Robonet:

A Distributed Data Architecture for Embedded Systems

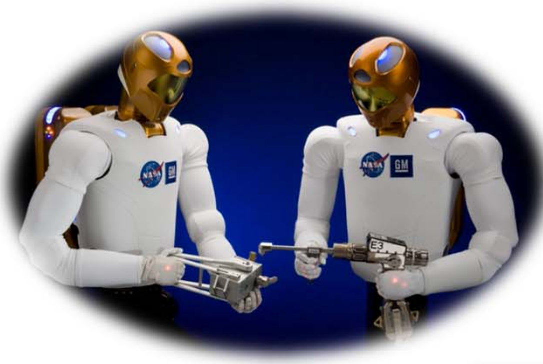
Presented by:
Phil Strawser
LEG Device TIM
8/8/2012

Outline

- Robotics Development
- Motivation and Background
- Vision
- Robonet Components
- Other I/O architectures
- How is Robonet better?
- Current & Potential Applications
- Backup slides

Evolving Paradigm for Rapid Development of World Class Robots

- Software, Robotics and Simulation Division uses a rapid development model that leverages core technologies
- Typical build cycle lasts 1 year







Robonaut Development History

- 1998
  - Subsystem Development
  - Testing of hand mechanism
- 1999
  - Single Arm Integration
  - Testing with teleoperator
- 2000
  - Dual Arm Integration
  - Testing with dual arm control
- 2001
  - Waist and Vision Integration
  - Testing under autonomous control
- 2002
  - R1A Testing of Autonomous Learning
  - R1B Integration
- 2003
  - R1A Testing Multi Agent EVA Team
  - R1B Segwanaut Integration
- 2004
  - R1A Autonomous Manipulation
  - R1B 0g Airbearing Development
- 2005
  - DTO Flight Audit
  - Begin Development of R1C
- 2006
  - Centaur base
  - Coordinated field demonstration

ROBONAUT
Fall 1998

ROBONAUT
Fall 1999

ROBONAUT
Fall 2000

ROBONAUT
Fall 2001

ROBONAUT
Fall 2002

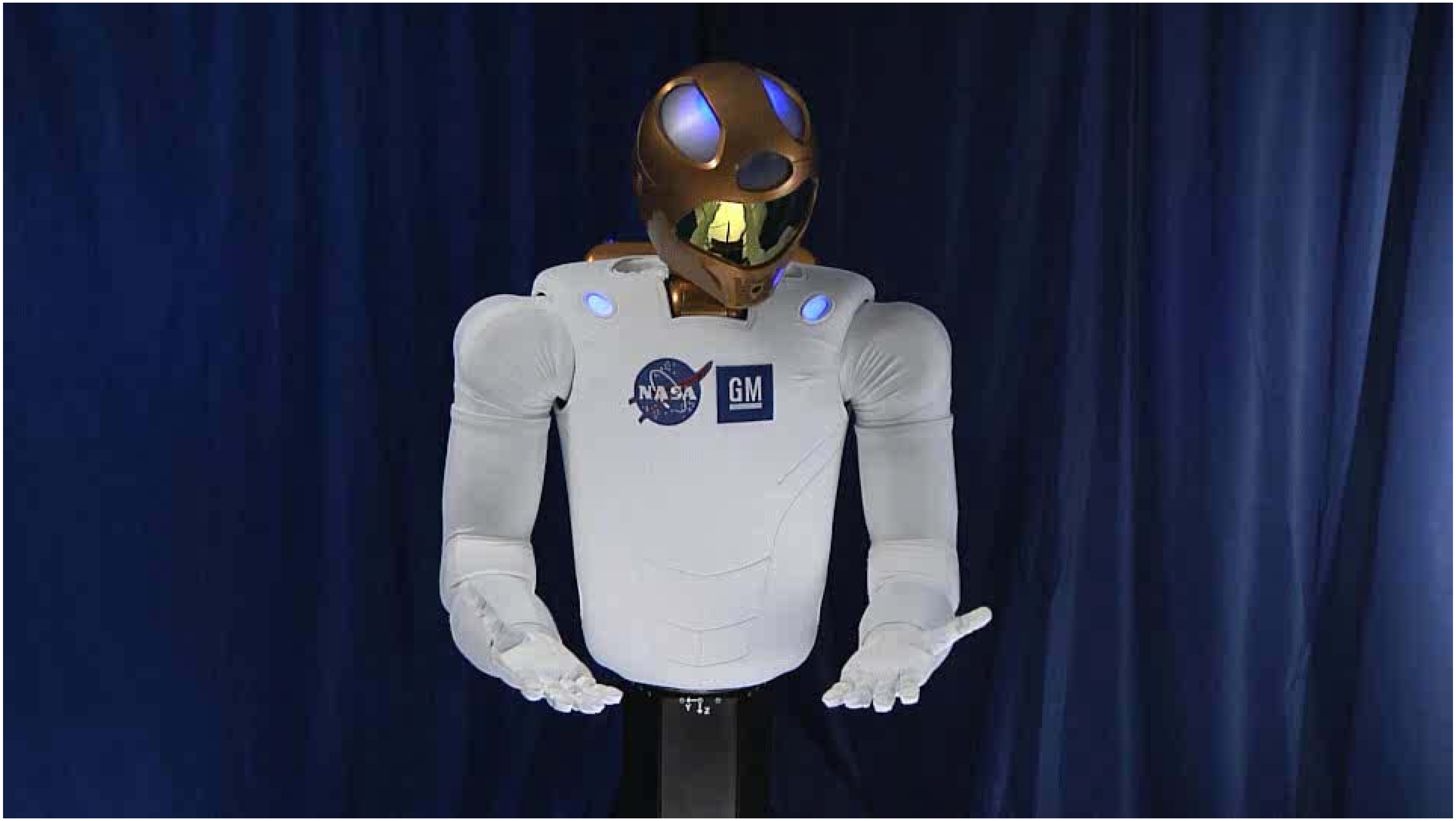
ROBONAUT
Fall 2003

ROBONAUT
Fall 2004

ROBONAUT
Fall 2006

Robonaut 2 (R2)

- 2007 – 2009 (26 months)
- Launched aboard STS-133 in February of 2011



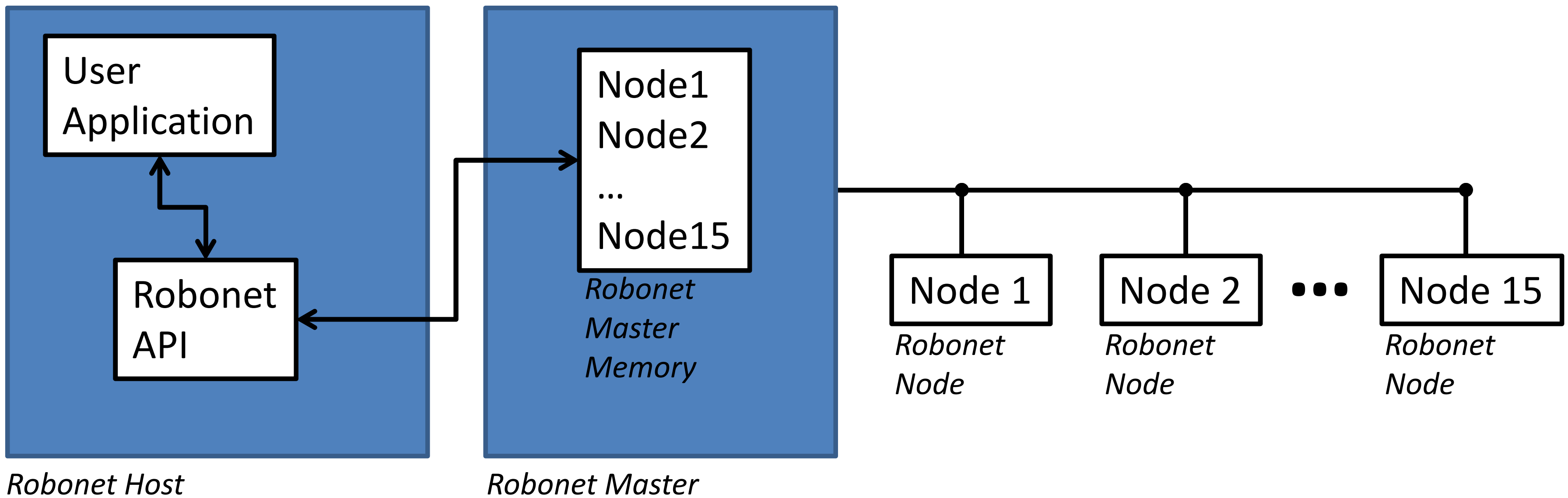
Motivation and Background

- High-performance robotic systems have the following data requirements:
  - Millisecond latency
  - KHz loop rates
  - Tight physical constraints
    - Low physical packaging footprint
    - Low wire count
- Developer objectives:
  - Portability across many unique platforms
  - Easily deployable software/firmware interfaces

Vision Behind Robonet

- What?
  - Create a high-performance, flexible data infrastructure for a network of embedded devices
- Why?
  - No existing protocol sufficiently satisfy all the hard problems for distributed robotic systems
- How?
  - Define a communication protocol between a system master and a network of nodes over a two-wire interface
  - Implement a run-time configurable, memory-mapped I/O architecture

Robonet System Diagram



Robonet Host

- implements the software API for all of the hardware and data contained in the lower levels of the system.
- This software layer can reside along-side user-level controller applications for the robot
- The host can be any microprocessor capable of communicating with a Robonet system master
- The host configures the master’s “schedule” at run-time

Robonet Master

- The master manages the communication between the host system and the nodes
- All communication is initiated by the master
- The master contains the data for all nodes in the system
- Configured at run-time to schedule data transmitted over the network
- Once configured, the Master can move data around completely independent of the host, freeing the host to work on other processes

Robonet Node

- Interfaces hardware (sensor data, status, etc) with the Robonet data bus.
- The firmware for running a Robonet node can be loaded to a suitably capable FPGA that interfaces to an MLVDS transceiver
- Examples of suitable FPGAs:
  - Xilinx Spartan3
  - Virtex4
  - Actel families
  - and many others.

Current I/O Protocols

- I^2C - 3.4Mbit/s, single-ended data, single-ended clock
- Ethernet/Ethercat – fast, but 8 conductor and point-to-point
- Rapid IO – made for chips on a circuit board, boards across a back-plane
- CAN bus – only 1-Mbit
- USB/IEEE 1394 – very fast, but > 4-wires, point-to-point
- Space Wire – 400Mbps, but 9-wires, point-to-point

How is Robonet Better?

- Two-wire – fewer wires means fewer electrical connections, fewer failures, smaller physical footprint
- High-speed (50 Mbps -> ~2kBytes @ 1Khz) – necessary for high-performance, high-bandwidth embedded applications
- Multi-point – Single pair of wires for multiple nodes. Allows multiple nodes to listen to all traffic on the bus without re-transmitting data
- Memory-mapped architecture – hardware implementation agnostic paradigm allowing maximum flexibility and adaptability for different configurations

NO OTHER I/O ARCHITECTURE HAS ALL OF THESE ATTRIBUTES

Current Applications

- Robonaut 2
  - Robonet is the backbone communication path between the brainstem and all joints inside the robot
  - Currently deployed in R2A at JSC, R2B on ISS, R2C1 at JSC, R2C2 at General Motors
- Centaur2
- SEV
- R2 Zero-G Climbing legs (not pictured, still in development)

NASA Exoskeleton Development

- NASA's Interest in Exoskeletons
  - Exoskeletons provide potential for improved astronaut health countermeasures and performance augmentation

- K-Glove
  - Developed in Cooperation with General Motors
  - Utilizes R2 hand technology to provide user with increased grip strength and endurance

- X1
  - Developed in Cooperation with IHMC (Florida Institute for Human Machine Cognition)
  - Robotic exoskeleton for Resistive Exercise, Rehabilitation and Mobility Augmentation
  - From conceptual design to manufacturing complete in 10 months
  - Leverages technology from R2 aboard International Space Station

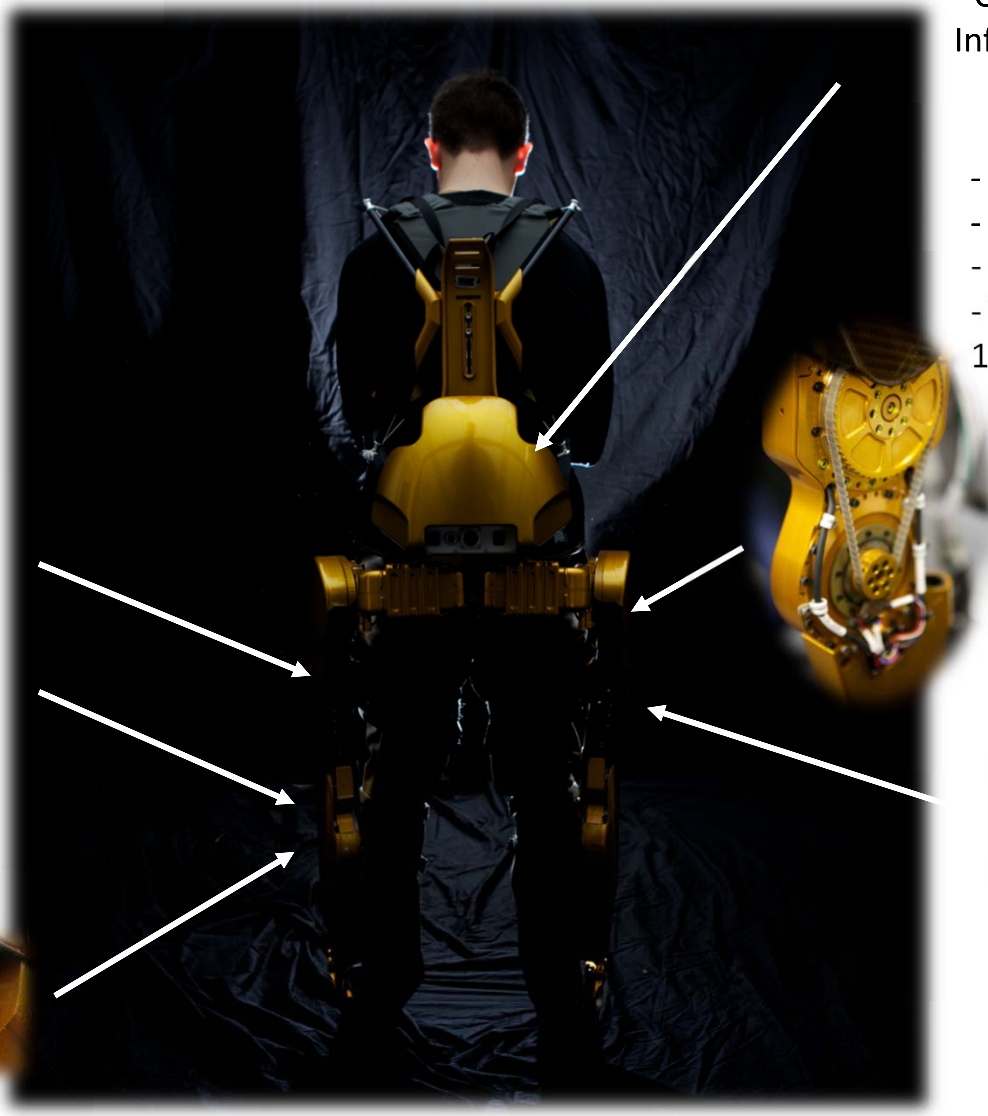
X1

4 Active DOFs
- Knee Extension/Flexion
- Hip Extension/Flexion

6 Passive DOFs
- Abduction/Adduction
- Internal/External Rotation
- Dorsiflexion/Plantarflexion

Custom Drivers at Each DOF
- Distributed control of joints
- ARM R4F LS Processor
- Actel FPGA
- 10 kHz loop rate
-Current, Temp Sensing, DAQ

Integrated Hard stops at Each Joint



Computer
- Onboard High Level Control
- RT Java running on Linux
- Custom Communication Infrastructure (RoboNET)

Series Elastic Actuators
- Low Profile
- Torque Sensing
- No Load Speed: 8 rad/s
- Torque: 300 Nm Peak , 150 Nm Continuous

Lightweight Composites

X1 User Interface

- Comprised of two shin cuffs, two thigh cuffs, waist belt and shoulder harness
- Designed for comfort and breathability
- Mechanically adjustable cuffs allow for variations in body types
- Current materials space qualified

- Exo in use movie

Other applications

Medical – small foot-print, low wire-count ideal for prosthetics

Robotics – industrial or commerical

Automotive – communication between a central processor and sensor packages

…any system containing a network of embedded systems