Using MATLAB/Simulink on Mars*

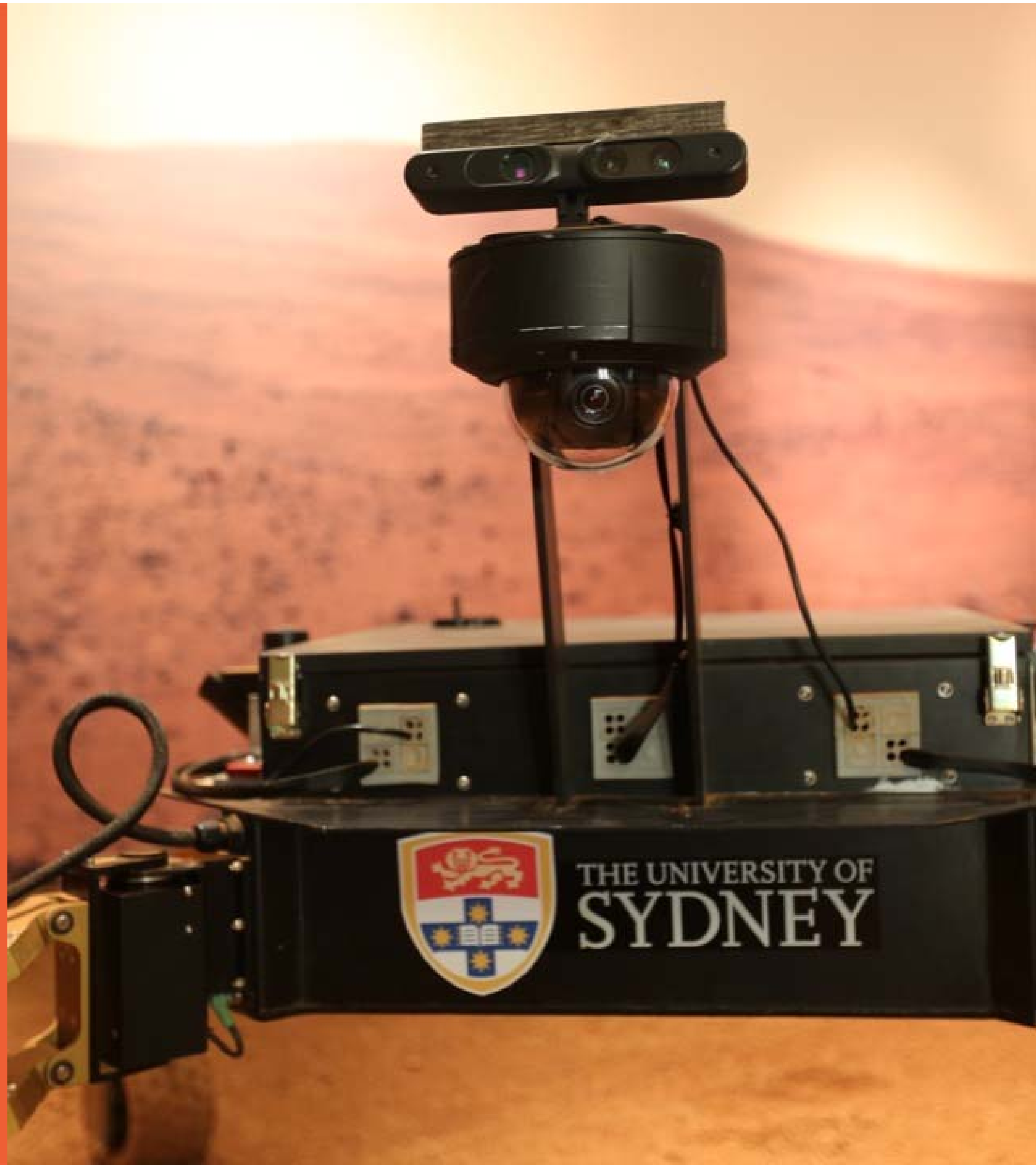
***+/- 401 million km**

William Reid

May 24, 2016
MATLAB Conference Sydney
The Museum of Applied Arts and Sciences, Sydney

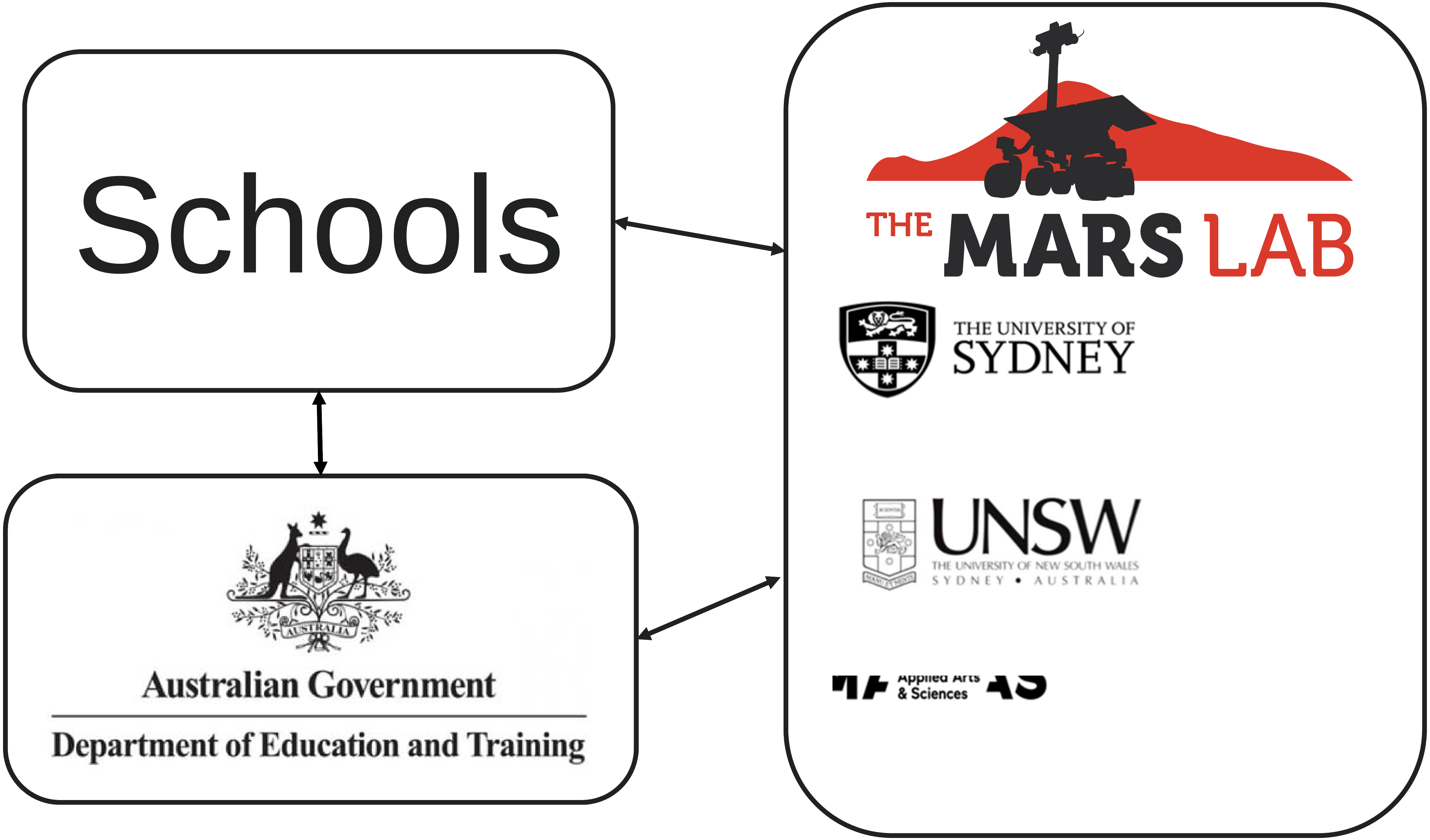
THE UNIVERSITY OF SYDNEY

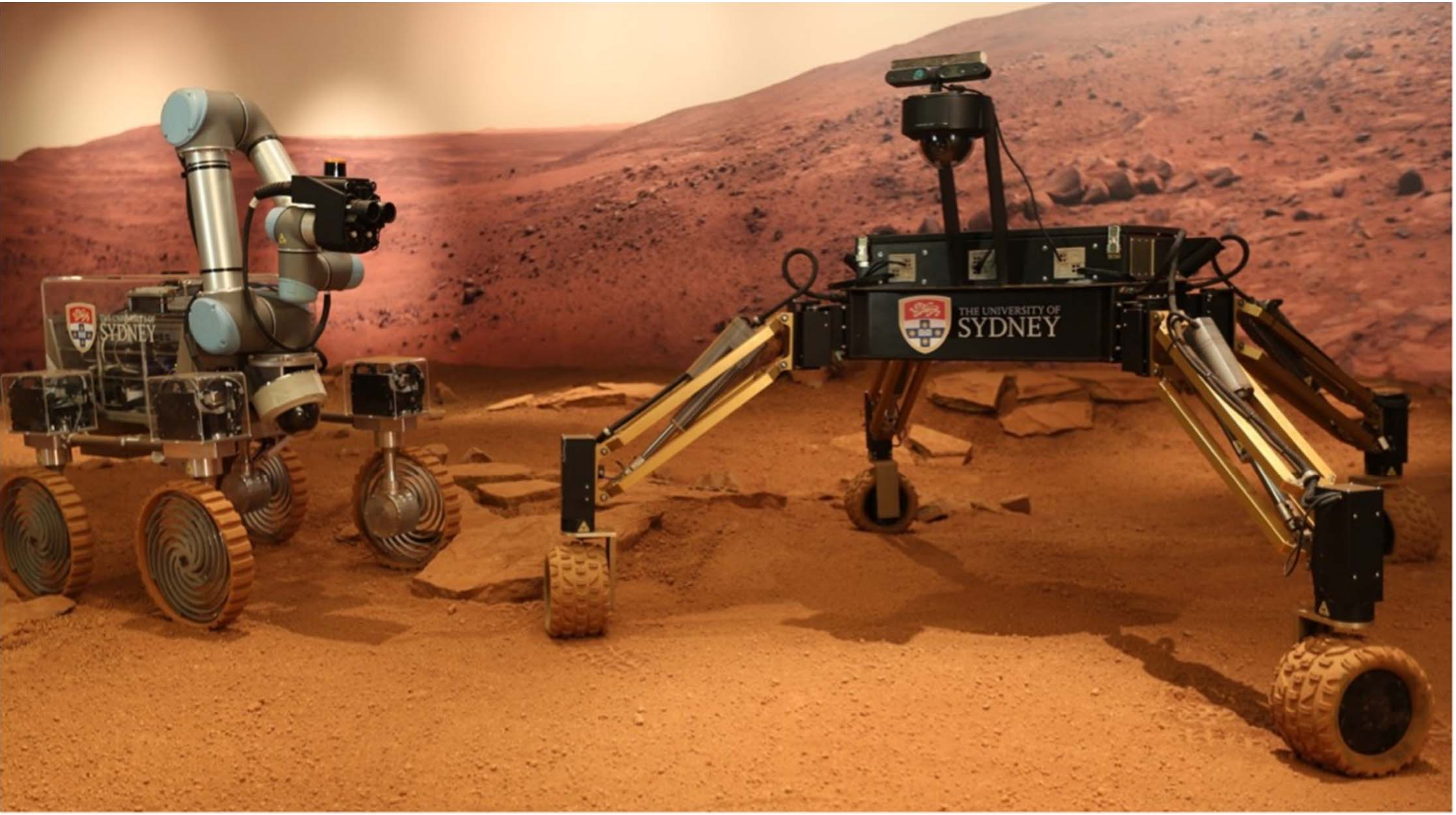
ACFR AUSTRALIAN CENTRE FOR FIELD ROBOTICS

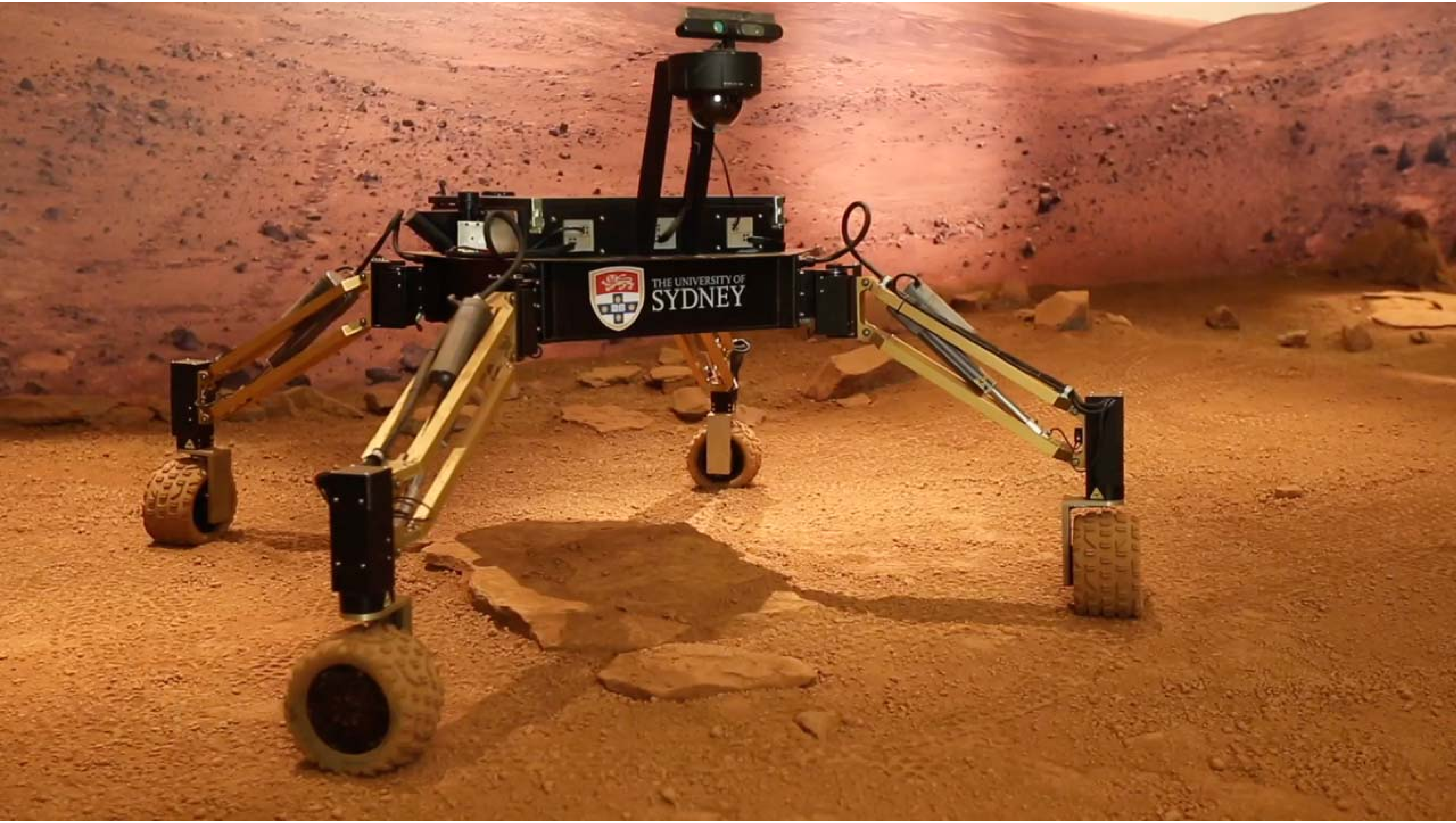


Introduction/Outline

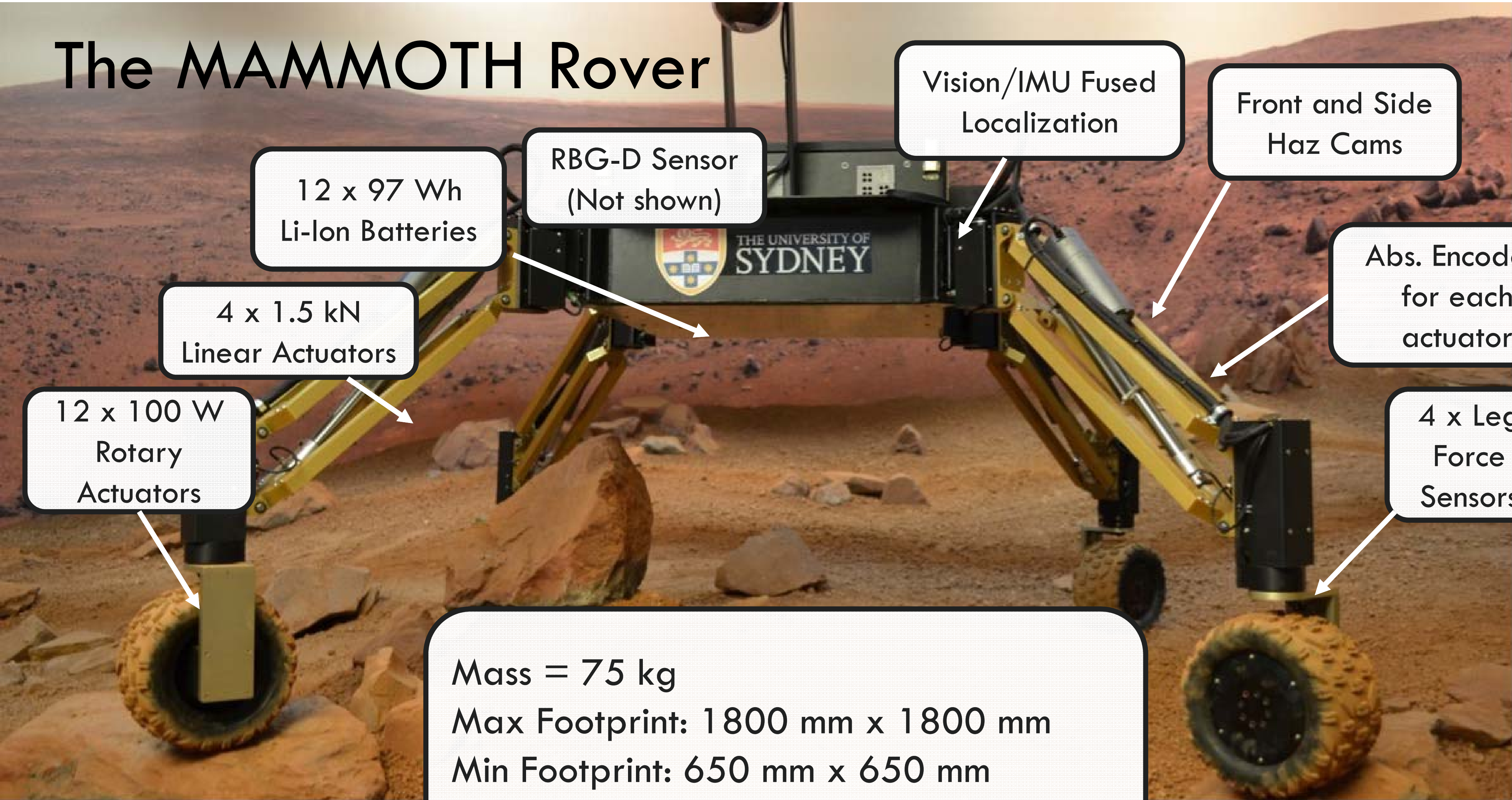
- Engineer/Educator/Space Enthusiast
- The Mars Lab
- The MAMMOTH Rover
- MATLAB/Simulink
- MAMMOTH Software
- Other MAMMOTH adventures


Schools
Australian Government
Department of Education and Training
THE MARS LAB
THE UNIVERSITY OF SYDNEY
UNSW
THE UNIVERSITY OF NEW SOUTH WALES
SYDNEY • AUSTRALIA
& Sciences

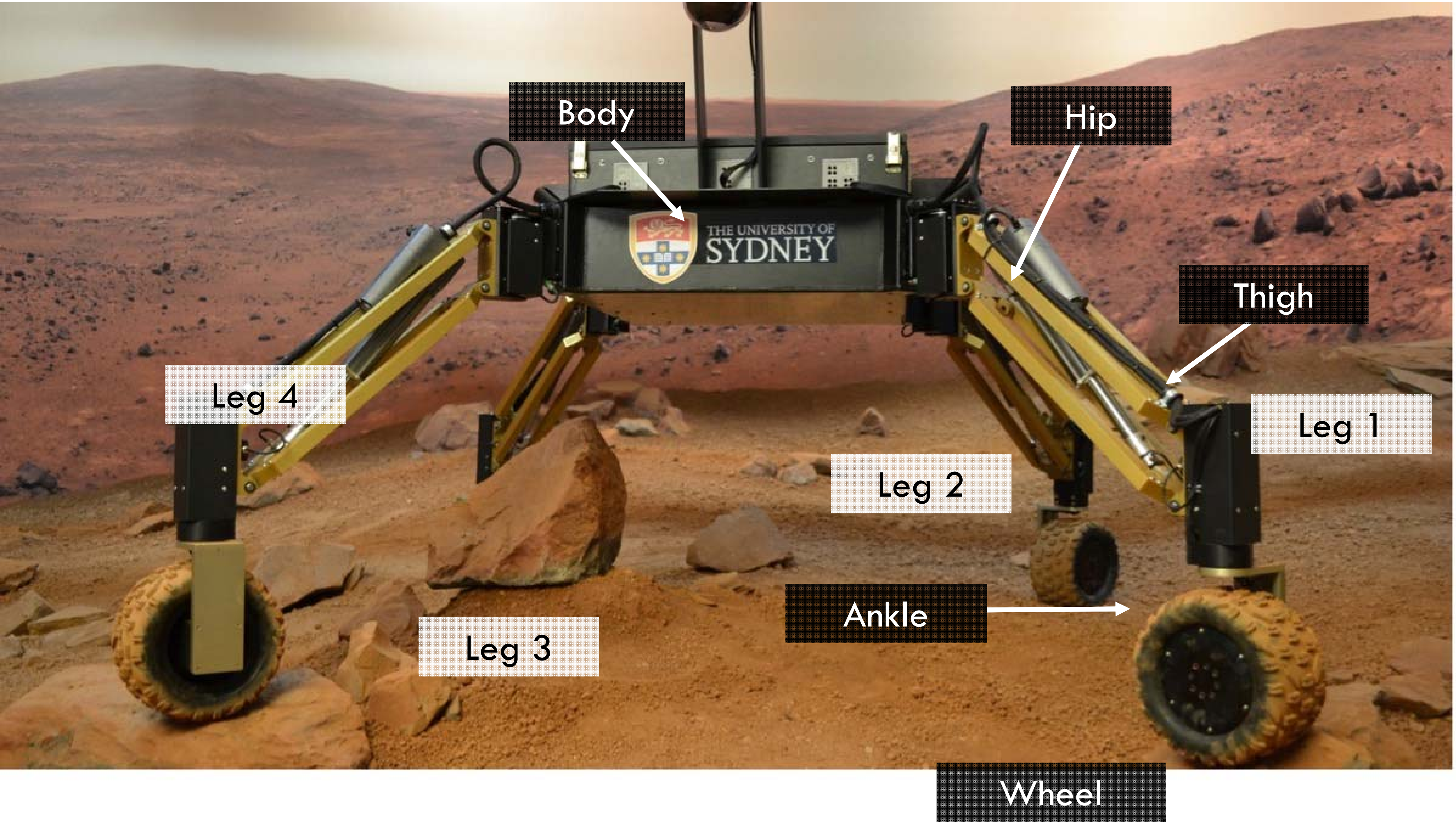

SYDNEY
THE UNIVERSITY OF
SYDNEY


THE UNIVERSITY OF
SYDNEY

The MAMMOTH Rover



Mass = 75 kg
Max Footprint: 1800 mm x 1800 mm
Min Footprint: 650 mm x 650 mm
Max Flat Plane Clearance: 800 mm


Body
Hip
Thigh
Leg 4
Leg 1
Leg 2
Ankle
Leg 3
Wheel
THE UNIVERSITY OF
SYDNEY

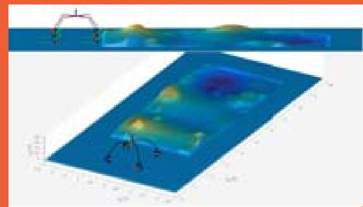
MATLAB/Simulink Software Development

Algorithmic Development

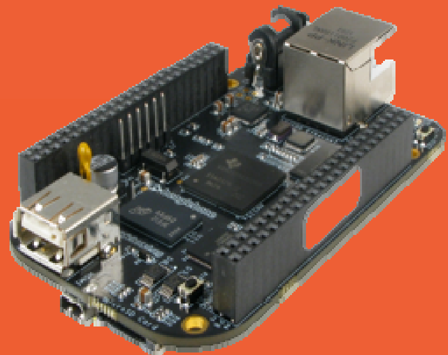
$$\underline{v}_C^W = B(\underline{p})\underline{V} + Q(\underline{\gamma})\dot{\underline{\gamma}} + \dot{\underline{\sigma}}$$

$$\dot{\underline{\zeta}} = (S_{Q_\zeta} Q_\zeta)^{\#} \left(\underline{v}^W_{C,z} - \left[S_B B \quad S_Q Q_\epsilon\right] \dot{\underline{\eta}}\right)$$

Software-In-The-Loop

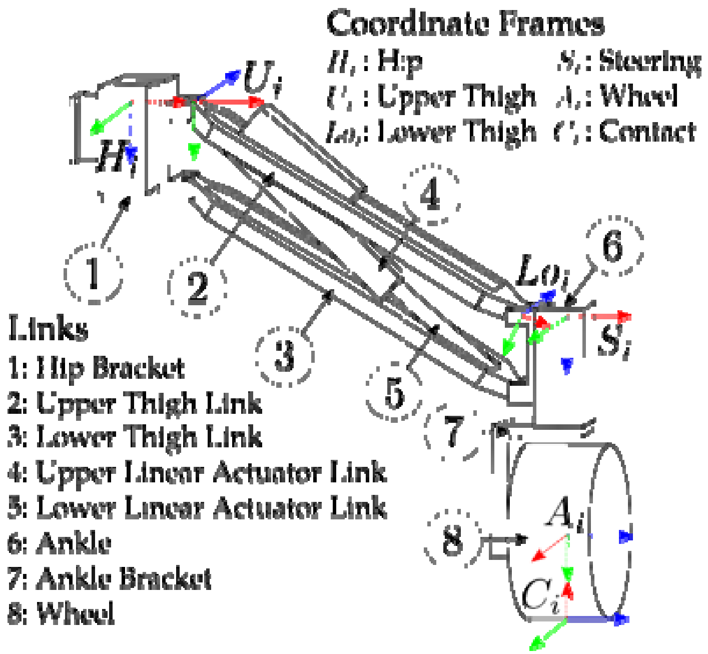


Hardware-In-The-Loop



C++ Generation for Deployment

Actively Articulated Suspension – Algorithmic Development



Body pose vector:

$$\underline{p} = \begin{bmatrix} x_B^W & y_B^W & z_B^W & \phi_B^W & \theta_B^W & \psi_B^W \end{bmatrix}^T$$

Body velocity vector:

$$\underline{V} = \begin{bmatrix} \underline{v}_B^W & \underline{\omega}_B^W \end{bmatrix}^T$$

Wheel contact point velocity:

$$\underline{v}_C^W = B(\underline{p})\underline{V} + Q(\underline{\gamma})\dot{\underline{\gamma}} + \dot{\underline{\sigma}}$$

Hip and thigh joints:

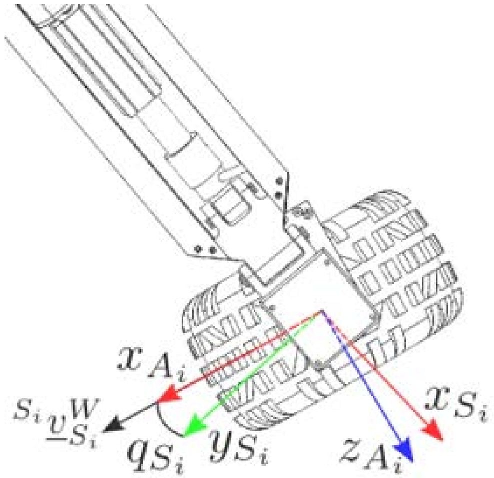
$$\underline{\epsilon}_i = q_{H_i}, \ \underline{\zeta}_i = \begin{bmatrix} q_{U_i} & q_{Lo_i} \end{bmatrix}^T$$

Single leg kinematic model:

$$\underline{v}_{C_i}^W = B(\underline{p})\underline{V} + Q_{\epsilon_i}(\underline{\epsilon}_i)\dot{\underline{\epsilon}}_i + Q_{\zeta_i}(\underline{\zeta}_i)\dot{\underline{\zeta}}_i$$

Actively Articulated Suspension – Algorithmic Development

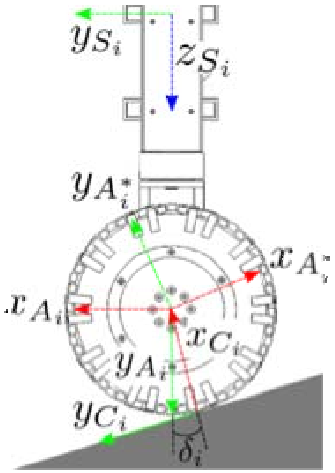
Steering Joint



$$ {}^{S_i}\underline{v}^W_{S_i} = R^{S_i}_W \left[B(\underline{p}) + Q_{\epsilon_i}(\underline{\epsilon}_i)\dot{\underline{\epsilon}}_i + Q_{\zeta_i}(\underline{\zeta}_i)\dot{\underline{\zeta}}_i \right] $$

$$ q_{S_i} = \arctan\left(\frac{{}^{S_i}\underline{v}^W_{S_i,x}}{{}^{S_i}\underline{v}^W_{S_i,y}} \right) $$

Drive Joint



$$ \dot{q}_{A_i^*} = \frac{R^{A_i}_W \underline{v}^W_{A_i}}{L} $$

Wheel contact-point velocity estimation

$$ v^W_{C_i,z} = \frac{\Delta r^W_{terrain_i,z}}{\Delta t} $$

Kinematic equation reduction

$$ \underline{v}^W_{C,z} = S_B B(\underline{p})\dot{\underline{p}} + S_Q Q(\underline{\gamma})\dot{\underline{\gamma}} $$

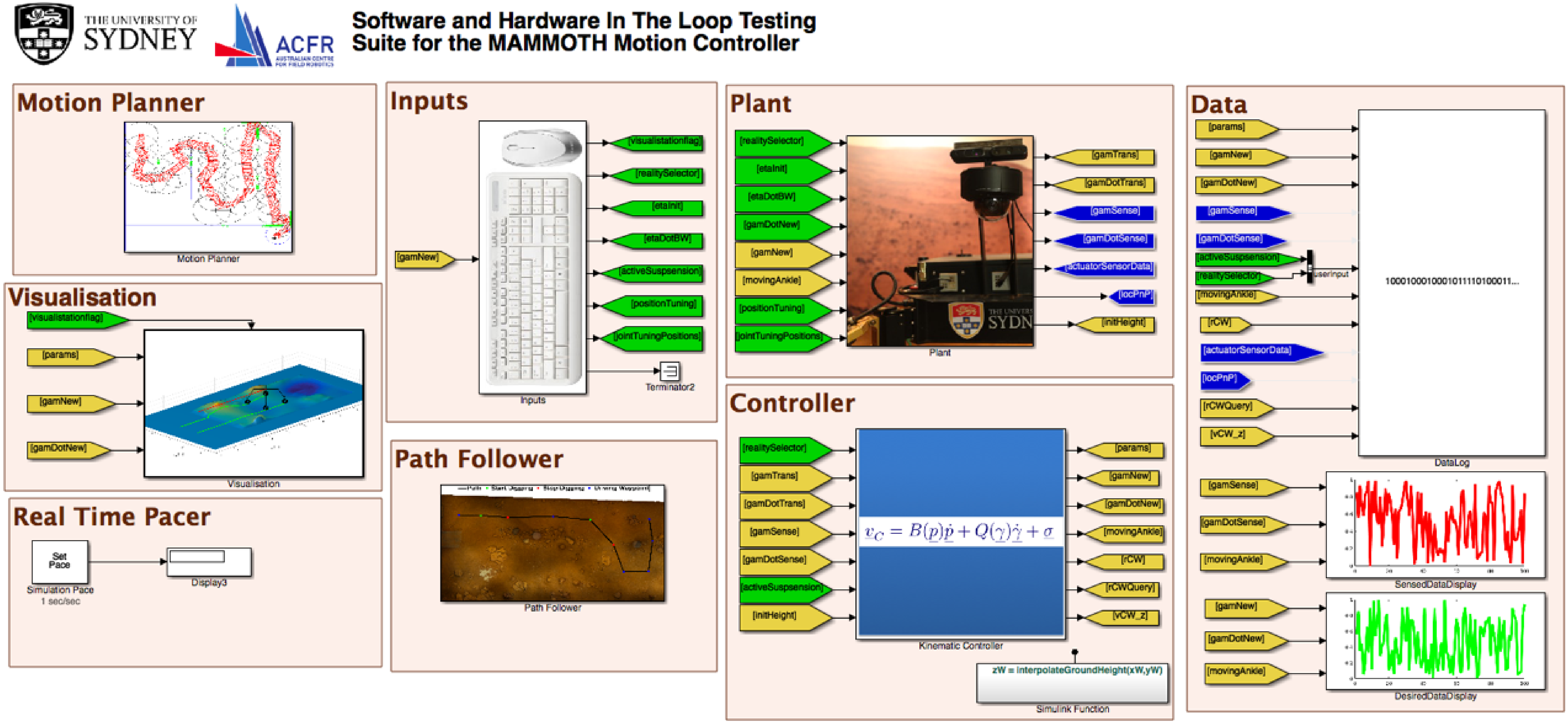
Re-arrange

$$ \begin{aligned} \underline{v}^W_{C,z} &= S_B B(\underline{p})\dot{\underline{p}} + S_{Q_\epsilon} Q_\epsilon(\underline{\epsilon})\dot{\underline{\epsilon}} + S_{Q_\zeta} Q_\zeta(\underline{\zeta})\dot{\underline{\zeta}} \\ &= \left[S_B B \quad S_{Q_\epsilon} Q_\epsilon \right] \dot{\underline{\eta}} + S_{Q_\zeta} Q_\zeta(\underline{\zeta})\dot{\underline{\zeta}}, \end{aligned} $$

Solve for the joint actuation rates

$$ \dot{\underline{\zeta}} = (S_{Q_\zeta} Q_\zeta)^{\#} \left(\underline{v}^W_{C,z} - \left[S_B B \quad S_Q Q_\epsilon \right] \dot{\underline{\eta}} \right) $$

Simulink Model



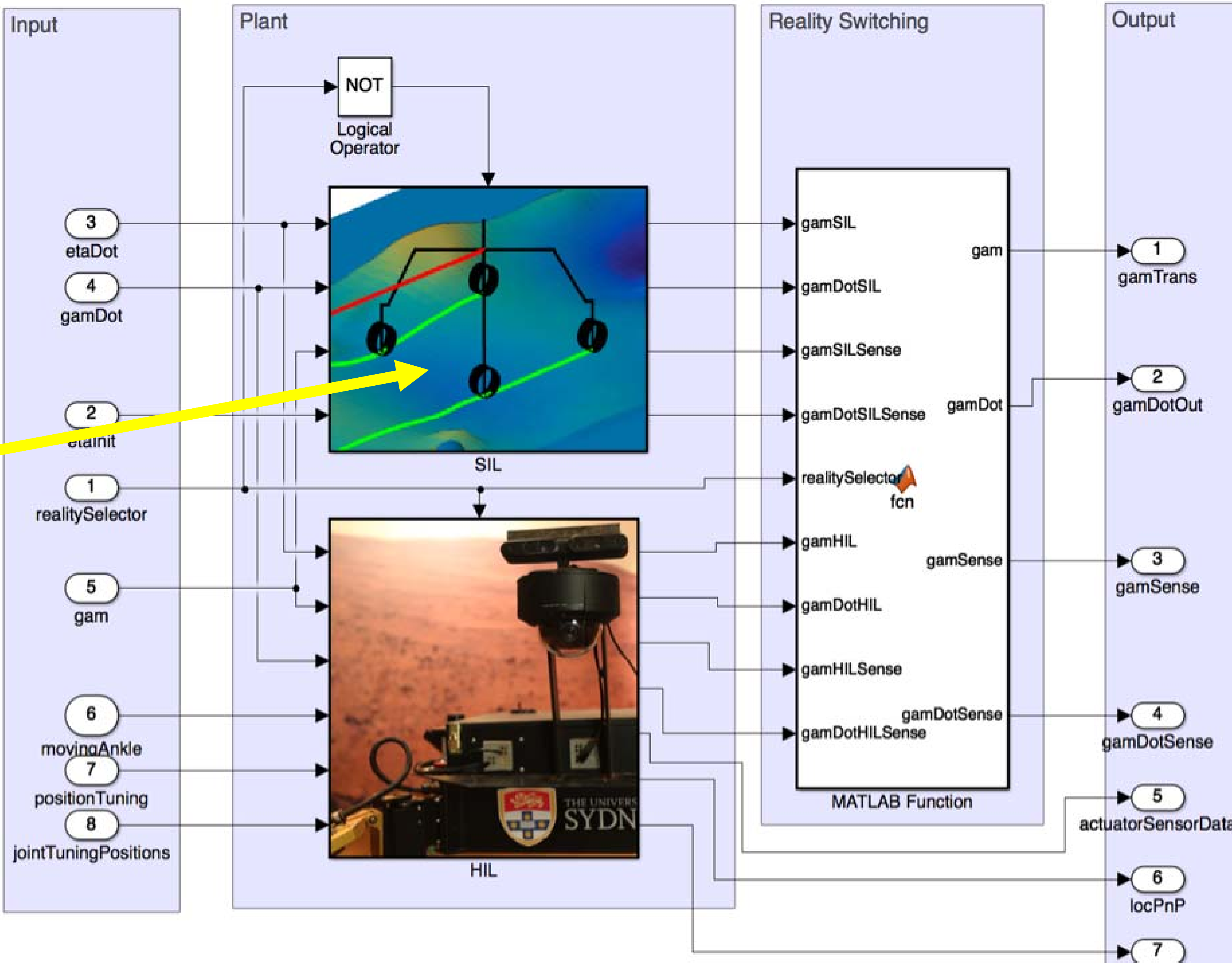
Simulink Model

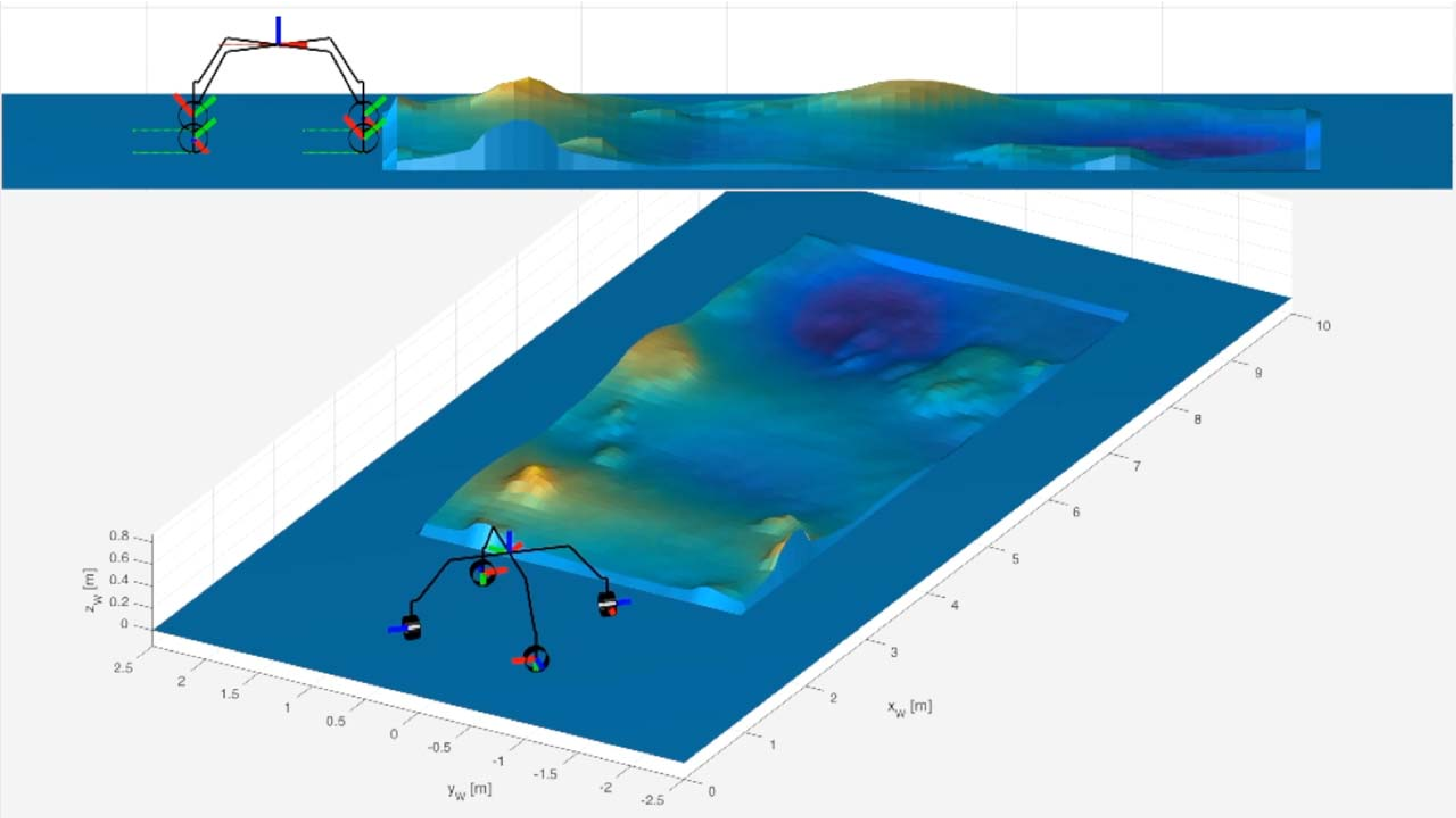
Common interface for SIL and HIL blocks

SIL uses Simulink numerical ODE solver to run native simulation

SIL may be interfaced with external simulation applications:

– Gazebo, V-REP





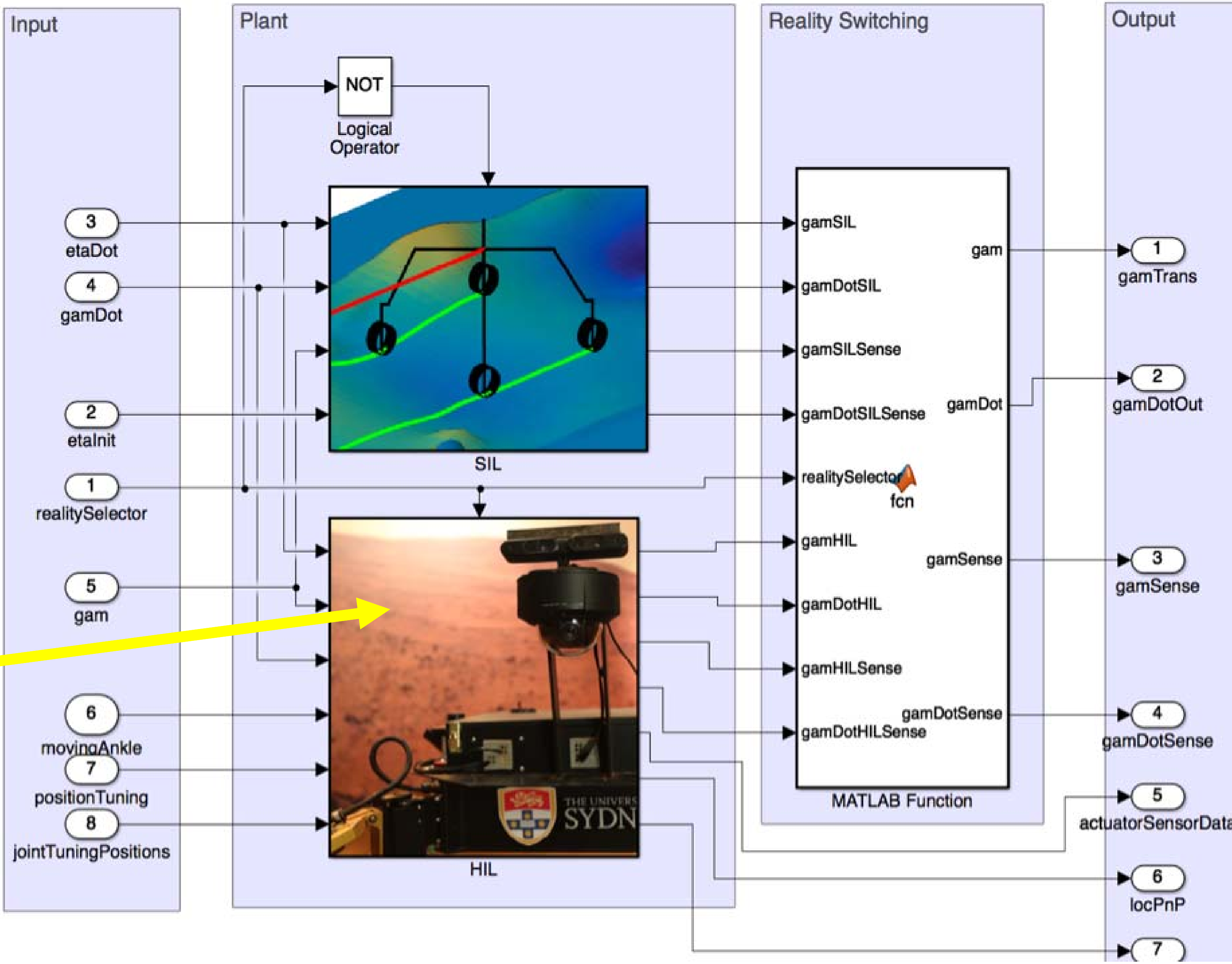
0.8
0.6
0.4
0.2
0
z_W [m]
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-2.5
y_W [m]
0
1
2
3
4
5
6
7
8
9
10
x_W [m]

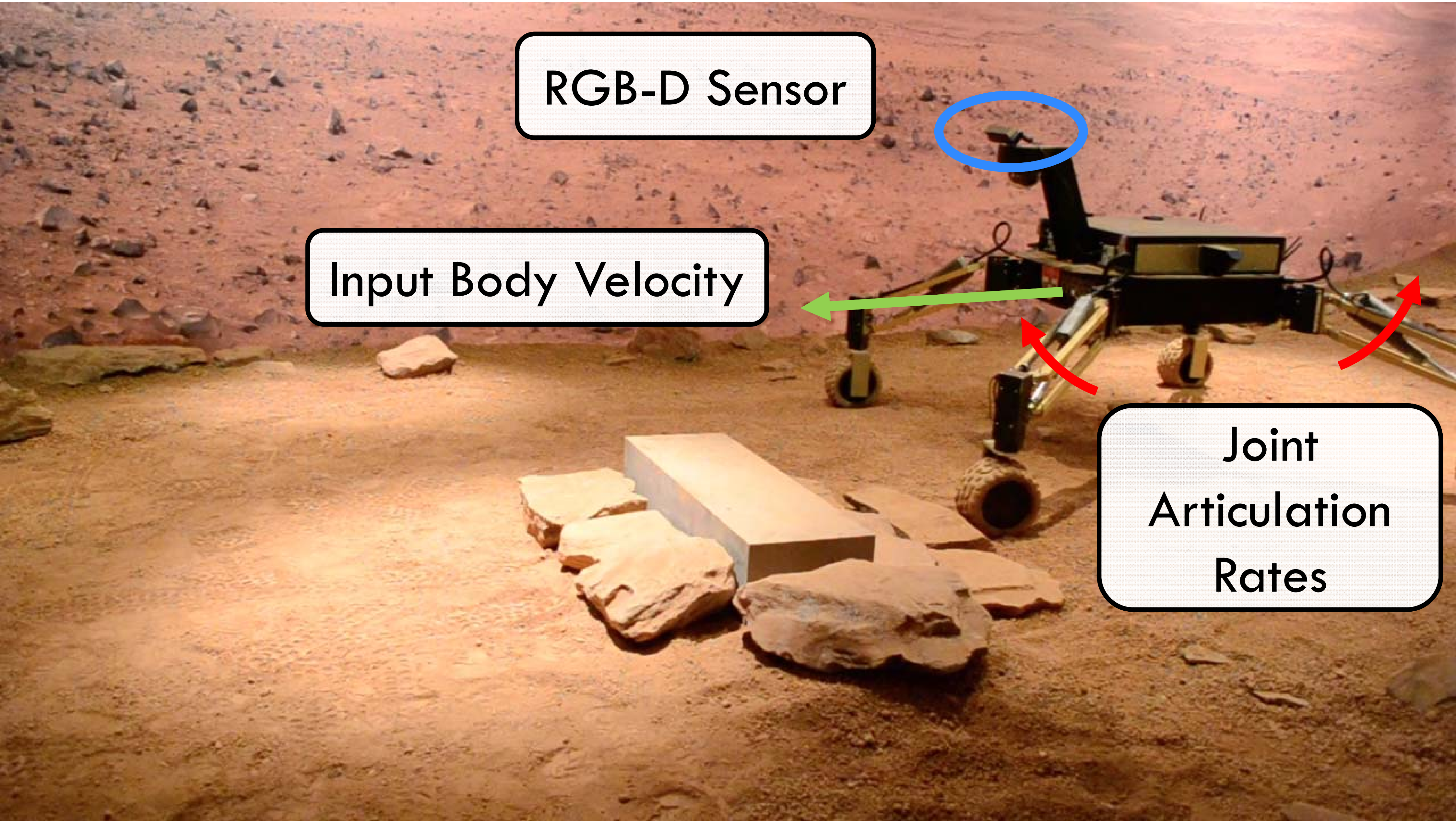
Simulink Model

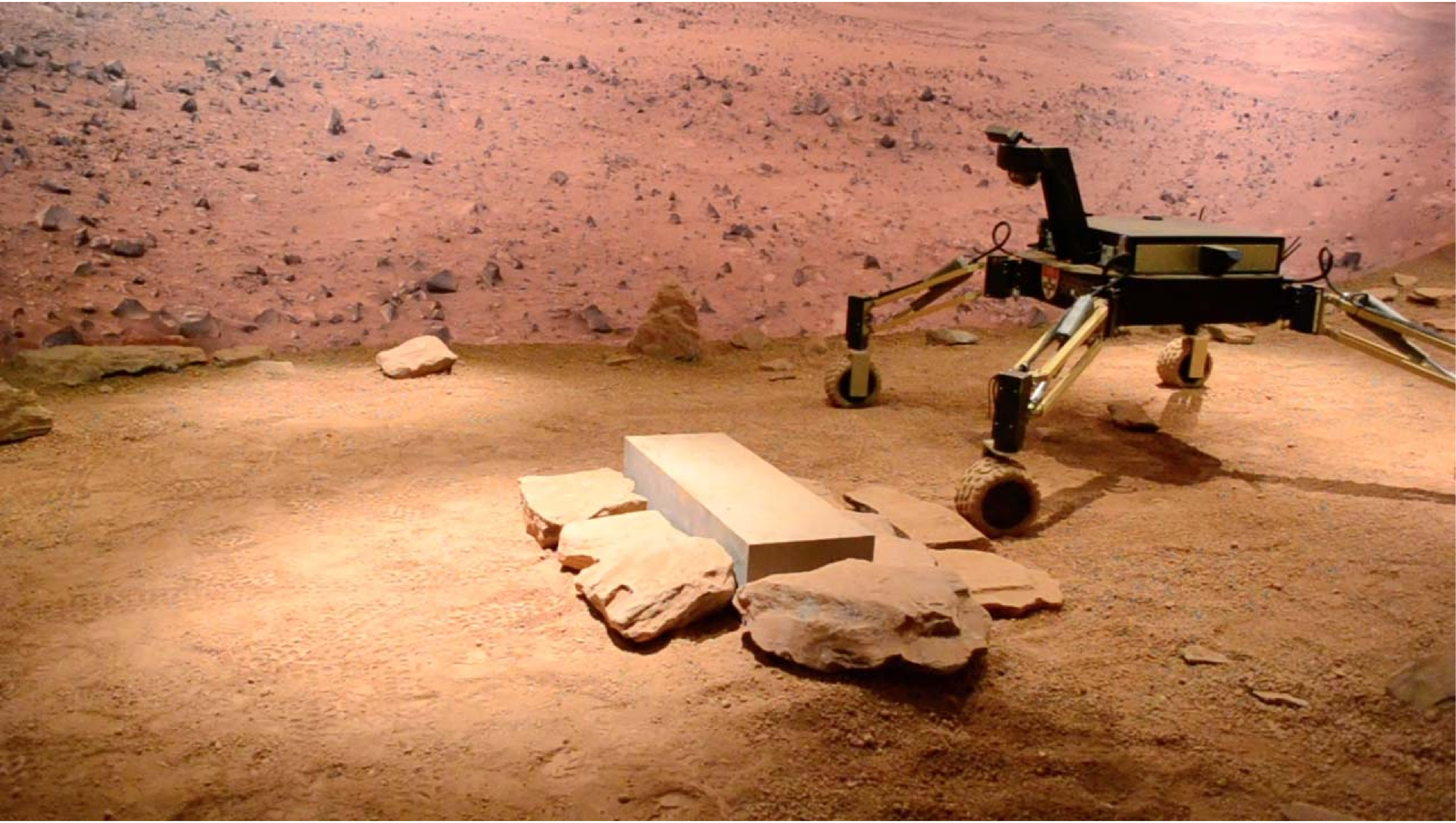
Common interface for SIL and HIL blocks

HIL receives and transmits inputs and outputs over LAN

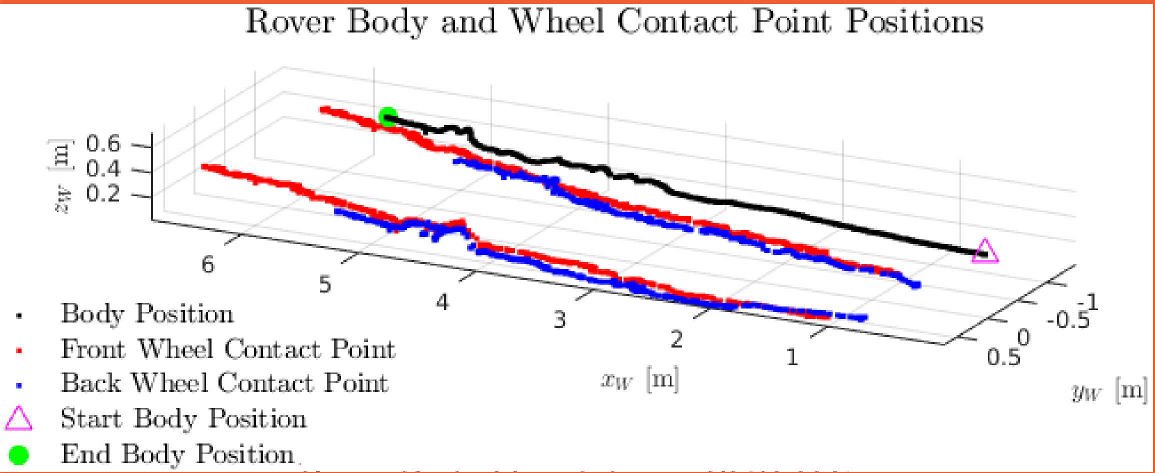
Robotics System Toolbox used to interface directly with ROS




RGB-D Sensor
Input Body Velocity
Joint
Articulation
Rates



Analytical Tools



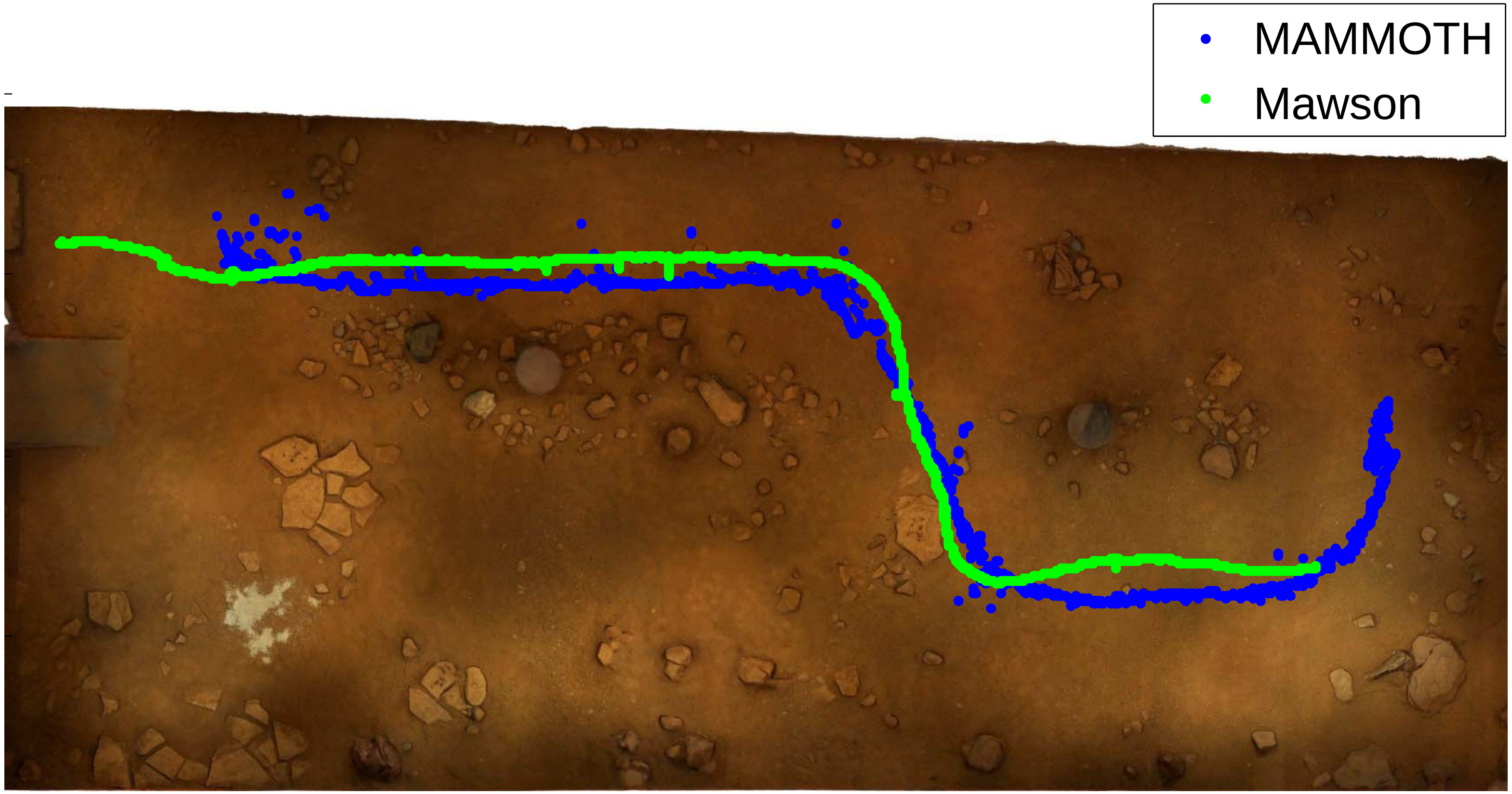
Other MAMMOTH Adventures – Digging Mission

MAMMOTH is used to dig a series of trenches along an operator-defined path

The Mawson Rover follows MAMMOTH to inspect each of the trenches

Multi-rover digging mission video

Digging Mission




MAMMOTH the Farmer

Conclusions

- Novel wheel-on-leg platform
- Model-based design for rapid software development
- Interfacing with various sensors/actuators/software-in-the-loop applications
- Variety of applications