Seminar Paper Presentation:

Software Architecture of the da Vinci Research Kit

Zihan Chen et. al

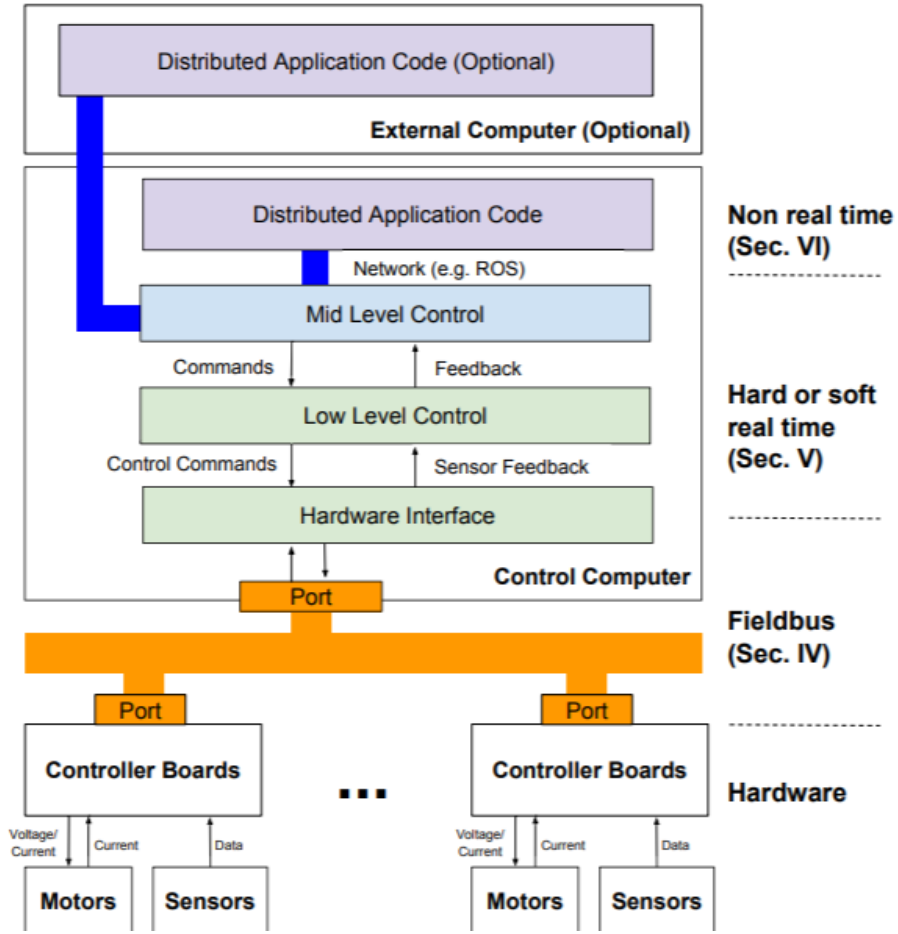
Anurag Madan (Group 3)

3/5/19

Computer Integrated Surgery II

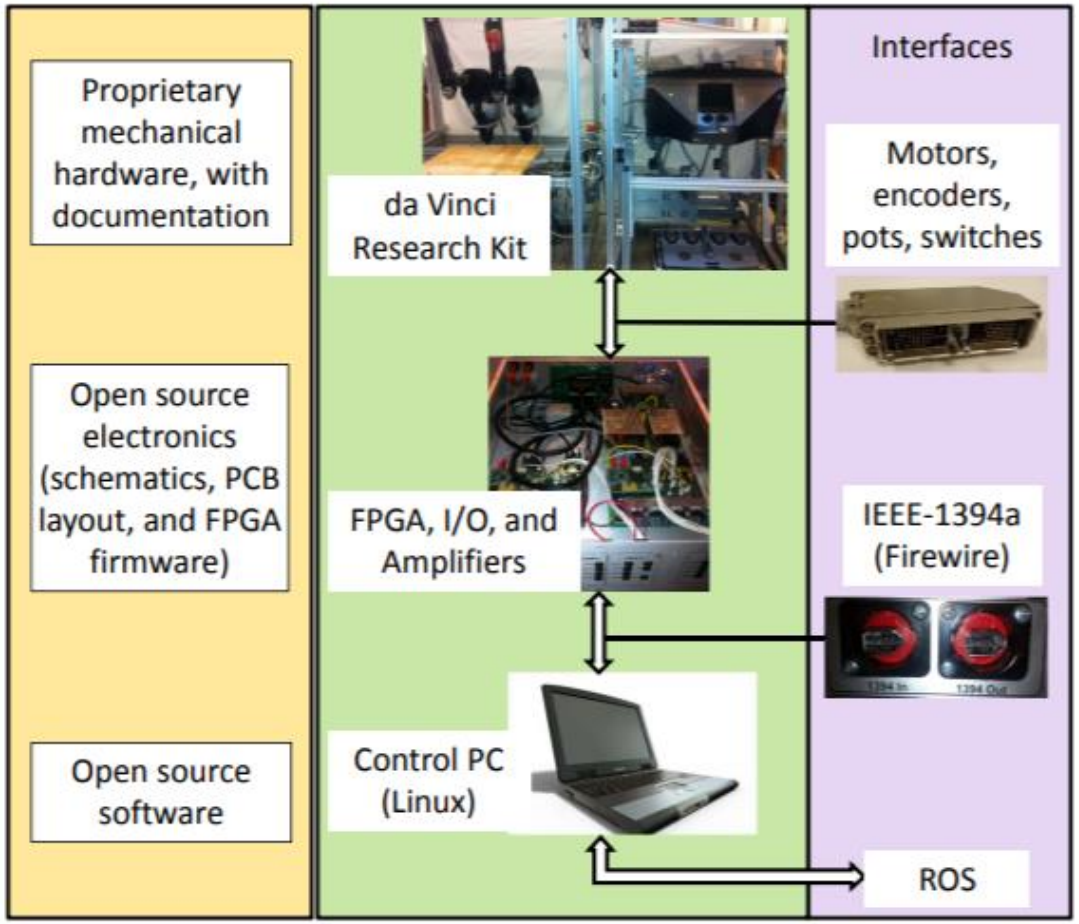
Background

- Telerobotic systems have a proven track record in several application domains
- Mostly direct teleoperation in robotic systems, largely due to lack of a robust common research platform
- Recently addressed by platforms like DVRK



DVRK system overview

- Mechanical components
  - 2 Master Tool Manipulators
  - 2 Patient Side Manipulators
  - High Resolution Stereo Viewer
  - Footpedal tray
- Electrical components
  - FPGA Board
  - Quad linear amplifier



Goal

- Provide a common research platform for the DVRK
  - Scalable
  - Reconfigurable
  - Real-time
  - ROS compatible

Scalable and Reconfigurable Hardware Interface

- Design goals
  - Provide deterministic performance with low latency – minimize total number of communication transactions
  - Support daisy chain connections – allows scalability and reconfigurability by introducing a different control PC
  - Sufficient bandwidth for all hardware
- Design analysis
  - FireWire used to implement daisy chain
  - Ethernet port used as bridge between PC and FireWire network

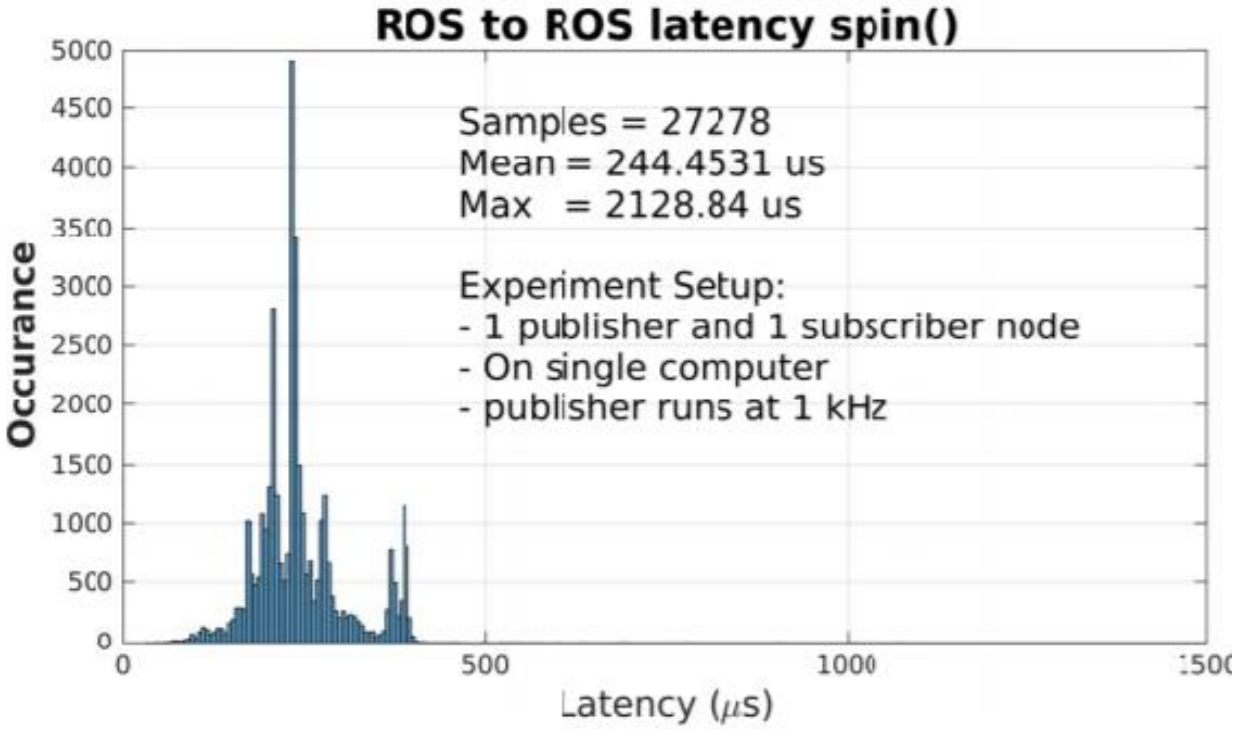
Scalable and Reconfigurable Hardware Interface

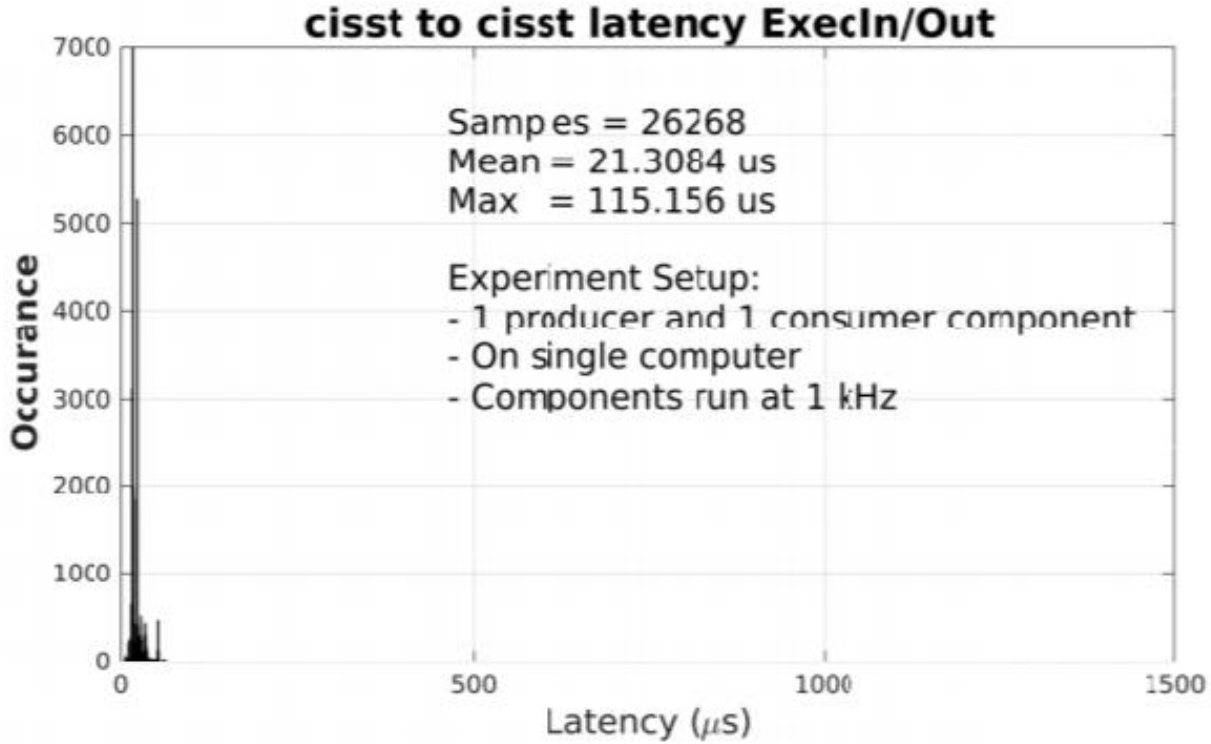
- Implementation
  - Use FireWire broadcast and P2P connection
  - Query, and wait for 5N us, N being number of boards
  - Each board broadcasts feedback after 5n us, n being node number
  - PC reads complete status information from one board, broadcasts control as single packet.
  - Each board extracts own commands based on node number
  - Ethernet used by PC to communicate with first board, FPGA firmware communicates between boards via FireWire network

Real Time Framework

- Design goals
  - A component-based framework, with well-defined interfaces between components
  - Efficient communication between components to support control rates of 1 kHz or more
- Design analysis
  - Using ROS subscriber gets reliable data transmission between controllers of different frequency
  - ROS publisher and subscriber have significant communication latency for high frequency controllers
  - Using multi-threaded component like JHU CISST reduces latency

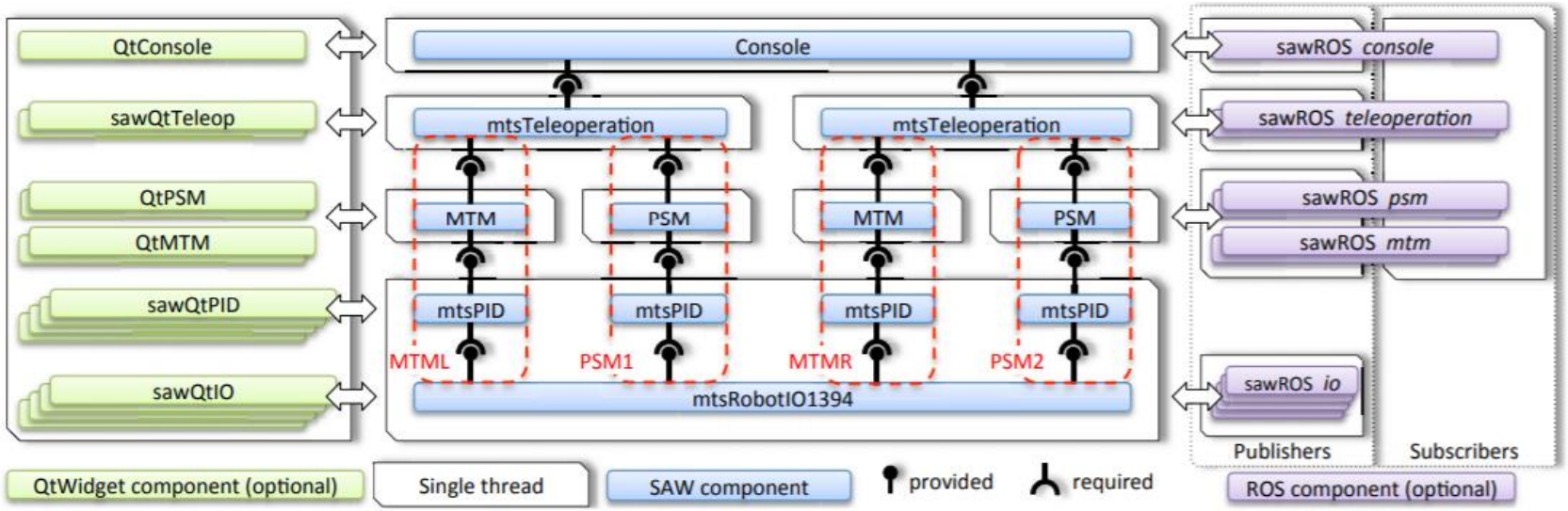
Real Time Framework





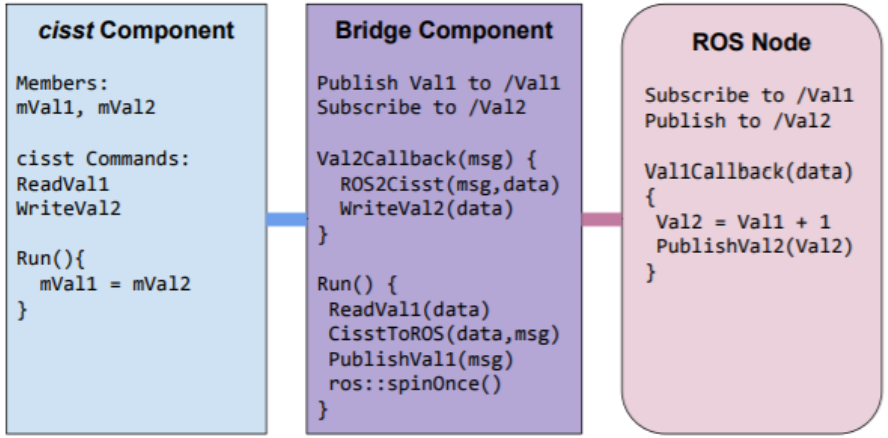
Real Time Framework

- Implementation
  - Shared memory, multi-threaded design is better suited for the high frequency, low-latency control requirements for the dVRK
  - Extend from hardware interface to low level and mid level controller



System ROS integration

- CISST to ROS Bridge
  - Bridge based design to communicate between CISST and ROS
  - Contains a CISST publisher and subscriber, conversion functions, and bridge component
  - CISST converted to Catkin packages, allows for easier access from ROS
  - Robot models converted to URDF format for simulation purposes



Conclusion and Relevance

- Presented a scalable, reconfigurable, real-time and ROS-compatible software architecture for DVRK
  - distributed hardware interface
  - real-time component-based framework with multi-threading
  - High level integration with the ROS ecosystem
- CISST architecture used to run many robots, and software element of project will involve this architecture.

References

- Chen, Z., Deguet, A., Taylor, R. H., & Kazanzides, P., (2017). *Software Architecture of the da Vinci Research Kit.* Proceedings – *1st IEEE International Conference on Robotic Computing, IRC 2017*, *2017* *http://doi.org/10.1109/IRC.2017.69*