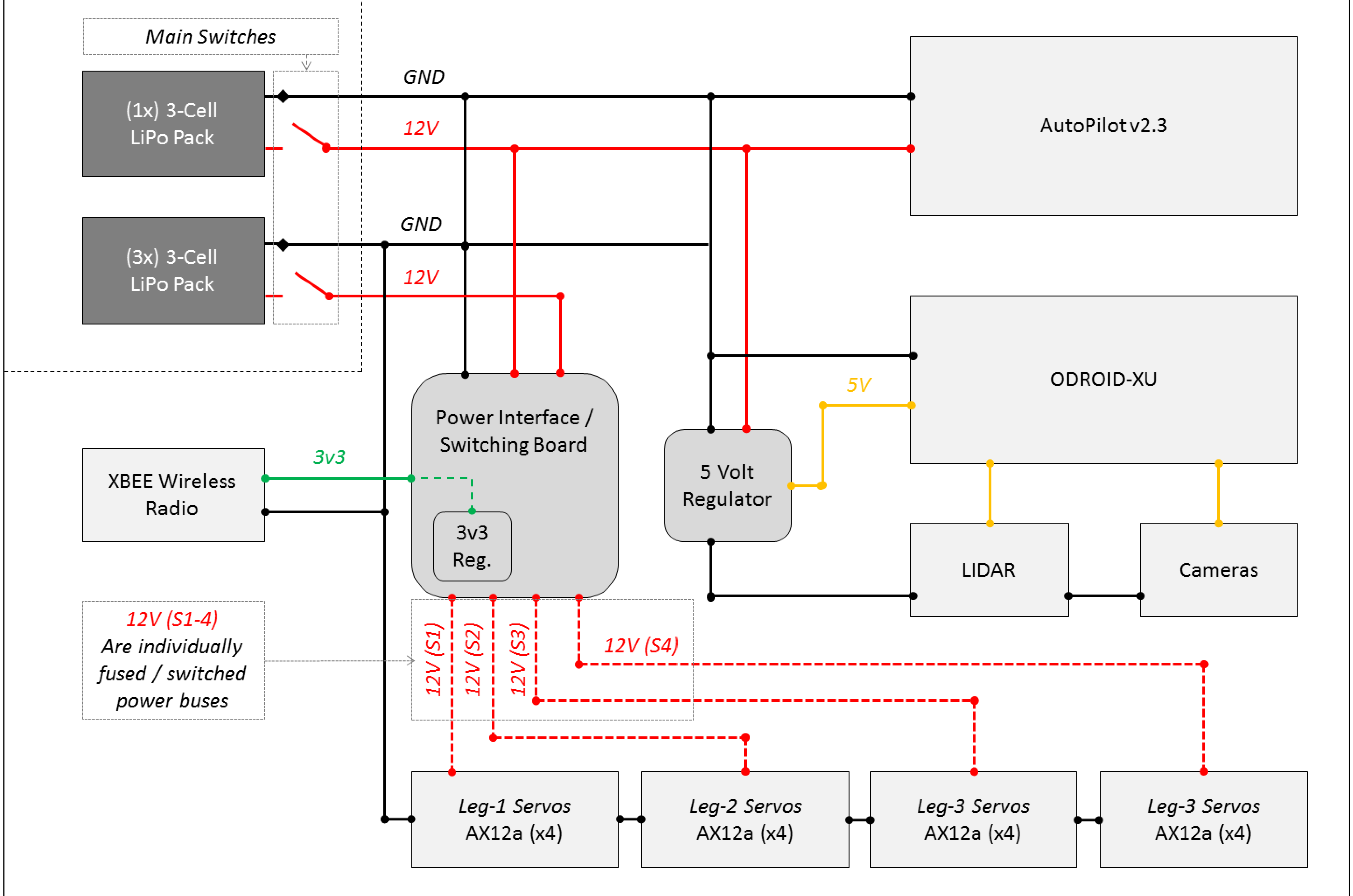


# Leg Design


Smart Servos
Ankle
DYNAMIXEL
AX-12A
Knee
Lateral-Hip
Contact Foot

# Power Requirements

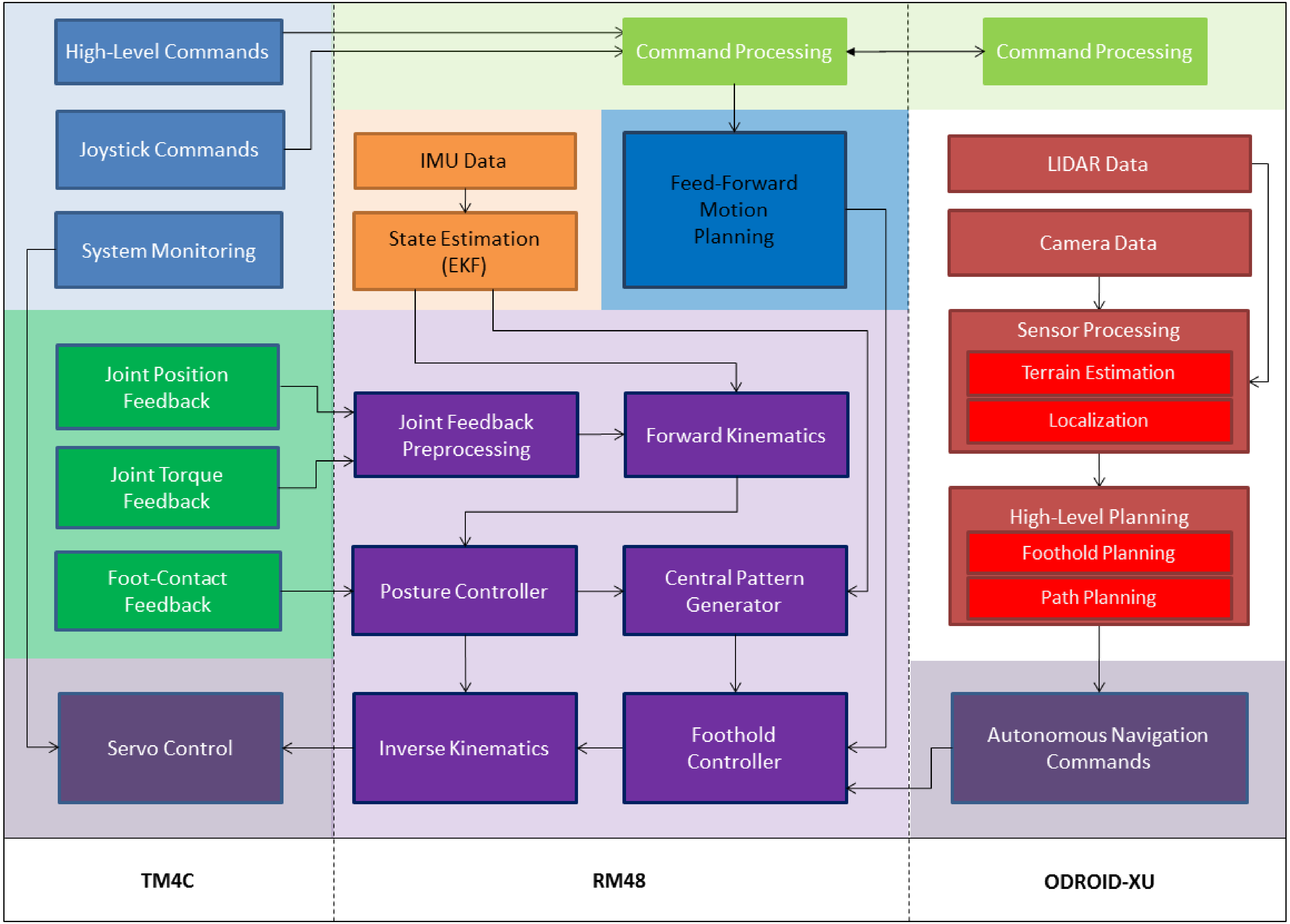
| Net | Device | $V_{op}, V$ | $I_{nom}, A$ |
|---|---|---|---|
| 1 | AutoPilot (RM48, TM4C) | 12.0 | 0.40 |
| 1 | XBEE Radio | 3.3 | 0.25 |
| 1 | ODROID-XU | 5.0 | 2.0 |
| 1 | Logitech-9000 | 5.0 | 0.1 |
| 1 | Hokuyo-URG | 5.0 | 0.5 |
| 2 | AX12a Servos (x16) | 12.0 | 9.6 ( 0.6 ea.) |
| **Total Power Consumption** | | 137.8250 W | |
| **Worst-Case Power Consumption** | | ≈200.000 W | |


Main Switches
(1x) 3-Cell LiPo Pack
(3x) 3-Cell LiPo Pack
GND
12V
GND
12V
AutoPilot v2.3
ODROID-XU
5V
Power Interface / Switching Board
3v3 Reg.
5 Volt Regulator
XBEE Wireless Radio
3v3
LIDAR
Cameras
12V (S1-4) Are individually fused / switched power buses
12V (S1)
12V (S2)
12V (S3)
12V (S4)
Leg-1 Servos AX12a (x4)
Leg-2 Servos AX12a (x4)
Leg-3 Servos AX12a (x4)
Leg-3 Servos AX12a (x4)

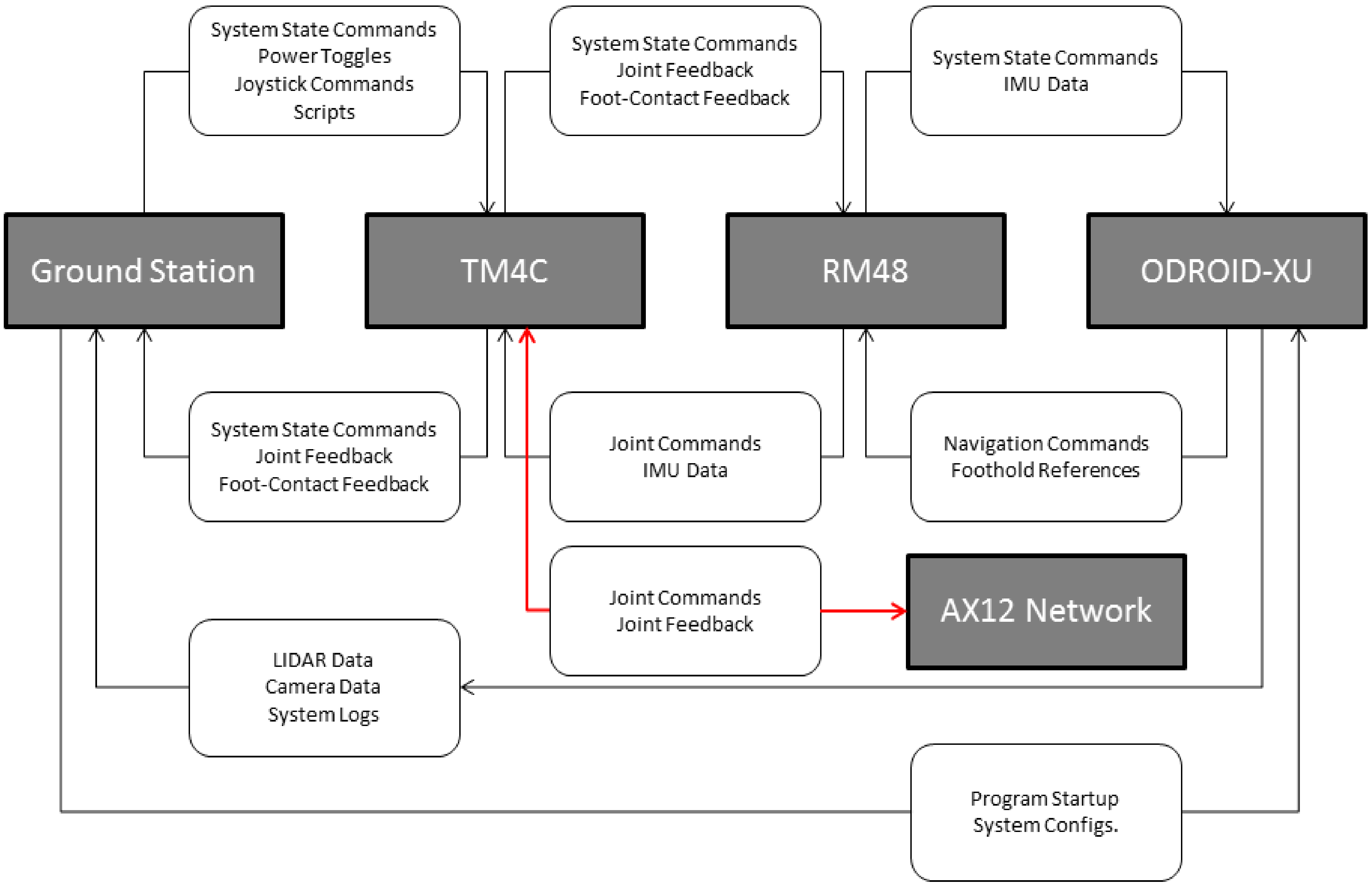
# Power Capacity and Run-time

| Net | Battery Pack | $V_{out}, V$ | Rating, $A.hr$ |
|---|---|---|---|
| 1 | 3S LiPo Pack (x1) | 12.0 | 2.0 |
| 2 | 3S LiPo Pack (x3) | 12.0 | 6.0 |
| **Average Runtime** | | ≈41 minutes | |
| **Worst-Case Runtime** | | ≈28 minutes | |

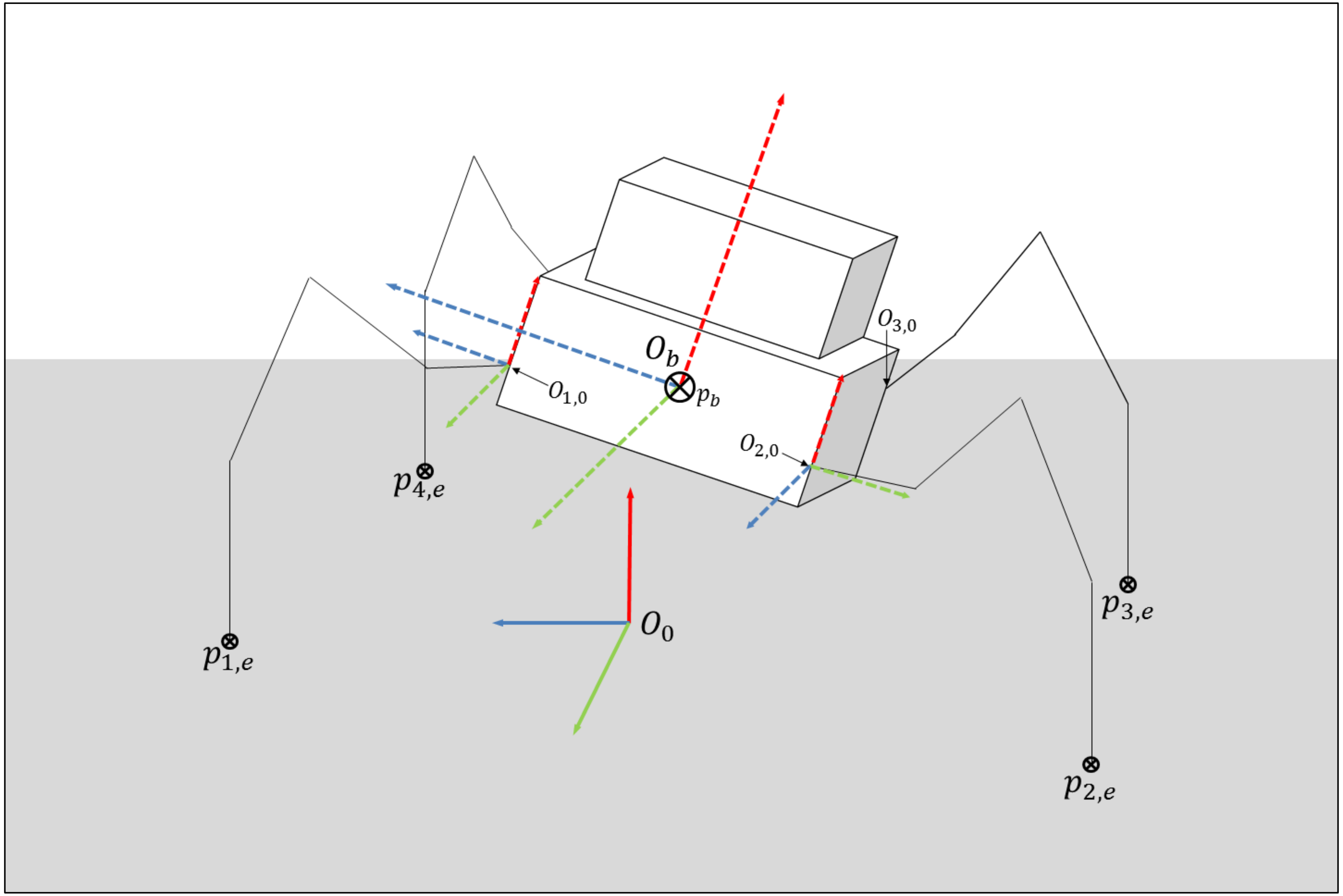
# Software: Overview



High-Level Commands
Command Processing
Command Processing
Joystick Commands
IMU Data
Feed-Forward Motion Planning
LIDAR Data
State Estimation (EKF)
Camera Data
System Monitoring
Sensor Processing
Terrain Estimation
Localization
Joint Position Feedback
Joint Feedback Preprocessing
Forward Kinematics
Joint Torque Feedback
High-Level Planning
Foothold Planning
Path Planning
Foot-Contact Feedback
Posture Controller
Central Pattern Generator
Servo Control
Inverse Kinematics
Foothold Controller
Autonomous Navigation Commands
TM4C
RM48
ODROID-XU


System State Commands
Power Toggles
Joystick Commands
Scripts
System State Commands
Joint Feedback
Foot-Contact Feedback
System State Commands
IMU Data
Ground Station
TM4C
RM48
ODROID-XU
System State Commands
Joint Feedback
Foot-Contact Feedback
Joint Commands
IMU Data
Navigation Commands
Foothold References
Joint Commands
Joint Feedback
AX12 Network
LIDAR Data
Camera Data
System Logs
Program Startup
System Configs.

# Coordinate Frame setup for Position Kinematics



# Position Kinematics: Frame Transformations

Position of the first joint of the $i^{th}$ leg in the world frame.

Trunk Orientation

$$H_0^{i,0} = \left[\begin{array}{c|c} R_{zyx}\boxed{(\theta_b)}R_z(\sigma_i) & R_{zyx}\boxed{(\theta_b)}\vec{\beta}_i + \boxed{p_b} \\ \hline 0 & 1 \end{array}\right]$$

Trunk Position

$$\begin{aligned} \sigma_i &= \tfrac{\pi}{2}(i-1) + \tfrac{\pi}{4} \\ \vec{\beta}_i &= R_z(\sigma_i)\vec{o}_\nu \\ \vec{o}_\nu &= [\nu, 0, 0]^T \end{aligned}$$

General transformation to each $j^{th}$ joint from the 1st DH-frame of each $i^{th}$ leg. Generated from Devanit-Hartenburg (DH) parameters.

$$H_{i,j}^{i,0} = \left[\begin{array}{c|c} R_{i,j}^{i,0} & p_{i,j}^{i,0} \\ \hline 0 & 1 \end{array}\right]$$

# System Modeling

# System State Vector

Considering the robot as a free-floating system with 4 legs (each 4DOF):

$$\eta = [p_b^T, \theta_b^T, q^T]^T$$

Joint angles $q \in \mathcal{R}^{16}$

Trunk Position $p_b \in \mathcal{R}^3$

Trunk Orientation $\theta_b \in \mathcal{R}^3$

Gives rise to the following state vector:

$$z = [\eta^T, \dot{\eta}^T]^T \in \mathcal{R}^{44}$$

$$z_1 = \eta \qquad z_2 = \dot{\eta}$$

# Position Kinematics: DH-Parameters

| **Link** | $a_i$ | $\alpha_i$ | $d_i$ | $\theta_i$ |
|---|---|---|---|---|
| 1 | $a_1$ | $\pi/2$ | 0 | $q^*_{i,1}$ |
| 2 | $a_2$ | 0 | 0 | $q^*_{i,2}$ |
| 3 | $a_3$ | 0 | 0 | $q^*_{i,3}$ |
| 4 | $a_4$ | 0 | 0 | $q^*_{i,4}$ |

$$H_{i,j}^{i,0} = \left[\begin{array}{c|c} R_{i,j}^{i,0} & p_{i,j}^{i,0} \\ \hline 0 & 1 \end{array}\right] \begin{cases} R_{i,j}^{i,0} = R_z(q^*_{i,j}) R_x(\alpha_i) \\ p_{i,j}^{i,0} = R_{i,j}^{i,0} \begin{bmatrix} a_i \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ d_i \end{bmatrix} \end{cases}$$

# Joint Position Limits

Joint position limits (hardware-constrained)

| Joint Var. | $q_{i,1}$, rad | $q_{i,2}$, rad | $q_{i,3}$, rad | $q_{i,4}$, rad |
|---|---|---|---|---|
| Max Range | $82^o$ | $102^o$ | $102^o$ | $102^o$ |
| Min Range | $-69^o$ | $-102^o$ | $-102^o$ | $-102^o$ |

Joint position limits (software-imposed)

| Joint Var. | $q_{i,1}$, rad | $q_{i,2}$, rad | $q_{i,3}$, rad | $q_{i,4}$, rad |
|---|---|---|---|---|
| Max Range | $45^o$ | $90^o$ | $90^o$ | $90^o$ |
| Min Range | $-45^o$ | $-90^o$ | $-90^o$ | $-90^o$ |

# Position Kinematics: Joint/Foot Positions

Position of the first joint of the $i^{th}$ leg in $O_0$.

$$p_{i,1} = E_p H_0^{i,0} e_p$$

Position of each $j^{th}$ joint of the $i^{th}$ leg in $O_0$.

$$p_{i,j} = E_p H_0^{i,0} H_{i,(j-1)}^{i,0} e_p \quad \forall \quad j \in \{2, 3, 4\}$$

Position of each foot of the $i^{th}$ leg in $O_0$.

$$p_{i,e} = E_p H_0^{i,0} H_{i,4}^{i,0} e_p$$

$$E_p = [I_{3\times3}, 0_{3\times1}] \qquad e_p = [0_{1\times3}, 1]^T$$

# Inverse Position Kinematics (IK)

Joint configuration solution for the $i^{th}$ leg, given $p_{i,e}$ and $\gamma_i$:

$$
\begin{aligned}
q_{i,1} &= \cos(i\pi)\left(\frac{\pi}{4} - \psi_i\right) \\
q_{i,2} &= \tan^{-1}\left(\frac{\zeta_{i,z}}{\sqrt{\zeta_{i,x}^2 + \zeta_{i,y}^2}}\right) \mp \cos^{-1}\left(\frac{a_3^2 - a_2^2 - \|\zeta_i\|^2}{2a_2\|\zeta_i\|}\right) \pm \pi \\
q_{i,3} &= \mp \cos^{-1}\frac{\|\zeta\|^2 - a_2^2 - a_3^2}{2a_2 a_3} \\
q_{i,4} &= \boxed{\gamma_i} - q_{i,2} - q_{i,3}
\end{aligned}
$$

Ankle configuration

# Inverse Position Kinematics, cont.

Foot position in local frame

Foot position configuration

$$\boxed{p_{i,e}^{i,0}} = E_p \left(H_{i,k}^{0}\right)^{-1} \begin{bmatrix} \boxed{p_{i,e}} \\ 1 \end{bmatrix}$$

$$\begin{aligned}
\psi_i &= \tan^{-1}\left(\frac{\left[p_{i,e}^{i,0}\right]_y}{\left[p_{i,e}^{i,0}\right]_x}\right) \\
\zeta_{i,x} &= \left[p_{i,e}^{i,0}\right]_y \sin(\psi_i) + \left[p_{i,e}^{i,0}\right]_x \cos(\psi_i) - a_4 \cos(\gamma_i) - a_1 \\
\zeta_{i,y} &= \left[p_{i,e}^{i,0}\right]_y \cos(\psi_i) - \left[p_{i,e}^{i,0}\right]_x \sin(\psi_i) \\
\zeta_{i,z} &= \left[p_{i,e}^{i,0}\right]_z - a_4 \sin(\gamma_i)
\end{aligned}$$

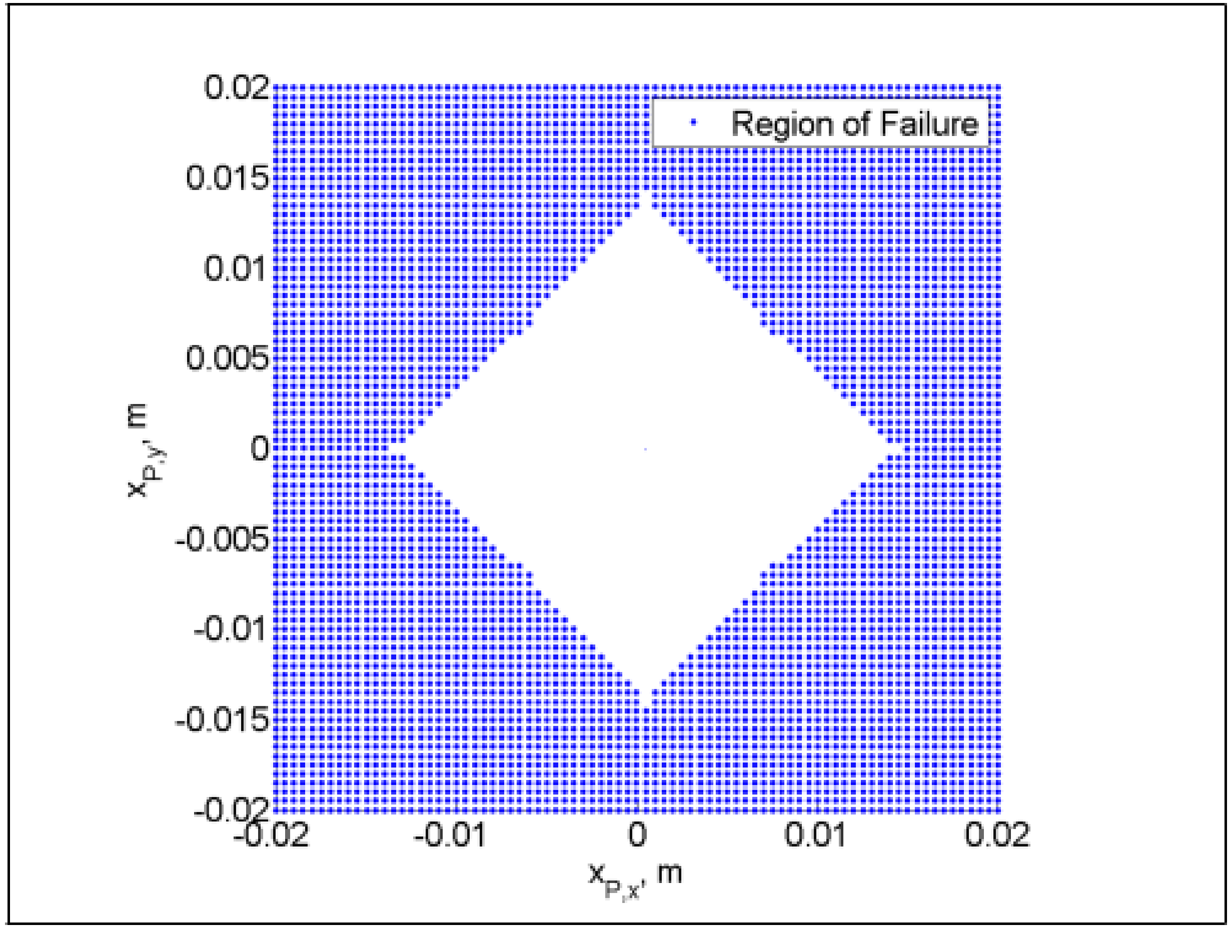
# Limits on Body Posing

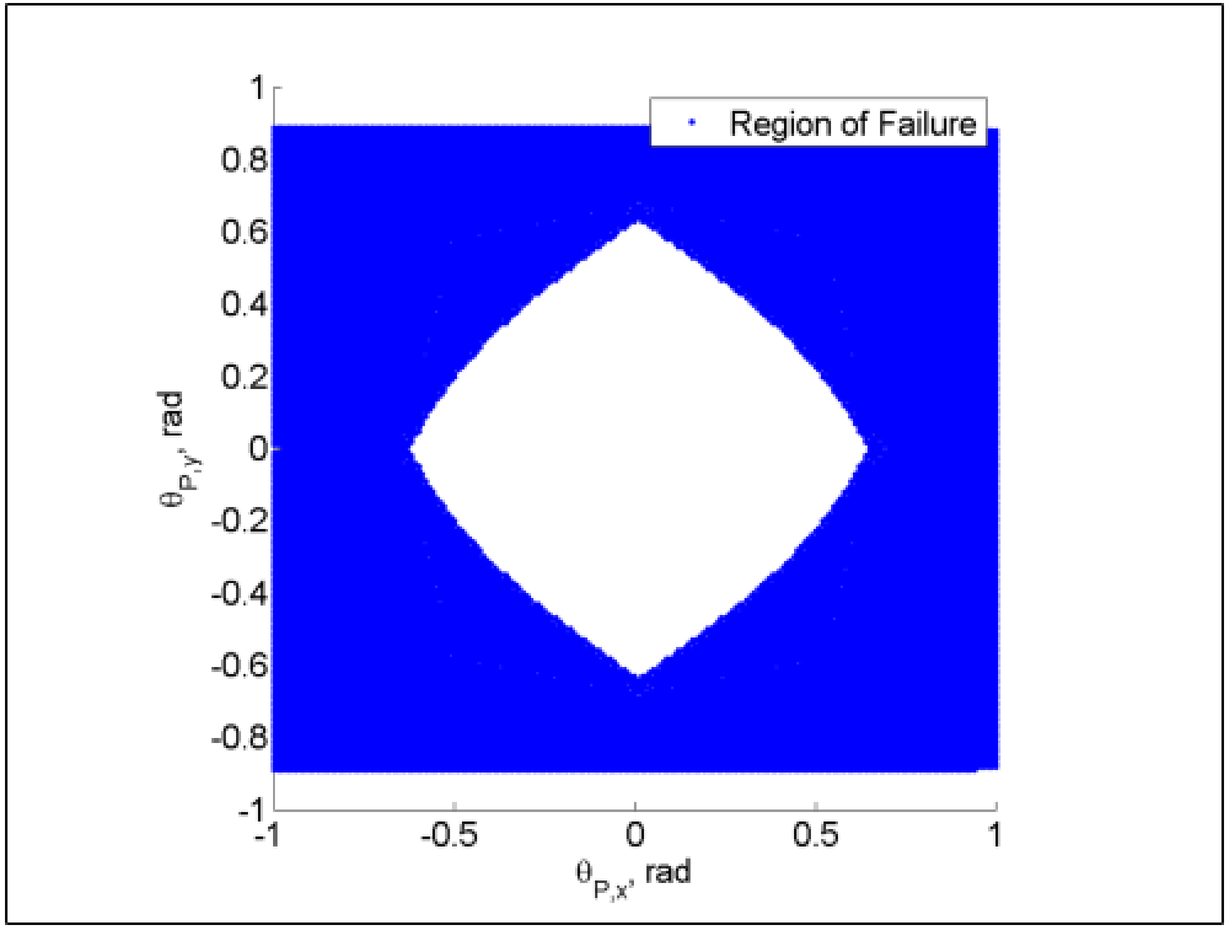
- Kinematic feasibility of each pose is checked using the inverse kinematic solution.

- Default robot stance when upright

| | $p_b^{b'}$ | $p_{1,e}^{b'}$ | $p_{2,e}^{b'}$ | $p_{3,e}^{b'}$ | $p_{4,e}^{b'}$ |
|---|---|---|---|---|---|
| x (m) | * | 0.165 | -0.165 | -0.165 | 0.165 |
| y (m) | * | 0.165 | 0.165 | -0.165 | -0.165 |
| z (m) | 0.115 | 0 | 0 | 0 | 0 |

# Limits on Body Posing, cont.




Trunk position (x,y)

Orientation (roll and pitch)

# Velocity Kinematics : Manipulator Jacobian

A mapping for foot velocity in each $i^{th}$ local leg frame defined by:

$$\dot{x}_{i,e}^{i,0} = J_{i,e}^{i,0} \dot{q}_i$$

$$\dot{x}_{i,e}^{i,0} = [(\dot{p}_{i,e}^{i,0})^T, (\dot{\theta}_{i,e}^{i,0})^T]^T \in \mathcal{R}^6$$

Manipulator Jacobian for the $i^{th}$ leg, given $p_{i,e}$ and $\gamma_i$:

$$J_{i,e}^{i,0} = \begin{bmatrix} \dot{j}_{1,1}^{i,0} & \cdots & \dot{j}_{1,4}^{i,0} \\ \vdots & \ddots & \vdots \\ \dot{j}_{6,1}^{i,0} & \cdots & \dot{j}_{6,4}^{i,0} \end{bmatrix}$$

# Velocity Kinematics : World Frame Tranformations

Given a body and foot velocities in $O_0$:

$$\Theta^{i,0} = \left(R^0_{i,0}\right)^T S\left(\dot{\theta}_{i,e} - \dot{\theta}_b\right) R^0_{i,0} = S\left(\dot{\theta}^{i,0}_{i,e}\right)$$

$$\dot{\theta}^{i,0}_{i,e} = \left[-\Theta^{i,0}_{1,2}, \Theta^{i,0}_{1,3}, -\Theta^{i,0}_{2,3}\right]^T$$

$$\dot{p}^{i,0}_{i,e} = \left(R^0_{i,0}\right)^T \left[\dot{p}_{i,e} - \dot{p}_b - S\left(\dot{\theta}_b\right)\left(p_{i,e} - p_b - R^0_{i,0}\vec{o}_\nu\right)\right]$$

# Velocity Kinematics : World Frame Tranformations

Given a body and foot velocities in $O_0$:

Angular rate of foot

Angular rate of body

$$\Theta^{i,0} = \left(R^0_{i,0}\right)^T S\left(\boxed{\dot{\theta}_{i,e}} - \boxed{\dot{\theta}_b}\right) R^0_{i,0} = S\left(\dot{\theta}^{i,0}_{i,e}\right)$$

$$\dot{\theta}^{i,0}_{i,e} = \left[-\Theta^{i,0}_{1,2}, \Theta^{i,0}_{1,3}, -\Theta^{i,0}_{2,3}\right]^T$$

$$\dot{p}^{i,0}_{i,e} = \left(R^0_{i,0}\right)^T \left[\boxed{\dot{p}_{i,e}} - \boxed{\dot{p}_b} - S\left(\dot{\theta}_b\right)\left(p_{i,e} - p_b - R^0_{i,0}\vec{o}_\nu\right)\right]$$

Foot translational velocity

Trunk translational velocity

# Inverse Velocity Kinematics

- Each leg has 4-DOF, thus $J_{i_e}^{i,0}$ is rank deficient.

$$
\begin{aligned}
\dot{q}_i &\approx \left[J_{i,e}^{i,0}\right]_{\Lambda_j}^{\dagger} \dot{x}_{i,e}^{i,0} \\
\dot{q}_i &\approx \left(\left(J_{i,e}^{i,0}\right)^T J_{i,e}^{i,0} + \Lambda_J\right)^{-1} \left(J_{i,e}^{i,0}\right)^T \dot{x}_{i,e}^{i,0}
\end{aligned}
$$

- Least-squares approximate solution for $\dot{q}_i$ by a $\Lambda_J$-weighted Moore-Penrose pseudo inverse.
- Weighting factor used to handle manipulator singularities.

# Dynamical Model

Dynamics of the system takes the following form:

$$M(\eta)\ddot{\eta} + C(\eta, \dot{\eta})\dot{\eta} + G(\eta) + \Delta H = \tau + J^T(\eta) f_{ext}$$

with external contact-forces:

$$f_{ext} = [f_{1,ext}^T, f_{2,ext}^T, f_{3,ext}^T, f_{4,ext}^T]^T \in \mathcal{R}^{24}$$

$$f_{i,ext} \in \mathcal{R}^6$$

and joint input torque dynamics:

$$\tau = [0_{1x6}, \tau_q^T]^T$$

$$\tau_q = k_s(q^r - q)$$

# Dynamical Model

Dynamics of the system takes the following form:

$$M(\eta)\ddot{\eta} + C(\eta, \dot{\eta})\dot{\eta} + G(\eta) + \boxed{\Delta H} = \tau + J^T(\eta) f_{ext}$$

Lump disturbance term

with external contact-forces:

$$f_{ext} = [f_{1,ext}^T, f_{2,ext}^T, f_{3,ext}^T, f_{4,ext}^T]^T \in \mathcal{R}^{24}$$

$$f_{i,ext} \in \mathcal{R}^6$$

and joint input torque dynamics:

$$\tau = [0_{1x6}, \tau_q^T]^T$$

$$\tau_q = k_s(q^r - q)$$

# Dynamical Model : State Space Representation

$$\dot{z}_1 = z_2$$

$$\dot{z}_2 = M^{-1}(z_1)(\tau + \Phi(z_1, z_2, f_{ext}))$$

$$\Phi(z_1, z_2, f_{ext}) = J^T(z_1) f_{ext} - C(z_1, z_2) z_2 - G(z_1) - \Delta H$$

First-order discretization with a sampling period $\Delta_s$:

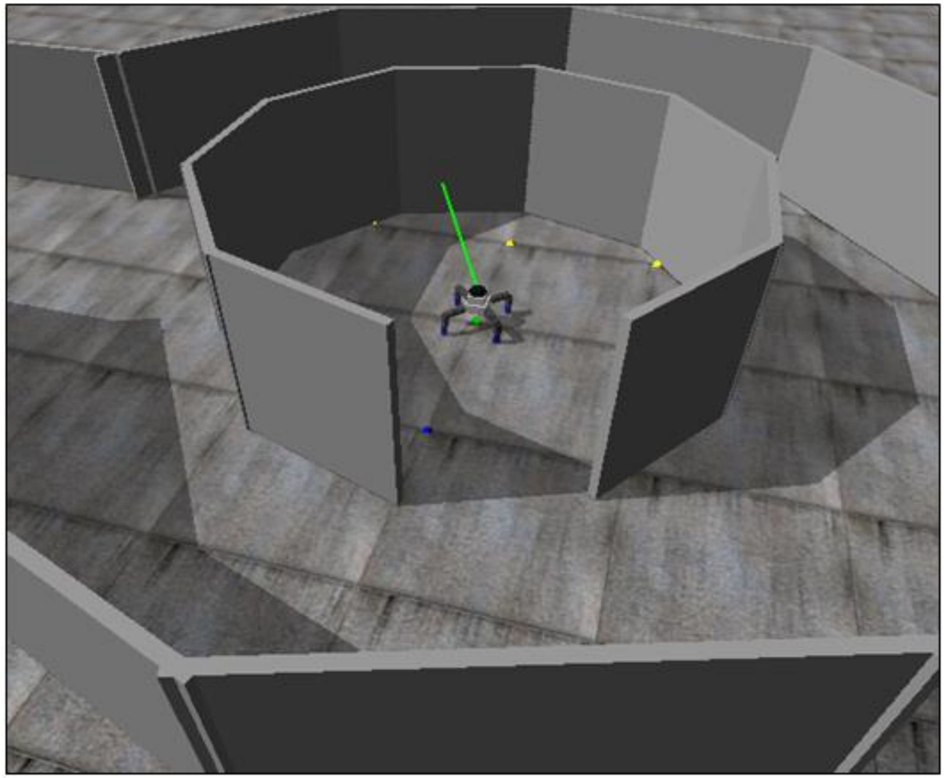
$$z_{1,k+1} = z_{1,k} + (e_{1,k}^{\Delta_s} + z_{2,k})\Delta_s$$

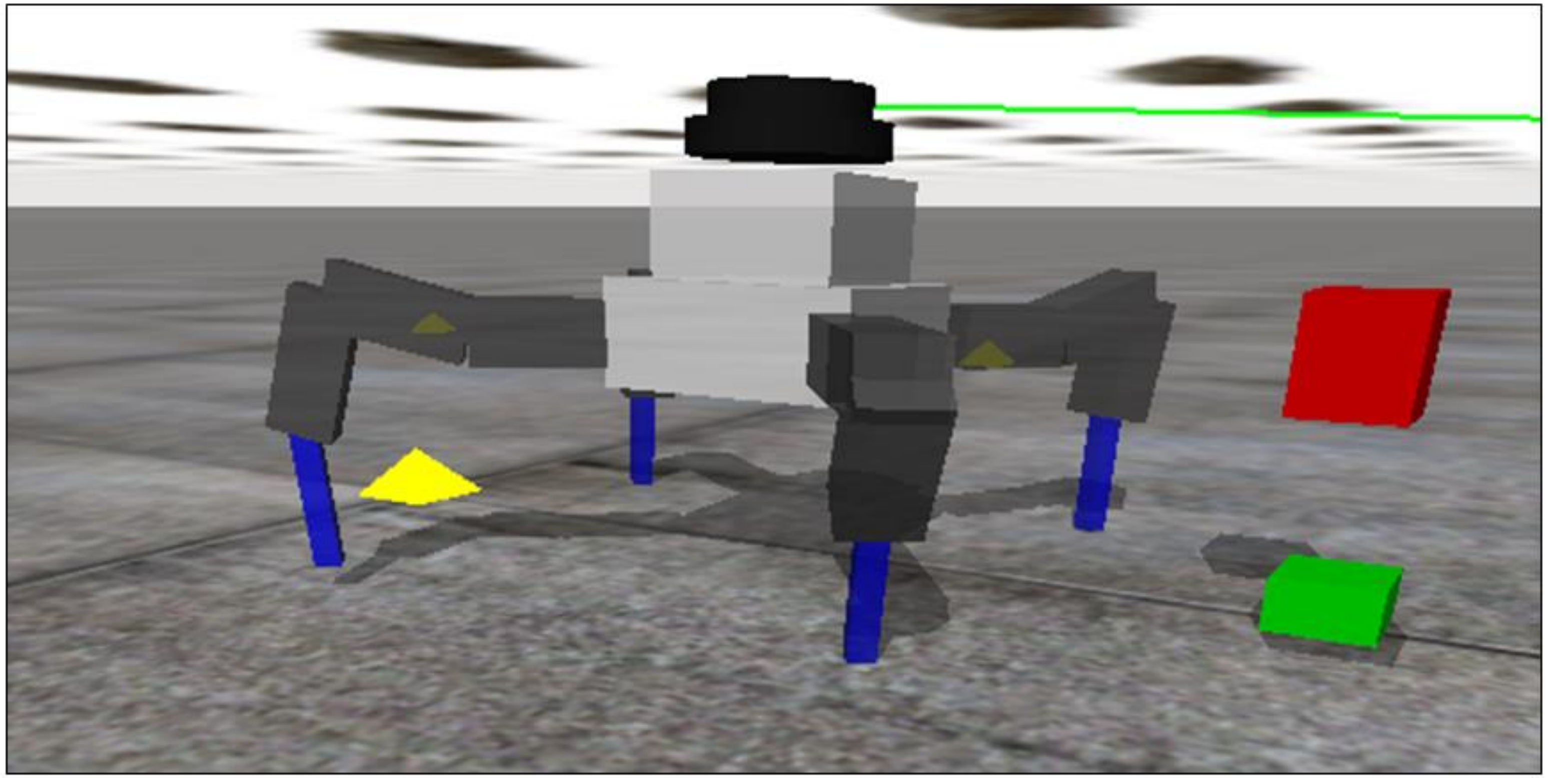
$$z_{2,k+1} = z_{2,k} + M_{1,k}^{-1}(e_{2,k}^{\Delta_s} + \tau_k + \Phi_k)\Delta_s$$

$$t = \Delta_s k$$

$$M_{1,k} = M(z_{1,k})$$

# BlueFoot Simulator





# BlueFoot Simulator, cont.

- Full quadruped dynamic requires an analysis of many dynamical modes for each foot contact configuration.

- Actual contact dynamics (friction, multiple collisions) can be complex

- A numerical dynamical simulator helps reduce complexity

- Configurable environments

# BlueFoot Simulator, cont.

- Implemented using the *Open Dynamics Engine* C++ library
  - Generates dynamical updates for all bodies via constrained quadratic minimization
  - Rigid body and contact dynamics formulated as a Linear Complimentary problem (LCP)
  - First-order integration with small time-step ($\Delta t \ll 1$)

# Gaiting and Stabilization

# Central Pattern Generator (CPG)-based Gaiting

# CPG with Hopf Unit Oscillators

Each $i^{th}$ Hopf oscillator has a state-pair $\{y_{1,i}, y_{2,i}\}$

which make up the CPG network state vectors $y_1, y_2 \in R^4$ by:

$$\begin{aligned} y_1 &= [y_{1,1}, y_{1,2}, y_{1,3}, y_{1,4}]^T \\ y_2 &= [y_{2,1}, y_{2,2}, y_{2,3}, y_{2,4}]^T \end{aligned}$$

# CPG with Hopf Unit Oscillators, cont.

The full Hopf-oscillator network dynamics are then

$$\begin{aligned}
\dot{y}_1 &= A_1 \left( \Psi_M M\left(y_1, y_2\right) - \Gamma \right) y_1 + \Psi_\omega W y_2 \\
\dot{y}_2 &= A_2 \left( \Psi_M M\left(y_1, y_2\right) - \Gamma \right) y_2 - \Psi_\omega W y_1 + K y_2
\end{aligned}$$

with $$M\left(y_1, y_2\right) = \begin{bmatrix} y_{1,1}^2 + y_{2,1}^2 & 0 & 0 & 0 \\ 0 & y_{1,2}^2 + y_{2,2}^2 & 0 & 0 \\ 0 & 0 & y_{1,3}^2 + y_{2,3}^2 & 0 \\ 0 & 0 & 0 & y_{1,4}^2 + y_{2,4}^2 \end{bmatrix}$$

# CPG with Hopf Unit Oscillators, cont.

The full Hopf-oscillator network dynamics are then

Amplitude Frequency

$$
\begin{aligned}
\dot{y}_1 &= A_1\left(\Psi_M M\left(y_1, y_2\right) - \boxed{\Gamma}\right) y_1 + \Psi_\omega \boxed{W} y_2 \\
\dot{y}_2 &= A_2\left(\boxed{\Psi_M} M\left(y_1, y_2\right) - \Gamma\right) y_2 - \boxed{\Psi_\omega} W y_1 + \boxed{K} y_2
\end{aligned}
$$

Amplitude Modulation Matrix

Frequency Modulation Matrix

Coupling Matrix

with $$M\left(y_1, y_2\right) = \begin{bmatrix} y_{1,1}^2 + y_{2,1}^2 & 0 & 0 & 0 \\ 0 & y_{1,2}^2 + y_{2,2}^2 & 0 & 0 \\ 0 & 0 & y_{1,3}^2 + y_{2,3}^2 & 0 \\ 0 & 0 & 0 & y_{1,4}^2 + y_{2,4}^2 \end{bmatrix}$$

# CPG “Reflex” Design

Reflexes designed using modulation parameters:

$$\begin{aligned} \Psi_\omega &= I + A_\omega \mathrm{diag}\left(\psi_i\right) \\ \Psi_M &= I - A_\mu \mathrm{diag}\left(\psi_i\right) \end{aligned}$$

$i^{th}$ leg reflexes using inertial feedback:

$$\begin{aligned} \psi_i &= \mathrm{sig}\left(w_i \upsilon_i - w_i c_i\right) \mu_i \\ \upsilon_i &= \|\dot{\epsilon}_\theta\| \left(1 + \frac{\Delta x_i}{\|\Delta x_i\|}^T \frac{\dot{\epsilon}_\theta}{\|\dot{\epsilon}_\theta\|}\right) \\ \Delta x_i &= p_{i,e} - p_b \end{aligned}$$

$$\dot{\epsilon}_\theta = R_{z_b}\left(\frac{\pi}{2}\right)\left(\dot{\hat{\theta}}_b - \dot{\theta}_b^r\right)$$

# CPG "Reflex" Design

Reflexes designed using modulation parameters:

$$\Psi_\omega = I + A_\omega \mathrm{diag}\left(\psi_i\right)$$

$$\Psi_M = I - A_\mu \mathrm{diag}\left(\psi_i\right)$$

$i^{th}$ leg reflexes using inertial feedback:

$$\psi_i = \boxed{\mathrm{sig}\left(w_i v_i - w_i c_i\right) \mu_i} \text{ — Sigmoid Activation}$$

$$v_i = \|\dot{\epsilon}_\theta\| \left(1 + \frac{\Delta x_i}{\|\Delta x_i\|}^T \frac{\dot{\epsilon}_\theta}{\|\dot{\epsilon}_\theta\|}\right)$$

$$\Delta x_i = p_{i,e} - p_b$$

$$\dot{\epsilon}_\theta = R_{z_b}\left(\frac{\pi}{2}\right)\left(\dot{\hat{\theta}}_b - \dot{\theta}_b^r\right)$$

# Oscillator Coupling Design

Coupling matrix is tuned to produce different phase offsets between oscillators:

$$K \equiv \begin{bmatrix} 0 & -1 & 1 & -0.5 \\ -1 & 0 & -0.5 & 1 \\ -1 & -0.5 & 0 & -1 \\ -0.5 & 1 & -1 & 0 \end{bmatrix}$$

Produces 4-Pace Walking Gait

$$K \equiv \begin{bmatrix} 0 & -1 & 1 & -0.5 \\ -1 & 0 & -0.5 & 1 \\ -1 & 0.5 & 0 & -1 \\ 0.5 & -1 & -1 & 0 \end{bmatrix}$$

Produces 2-Pace Trotting Gait

# Reflexive CPG Design, cont.

$$W = \omega_s \alpha \begin{bmatrix} \frac{1}{1+e^{\zeta y_{2,1}}} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \frac{1}{1+e^{\zeta y_{2,4}}} \end{bmatrix} + \omega_s (1-\alpha) \begin{bmatrix} \frac{1}{1+e^{-\zeta y_{2,1}}} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \frac{1}{1+e^{-\zeta y_{2,4}}} \end{bmatrix}$$

$$\omega_s = a_v v^r + a_\omega \left\| \omega^r \right\|$$

# Foothold Controller

$$
\begin{aligned}
\dot{\tilde{\theta}}_c &= \omega^r \\
\dot{\tilde{p}}_c &= v^r \vec{u}^r \\
\dot{\tilde{p}}_{i,e} &= v^r \vec{u}^r + S\left(\omega^r \vec{h}_b\right) \tilde{R}_b \left(\tilde{p}_{i,e} - \tilde{p}_c\right)
\end{aligned}
$$

$$\dot{\tilde{p}}^v_{i,e} = \frac{1}{m_i} \int^t P_{\vec{h}_i} \left(F_{s,i} - b_c \dot{p}_{i,e} + F_{\epsilon,i}\right) d\tau$$

$$
\begin{aligned}
F_{s,i} &= k_c \mu_i \left(\tilde{p}_{i,e} - p_{i,e}\right) U\left(y_{2,i}\right) \\
F_{\epsilon,i} &= a_\epsilon \dot{\epsilon}_\theta U\left(y_{2,i}\right)
\end{aligned}
$$

$$\dot{\tilde{p}}^y_{i,e} = \left(\alpha_v v^r g + \alpha_\omega \left\|\omega^r\right\|\right) \dot{y}_{2,i} \vec{h}_i$$

$$\dot{p}^r_{i,e} = \dot{\tilde{p}}^v_{i,e} + \dot{\tilde{p}}^y_{i,e}, \quad i = \{1, ..., 4\}$$

# ZMP Body-Placement Controller

$$\tau_{net} = \left(\bar{p}_b^{b'} - \hat{p}_{COG}^{b'}\right) \times (\vec{g} m_b) + \tau_{legs}$$

$$\begin{aligned}
\tau_{legs} &= \sum_{i=1}^{4} (1-\mu_i) \left( m_{i,e} p_{i,e}^{b'} + \sum_{i=1}^{4} \left( m_{i,j}^J d_{i,j}^J + m_{i,j}^L d_{i,j}^L \right) \right) \times \vec{g} \\
d_{i,j}^J &= p_{i,j}^{b'} - \hat{p}_{COG}^{b'}. \\
d_{i,j}^L &= \begin{cases} 0.5 \left( p_{i,j+1}^{b'} - p_{i,j}^{b'} \right) + p_{i,j}^{b'} - \hat{p}_{COG}^{b'} & \text{if } j < 4 \\ 0.5 \left( p_{i,e}^{b'} - p_{i,j}^{b'} \right) + p_{i,j}^{b'} - \hat{p}_{COG}^{b'} & \text{if } j = 4 \end{cases}
\end{aligned}$$

# ZMP Body-Placement Controller, cont.

$$
\begin{aligned}
\hat{p}_{COG}^{b'} &= \frac{1}{m_T}\left(m_b p_b^{b'} + \sum_{i=1}^{4}\left(m_{i,e} p_{i,e}^{b'} + \sum_{j=1}^{4}\left(m_{i,j}^{J} p_{i,j}^{b'} + m_{i,j}^{L} p_{i,j}^{b',L}\right)\right)\right) \\
p_{i,j}^{L} &= \begin{cases} 0.5\left(p_{i,j+1}^{b'} - p_{i,j}^{b'}\right) + p_{i,j}^{b'} & \text{if } j < 4 \\ 0.5\left(p_{i,e}^{b'} - p_{i,j}^{b'}\right) + p_{i,j}^{b'} & \text{if } j = 4 \end{cases}
\end{aligned}
$$

$$
m_T = m_b + \sum_{i=1}^{4}\left(m_{i,e} + \sum_{j=1}^{4}\left(m_{i,j}^{J} + m_{i,j}^{L}\right)\right)
$$

# ZMP Body-Placement Controller, cont.

$$p^{b'}_{ZMP} = R_{z_P}\left(\frac{\pi}{2}\right)\left(\frac{\|g\|}{m_b}\right)\tau_{legs} + \hat{p}^{b'}_{COG}$$

$$\dot{p}^{b',r}_b = P_{\vec{h}_i}\left(K_Z(p^{b'}_{ZMP} - p^{b'}_b) + K_F\frac{F_r}{m_b}\right)$$

$$\begin{aligned} F_r &= \sum_{i=1}^{4} e^{\left(k_l r_i^-\right)} + e^{\left(k_l r_i^+\right)} \\ r_i^+ &= \left\|p^{b'}_{i,e} - p^{b'}_b\right\| - r_{max} \\ r_i^- &= r_{min} - \left\|p^{b'}_{i,e} - p^{b'}_b\right\| \end{aligned}$$

# NARX-NN Trunk Leveling Controller

$$\begin{aligned}
\hat{\psi}_{k+1} &= \mathcal{N}(\hat{\Psi}_k^N, U_k^N) \\
\hat{\Psi}_k^N &= [\hat{\psi}_k, \hat{\psi}_{k-1}, ..., \hat{\psi}_{k-N+1}] \\
U_k^N &= [u_k, u_{k-1}, ..., u_{k-N+1}]
\end{aligned}$$

$$u_k = (z_{1,k}, z_{2,k}, f_{ext,k})$$

# NARX-NN Training Regimen

$$\psi_{k+1} = \tau_k - \hat{M}_{1,k}(z_{2,k+1} - z_{2,k})\Delta_s^{-1} = \Phi_k - e_{2,k}^{\Delta_s}$$

$$\hat{M}_{1,k} = \hat{M}_{nom} \forall k$$

$$\hat{M}_{nom} = \begin{bmatrix} \hat{M}_{bb} & \hat{M}_{bq} \\ \hat{M}_{qb} & \hat{M}_{qq} \end{bmatrix}$$

$$\psi_k \xrightarrow{BP(\gamma^{lr})} \mathcal{N}(\Psi_{k-1}^N, U_{k-1}^N)$$

# NARX-NN Training Regimen, cont.

$$\gamma^{lr} \leftarrow \begin{cases} \gamma^{lr}(1-\beta) & \text{if } MSE_k > MSE_{k-1} \\ \gamma^{lr}(1+\zeta\beta), & \text{otherwise.} \end{cases}$$

$$\begin{aligned} e_{\mathcal{N},k} &= \hat{\psi}_k - \psi_k \\ MSE_k &\leftarrow \lambda \|e_{\mathcal{N},k}\|_2^2 + MSE_{k-1}(1-\lambda) \end{aligned}$$

# NARX-NN Control Law

$$\ddot{\theta}_b = \Gamma_1 M^{-1}(z_1)(\Gamma_2 \tau_q + \Phi)$$

$$\begin{aligned}\Gamma_1 &= [0_{3\times 3}, I_{3\times 3}, 0_{3\times 16}] \\ \Gamma_2 &= [0_{16\times 6}, I_{16\times 16}]^T\end{aligned}$$

$$\ddot{\theta}_b = -K_b\theta_b - K_d\dot{\theta}_b$$

$$\tau_q \approx -\left[\Gamma_1 M^{-1}(z_1)\Gamma_2\right]^\dagger \Gamma_1 M^{-1}(z_1)(\Phi + K_b\theta_b + K_d\dot{\theta}_b)$$

$$\hat{\tau}_{q,k} = -\left[\Gamma_1 \hat{M}_{1,k}^{-1}\Gamma_2\right]^\dagger \Gamma_1 \hat{M}_{1,k}^{-1}(\hat{\psi}_{k+1} + K_b\theta_{b,k} + K_d\dot{\theta}_{b,k})$$

# NARX-NN Control Law

$$q_{1,k}^{r,*} = k_s^{-1}\left(\hat{\tau}_{q,k}\right) + q_{1,k}$$

$$\tilde{q}_k^r \leftarrow (1-\alpha)q_k^r + \alpha(q_k^{r,*})$$

# Potential Fields

$$
\begin{aligned}
\vec{u}_L^r &= \sum_{x_i^L \in S} g_\psi(x_i^L) f(x_i^L) \frac{x_i^L}{\|x_i^L\|} \\
P_L &= \alpha_p \sum_{x_i^L \in S^L} g_\psi(x_i^L) f(x_i^L) U\left(f(x_i^L)\right)
\end{aligned}
$$

$$
\begin{aligned}
\Delta d &\equiv \|x\| - d_{min} \\
f(x) &= \begin{cases} -\lambda_{c,1}\left(\Delta d\right)^2 & \text{if } \Delta d < 0 \\ (\Delta d)\left(1 - e^{-\lambda_{c,2}(\Delta d)^2}\right) & \text{otherwise} \end{cases}
\end{aligned}
$$

# Potential Fields

$$\operatorname{ang}(x) = \tan^{-1}\left(\frac{[x]_x}{[x]_y}\right)$$

$$g_\psi(x) = \begin{cases} \left(1 - \left\| \frac{\operatorname{ang}(x)}{\psi} \right\|\right)^{\alpha_g} & \operatorname{ang}(x) < \psi \\ 0 & \text{otherwise.} \end{cases}$$

$$\theta_L^r = \tan^{-1}\left(\frac{[\vec{u}_L^r]_x}{[\vec{u}_L^r]_y}\right)$$

$$\dot{v}_L^r = \beta_v\left(P_L + v_{L,min}^r - v_L^r\right)$$

$$\dot{\omega}_L^r = \beta_\omega\left(\left(\frac{\omega_L^{r,max}}{\pi}\right)(\theta_L^r - \theta_{b,z}) - \omega_L^r\right)$$

# Potential Fields

$$
\begin{aligned}
v_C^r &= v_C^{r,max}\left(1 - e^{-c_r(r-r_{min})^2}\right)\\
\omega_C^r &= \omega_C^{r,max}\left(\frac{w_{Im} - 2u}{w_{Im}}\right)\\
\theta_{b,x}^r &= \theta_{b,x}^{r,max}\left(\frac{2v - h_{Im}}{h_{Im}}\right)
\end{aligned}
$$

$$
v(r) \equiv \begin{cases} e^{-c_{mix}(r-r_{mix})^2} & \text{if } r < r_{mix} \\ 1 & \text{otherwise} \end{cases}
$$

$$
\begin{bmatrix} v^r \\ \omega^r \end{bmatrix} = v(r)\begin{bmatrix} v_C^r \\ \omega_C^r \end{bmatrix} + (1 - v(r))\begin{bmatrix} v_L^r \\ \omega_L^r \end{bmatrix}
$$

# Potential Fields

$$\theta^*_{b,x,k} = (\theta_{s,max} - \theta_{s,min}) \left(\frac{k}{K}\right) + \theta_{s,min}$$

$$\dot{\theta}^r_{b,x} = \alpha_\theta(\theta^*_{b,x,k} - \hat{\theta}_{b,x})$$

$$\forall k \in \{1, ..., K\}$$

$$\tilde{f}_{scan} \approx \frac{1}{K}\frac{\Delta\theta_L}{\omega_L}$$

# Potential Fields

$$H_0^L = H_b^L H_0^b$$

$$\begin{bmatrix} \bar{x}_{i,k} \\ 1 \end{bmatrix} = H_b^L H_0^b \begin{bmatrix} x_i \\ 0 \\ 1 \end{bmatrix} \forall x_i \in S$$

$$\bar{S} = \bigcup_{k=1}^{N_s} \bar{S}_k$$

# Potential Fields

$$\begin{bmatrix} \bar{p} \\ m_{i,j} \end{bmatrix} = \begin{bmatrix} i \\ j \\ \hline m_{i,j} \end{bmatrix} = \left[ \begin{bmatrix} n/d & 0 & 0 \\ 0 & m/w & 0 \\ 0 & 0 & 1 \end{bmatrix} \left( \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} (\bar{x} - \bar{p}_0^{ROI}) + \begin{bmatrix} d \\ 0 \\ 0 \end{bmatrix} \right) \right]$$

# Potential Fields

$$\left\{\mathcal{C} = f(\mathcal{M}, \Gamma) \ : \ \mathcal{R}^{n \times m} \rightarrow \mathcal{R}^{n \times m \times l}\right\}$$

$$\Gamma \equiv \left\{\bar{p}^{\mathcal{M}}_{1,e}, \bar{p}^{\mathcal{M}}_{2,e}, \bar{p}^{\mathcal{M}}_{3,e}, \bar{p}^{\mathcal{M}}_{4,e}\right\}$$

$$\bar{p}^{\mathcal{M}}_{v,e} \ = \ [i_v, j_v]^T \ \in \ \mathbb{Z}^{\mathcal{M}}$$

$$i_v \ \in \ \{1, ..., m\}$$

$$j_v \ \in \ \{1, ..., n\}$$

# Potential Fields

$$\Gamma'(\bar{p}_b^{\mathcal{M}}, u) = \left\lceil R_z^{\mathcal{M}}(\gamma(u))\Gamma \right\rceil + \bar{p}_b^{\mathcal{M}} B$$

$$\bar{p}_b^{\mathcal{M}} = [i, j]^T \in \mathbb{Z}^{\mathcal{M}}$$

$$R_z^{\mathcal{M}}(\gamma(u)) \in \mathcal{R}^{2\times 2}$$

$$B = [1, 1, 1, 1]$$

$$\gamma(u) = 2\pi \left(\frac{u}{l}\right) - \pi \quad \in [-\pi, \pi]$$

# Potential Fields

$$\Gamma' \equiv \Gamma'(\bar{p}_b^{\mathcal{M}}, u)$$

$$\mathcal{H}_v = \mathcal{M}(\mathrm{col}(\Gamma')_v)$$

$$\delta\mathcal{H}_v = \nabla\mathcal{M}(\mathrm{col}(\Gamma')_v)$$

$$\mathcal{C}(\bar{p}_b^{\mathcal{M}}, u) = k_{\mathrm{var}}\mathrm{var}(\mathcal{H}) + k_\delta \sum_{v=1}^{4} \delta\mathcal{H}_v^2 + k_\theta \gamma^2(u)$$

$$J(N) = \sum_{k=0}^{N} \mathcal{C}\left(\bar{p}_{b,k}^{\mathcal{M}}, u_k\right)$$

# Potential Fields

$$\min_{\psi_k, \lambda_k} \mathcal{C}\left(\psi_k, \lambda_k\right)$$

$$\text{s.t.}\ \ \psi_k \in \mathbb{Z}_k^w$$

$$\lambda_k \in \mathbb{Z}_k^u$$

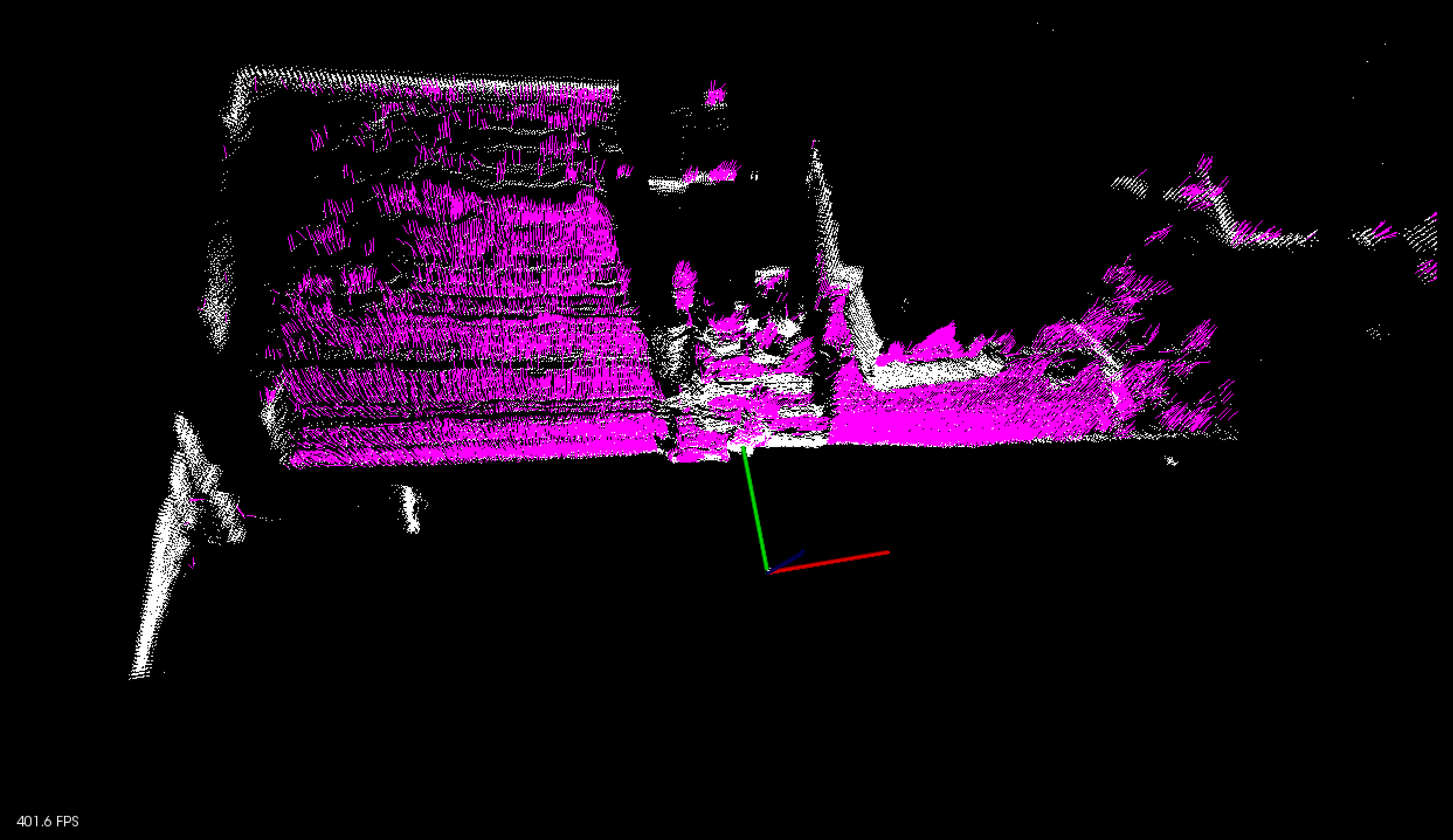
$$\mathbb{Z}_k^w \ \text{s.t.}$$

$$i_{\psi,k} \in \left\{\left(i_{b,k-1} - d_w\right), \left(i_{b,k-1} - d_w + 1\right), \ldots, \left(i_{b,k-1} + d_w\right)\right\}$$

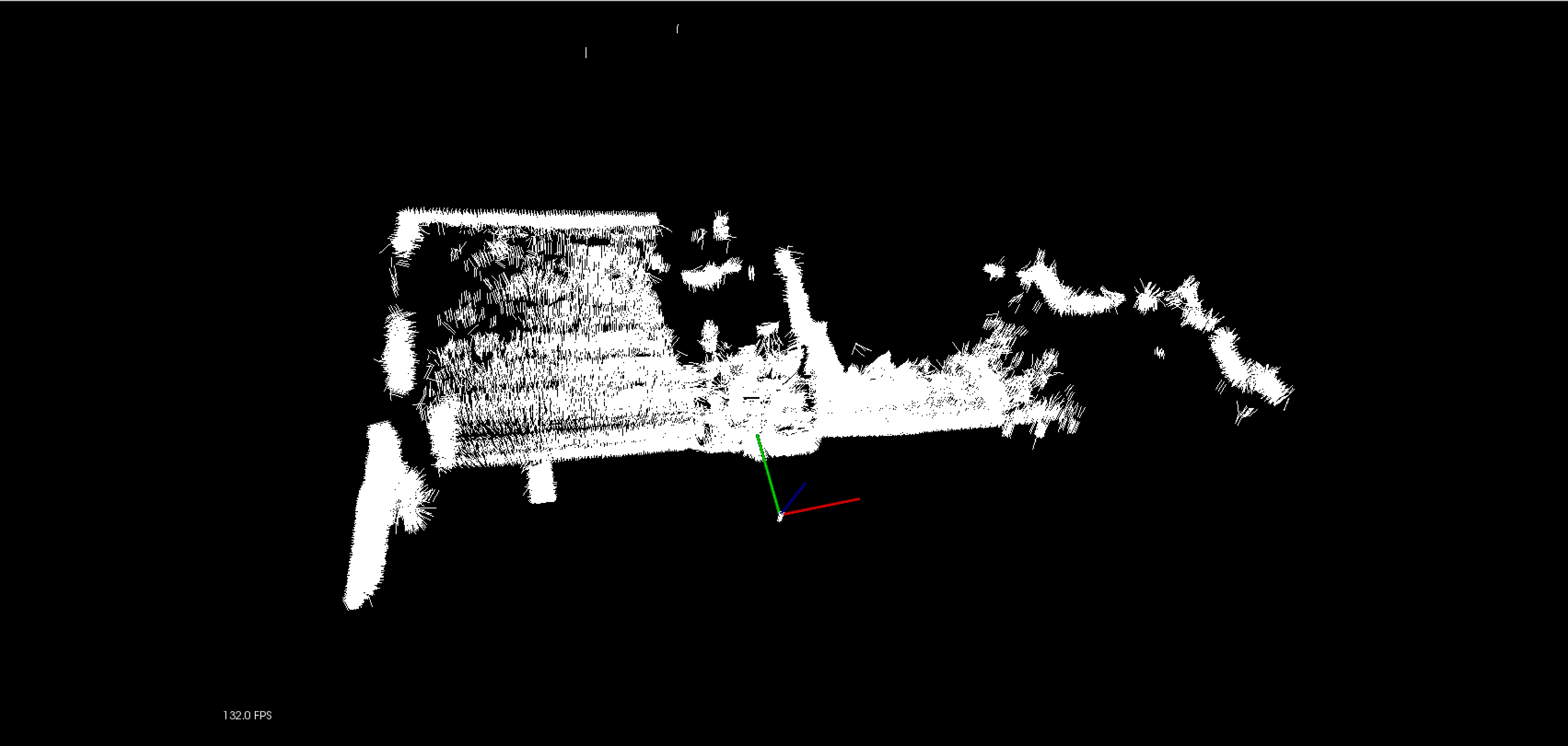
$$j_{\psi,k} \in \left\{\left(j_{b,k-1} + 1\right), \ldots, \left(j_{b,k-1} + d_w\right)\right\}$$

$$\mathbb{Z}_k^u \equiv \left\{\left(u_{k-1} - d_u\right), \left(u_{k-1} - d_u + 1\right), \ldots, \left(u_{k-1} + d_u\right)\right\}$$

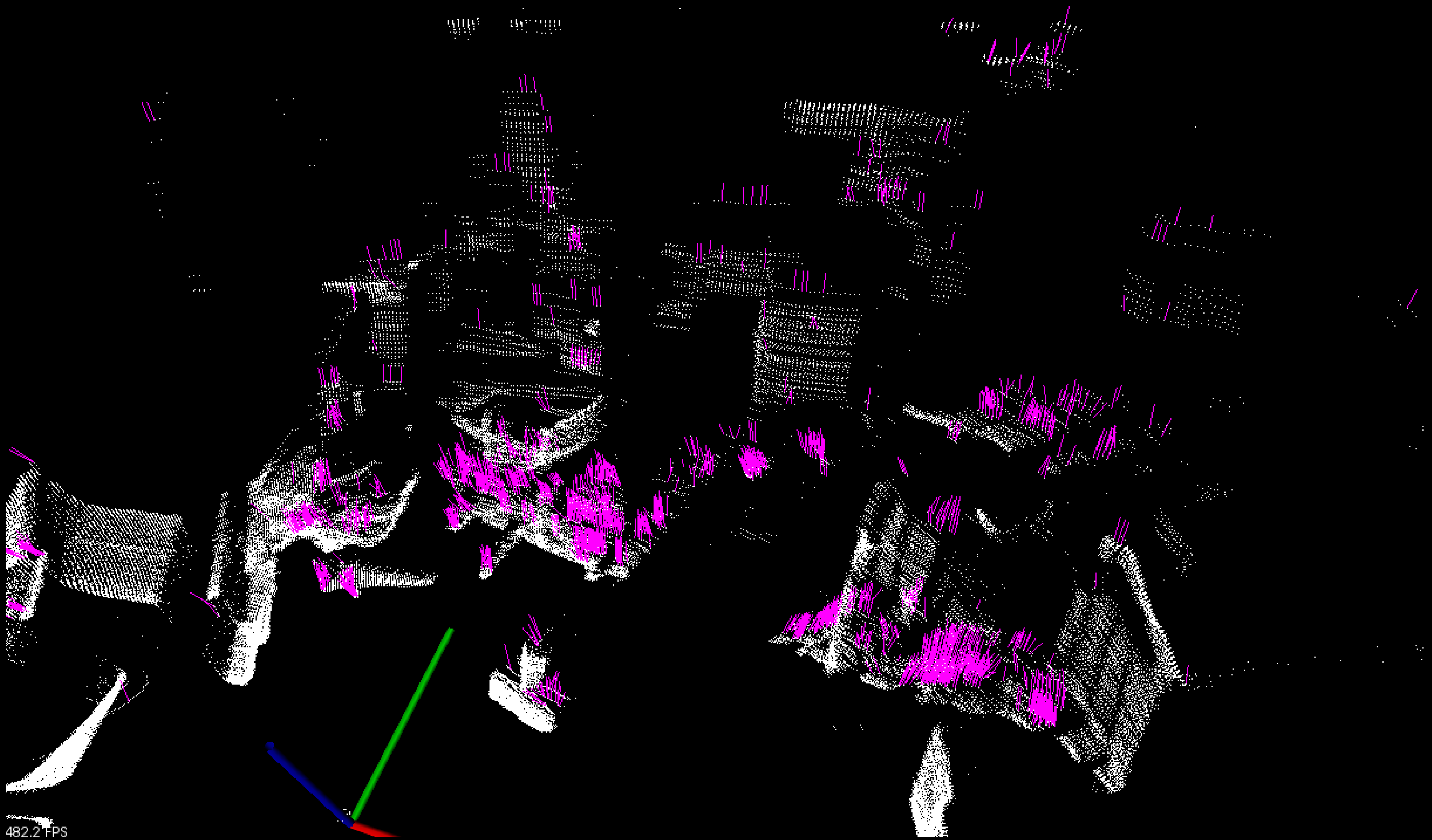
# Source Point Cloud (1)



# Surface Normal Estimation (2)



# Source Point Cloud (2)



# Surface Normal Estimation (2)