

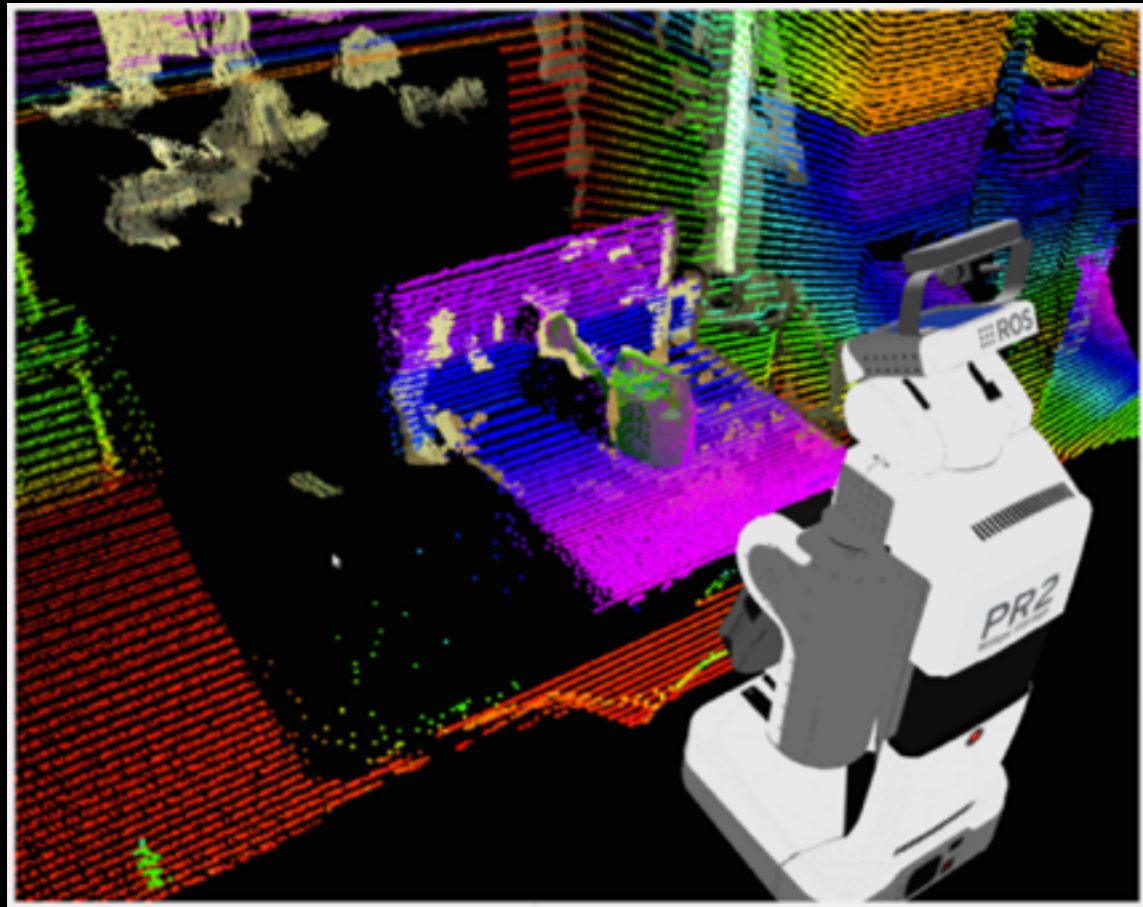


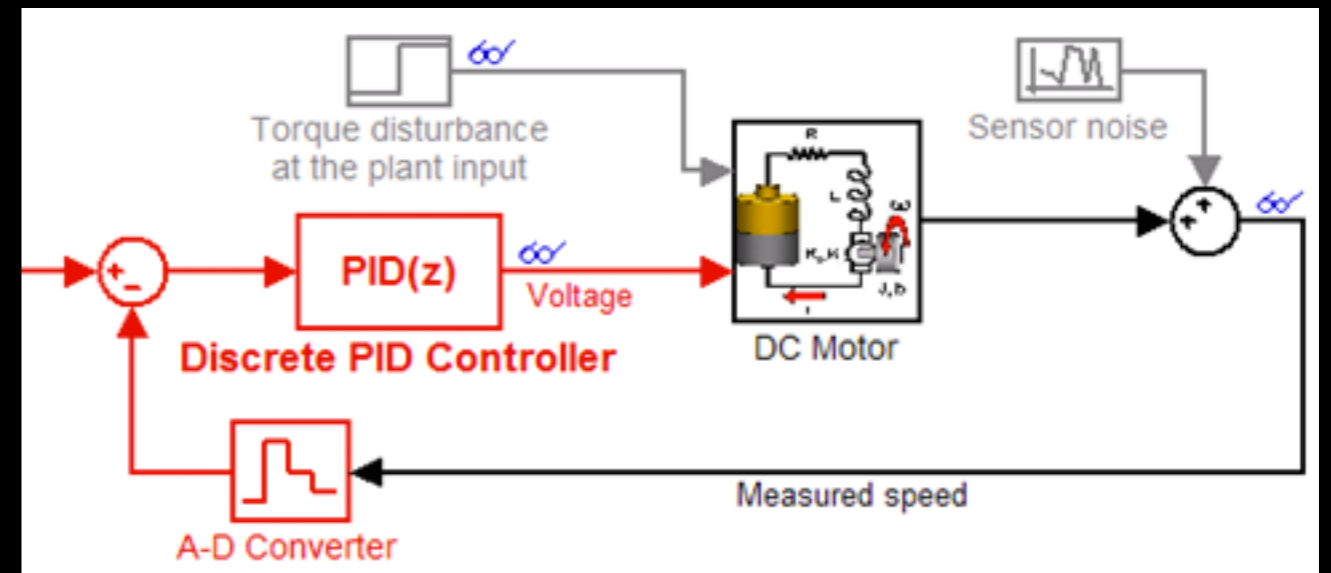
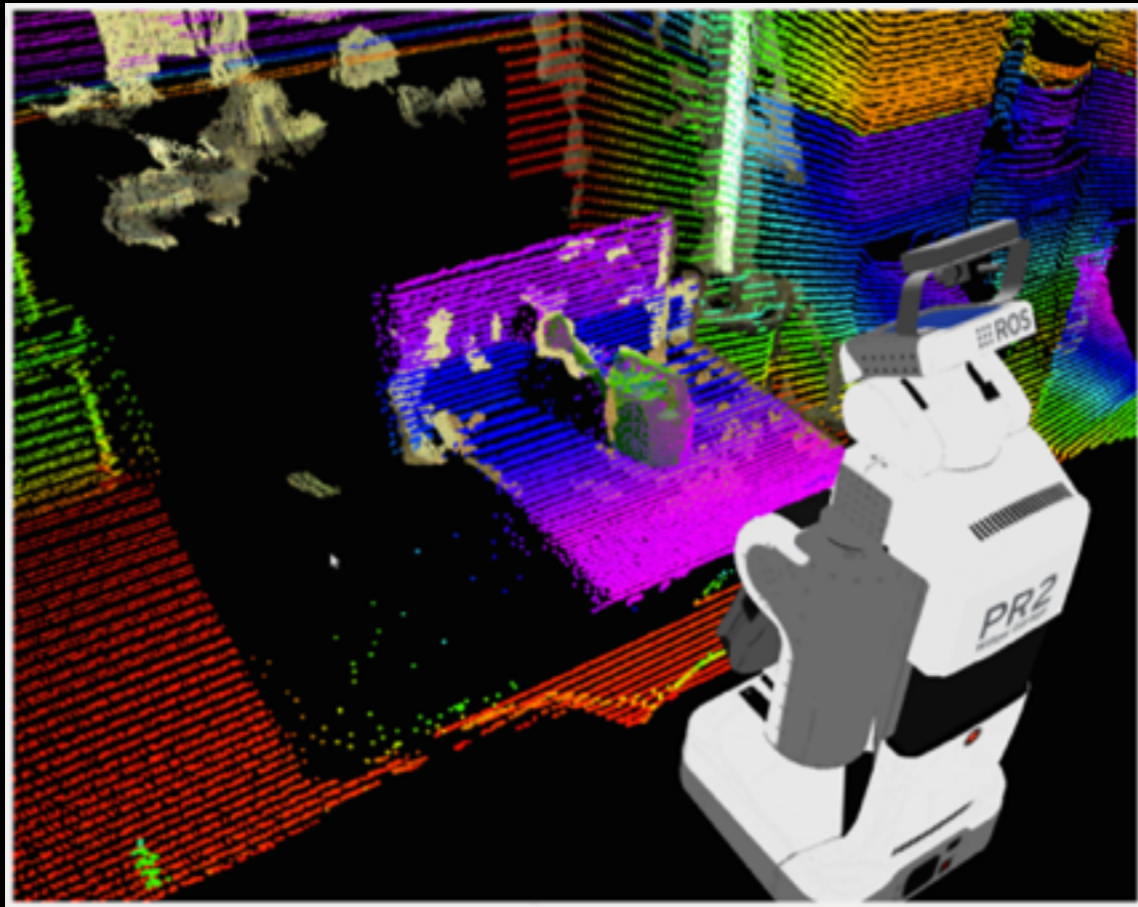
**SERIOUS
ROSSERIAL**

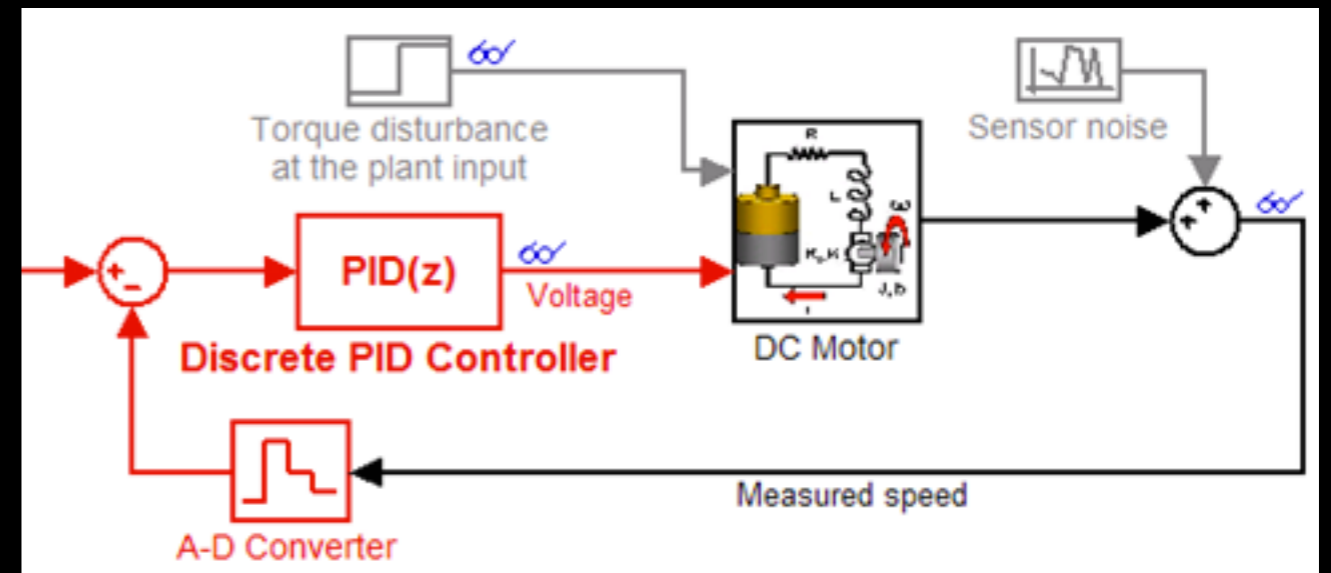
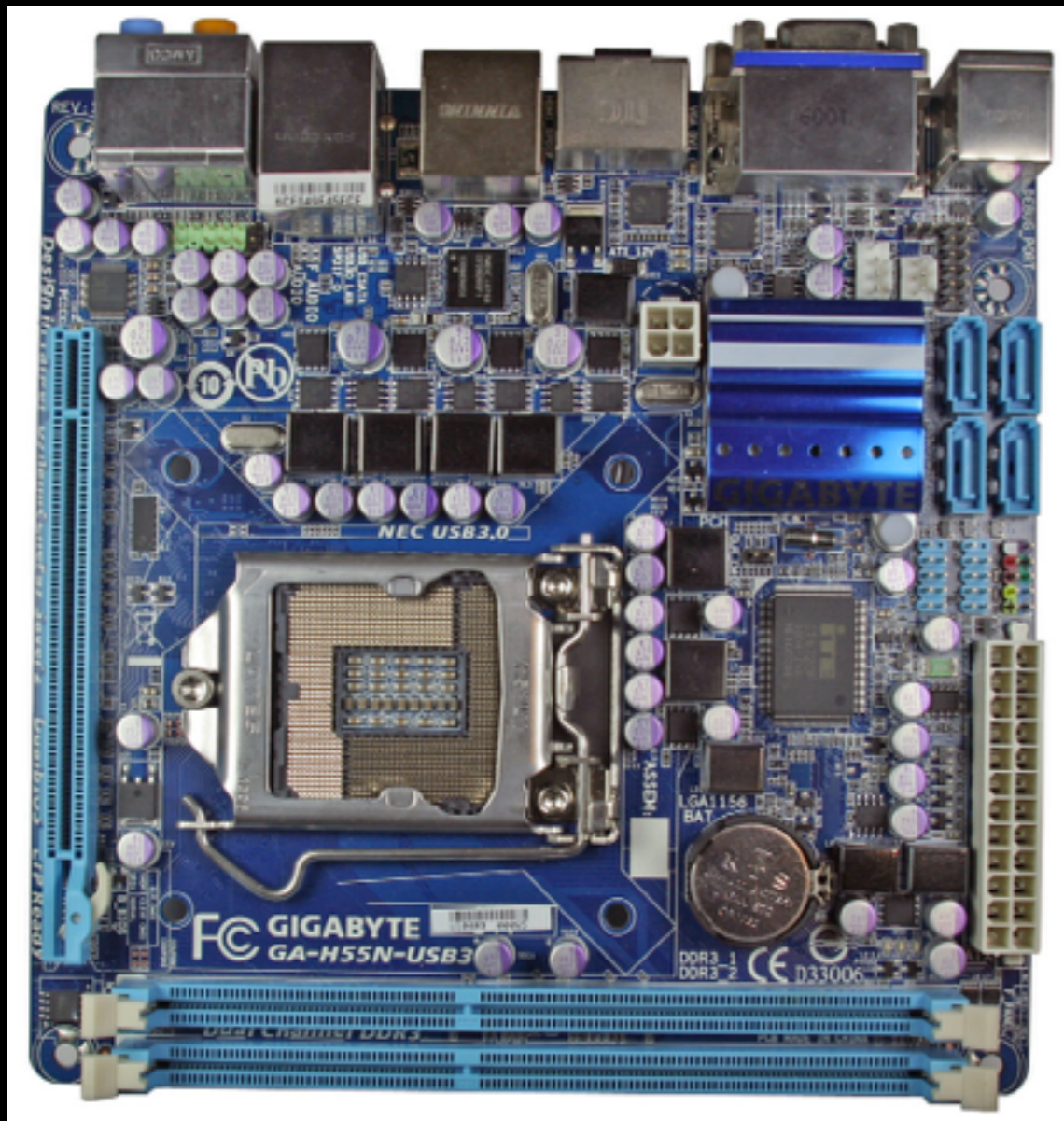
**MIKE PURVIS
ROSCON 2014**

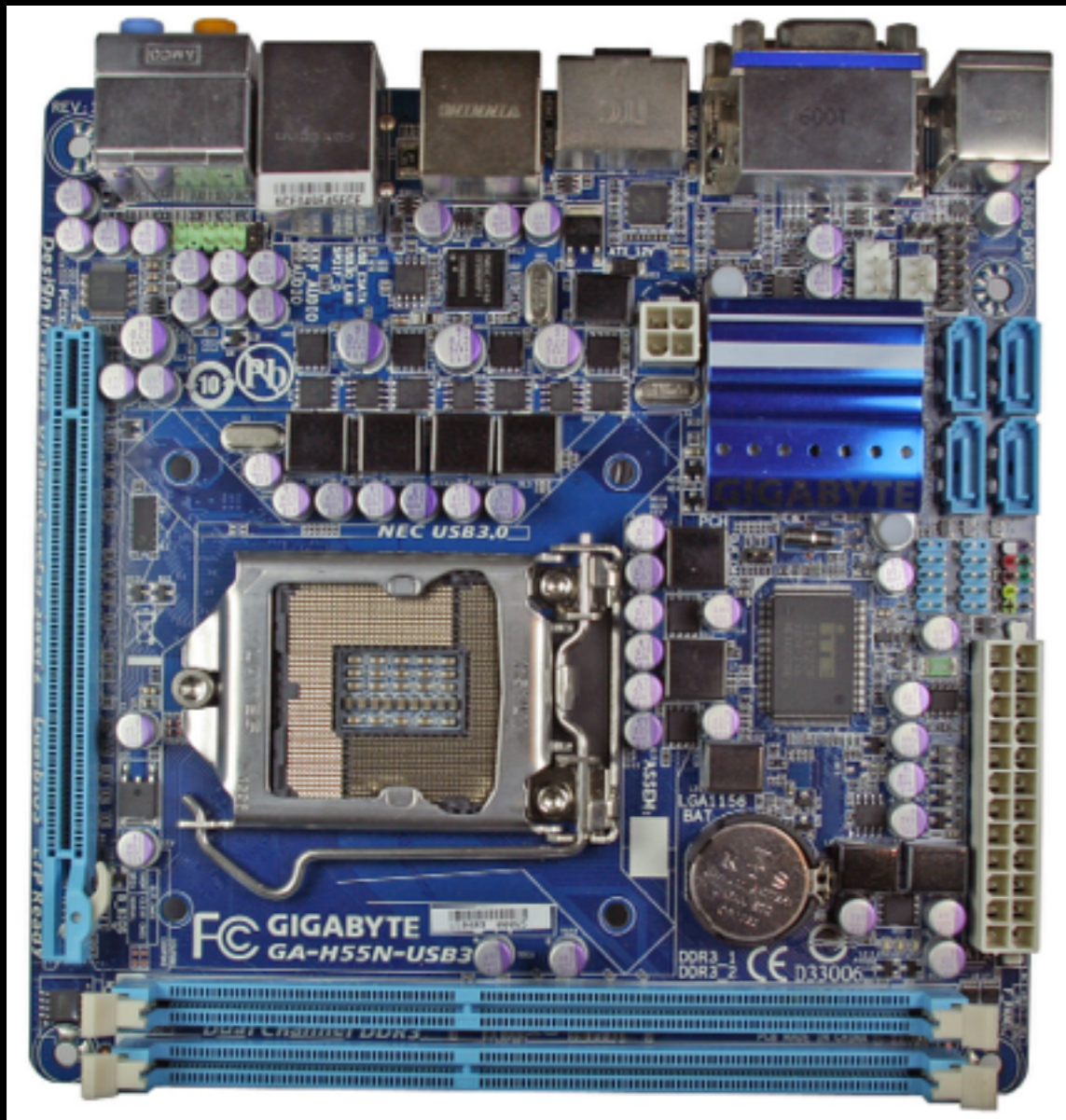


CLEARPATH
ROBOTICS™

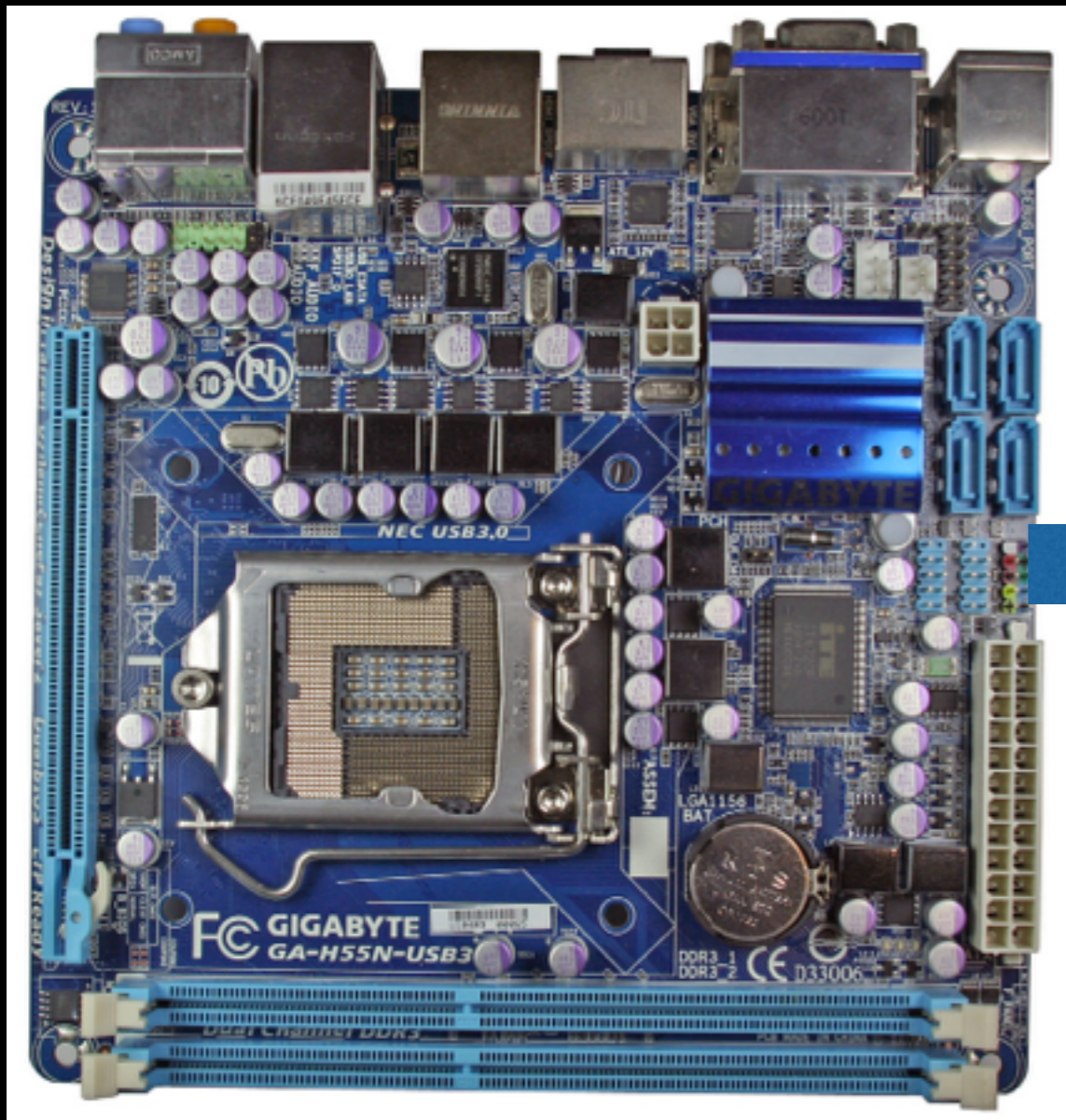




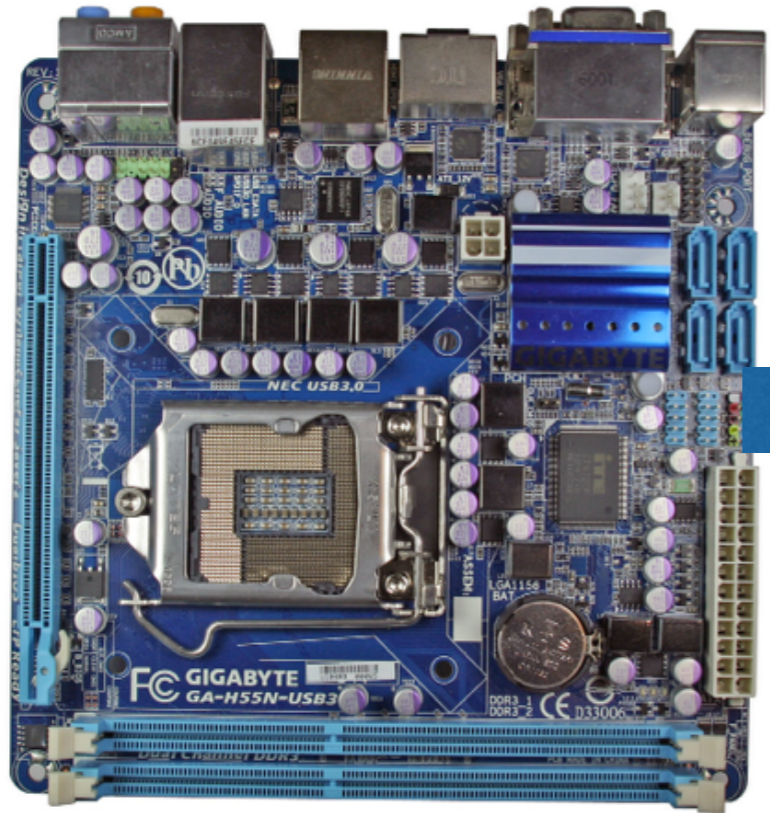




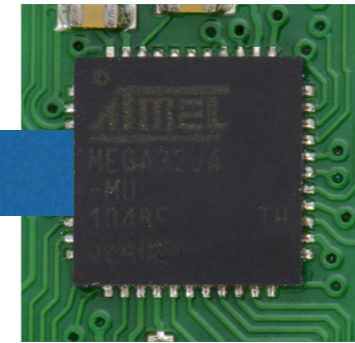
EtherCAT 

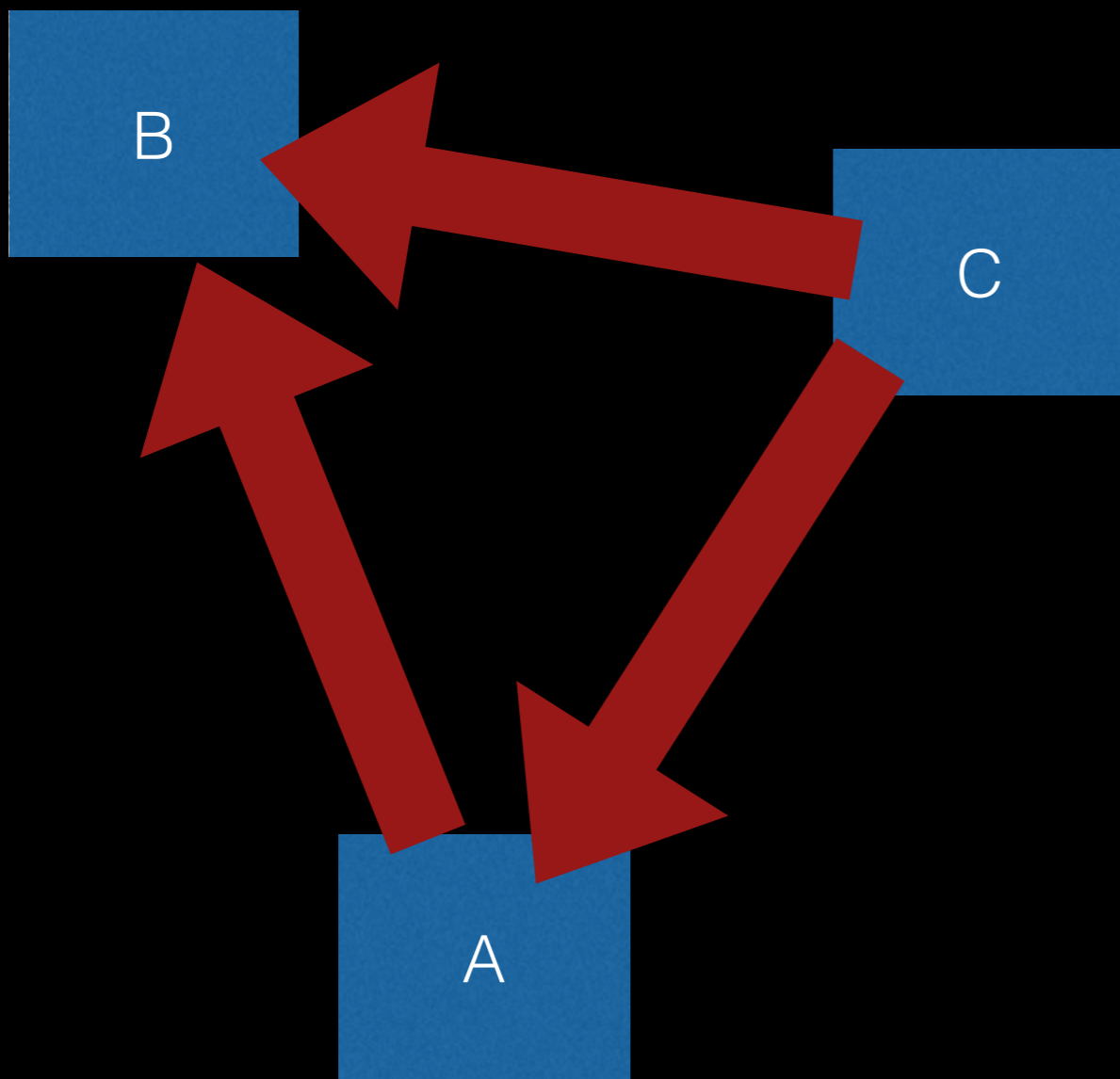


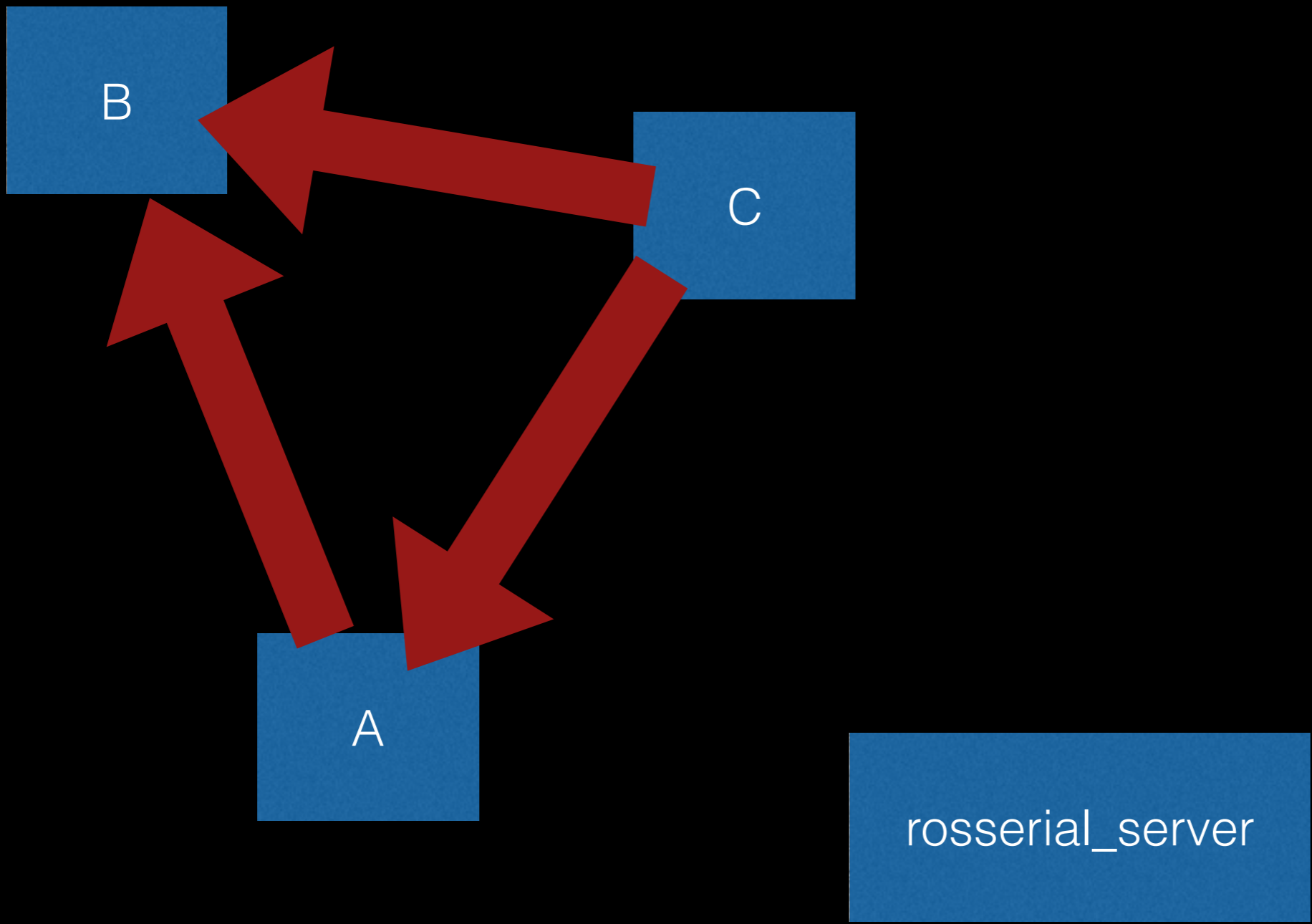


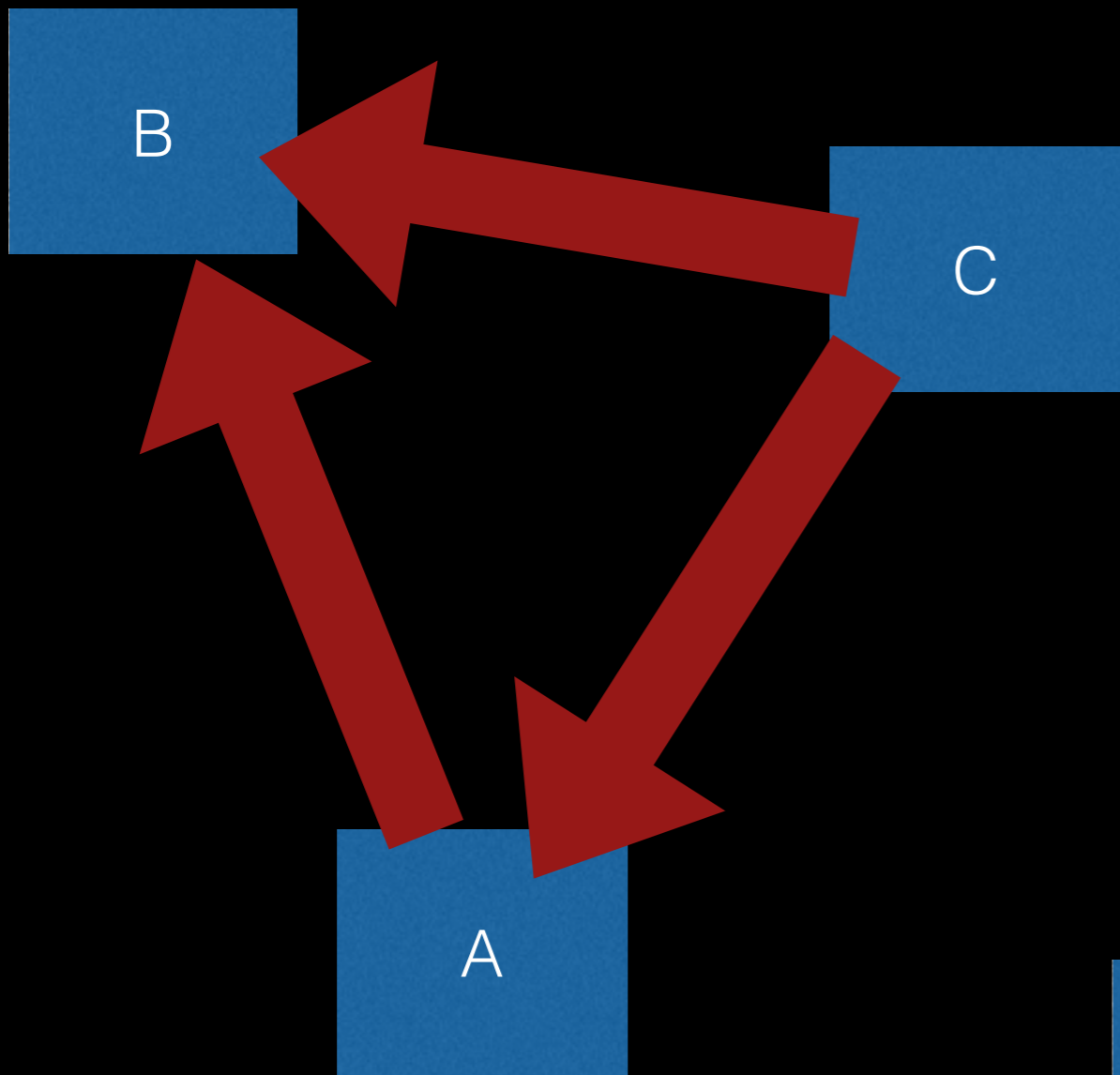


rosserial

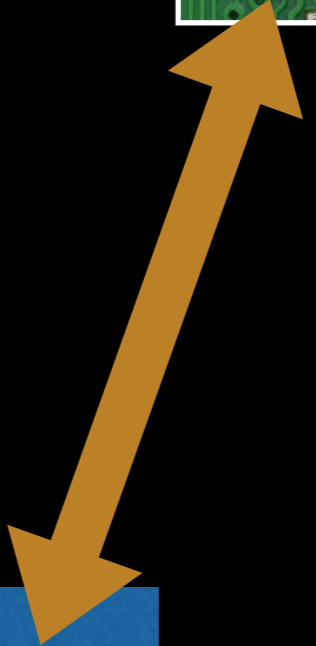


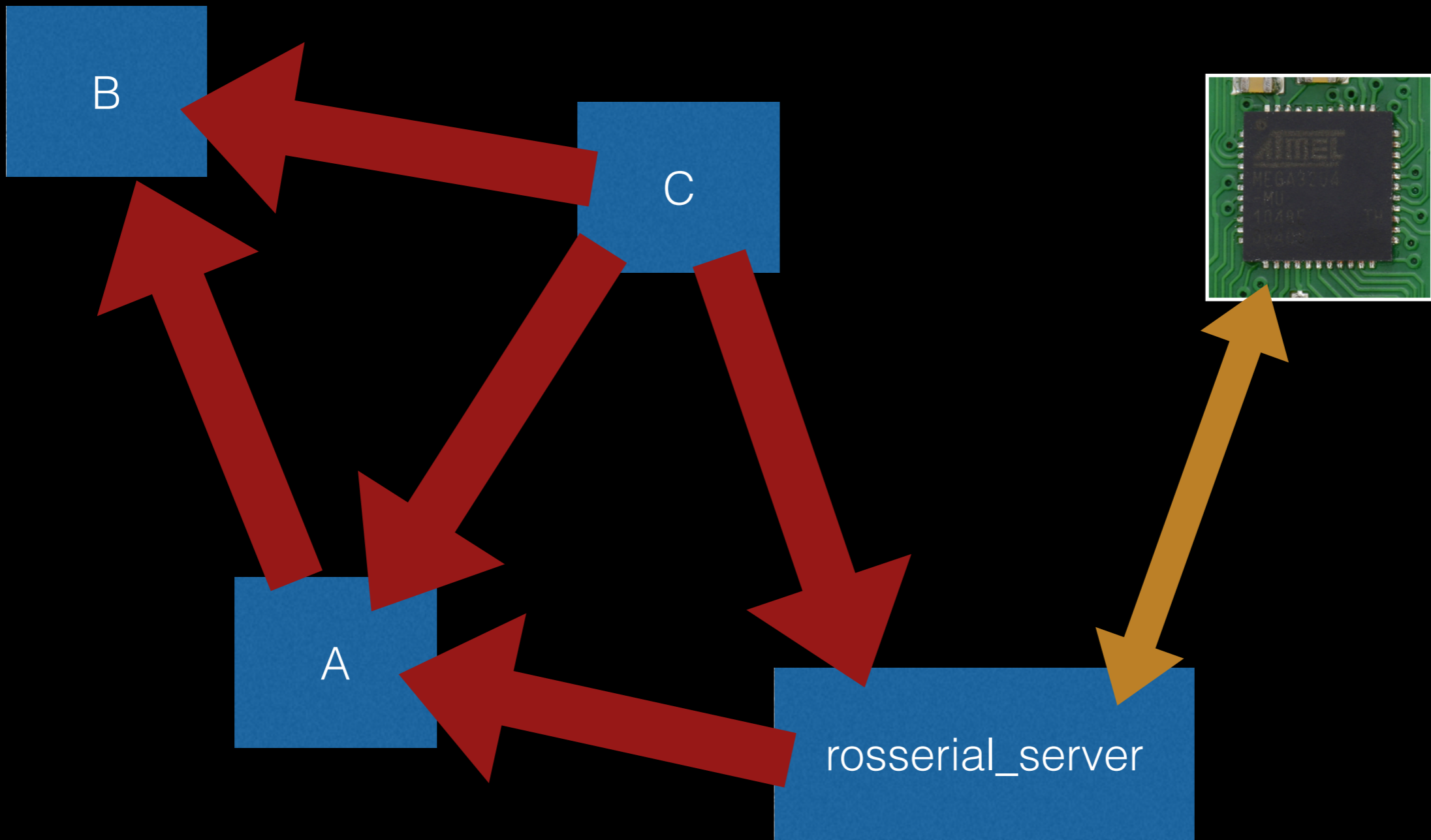






rosserial_server



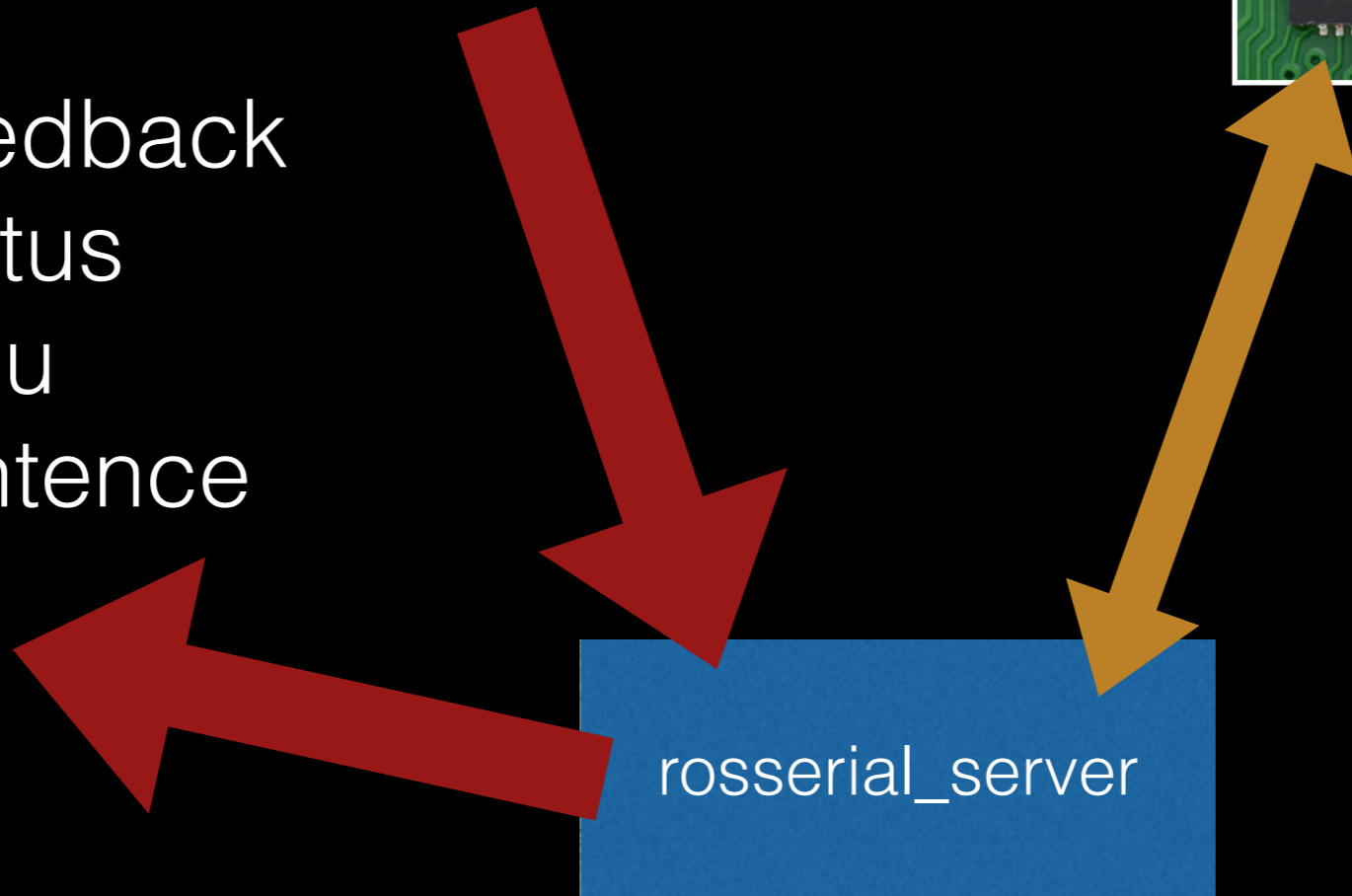


jackal_msgs/Drive

jackal_msgs/Feedback
jackal_msgs/Status
sensor_msgs/Imu
nmea_msgs/Sentence

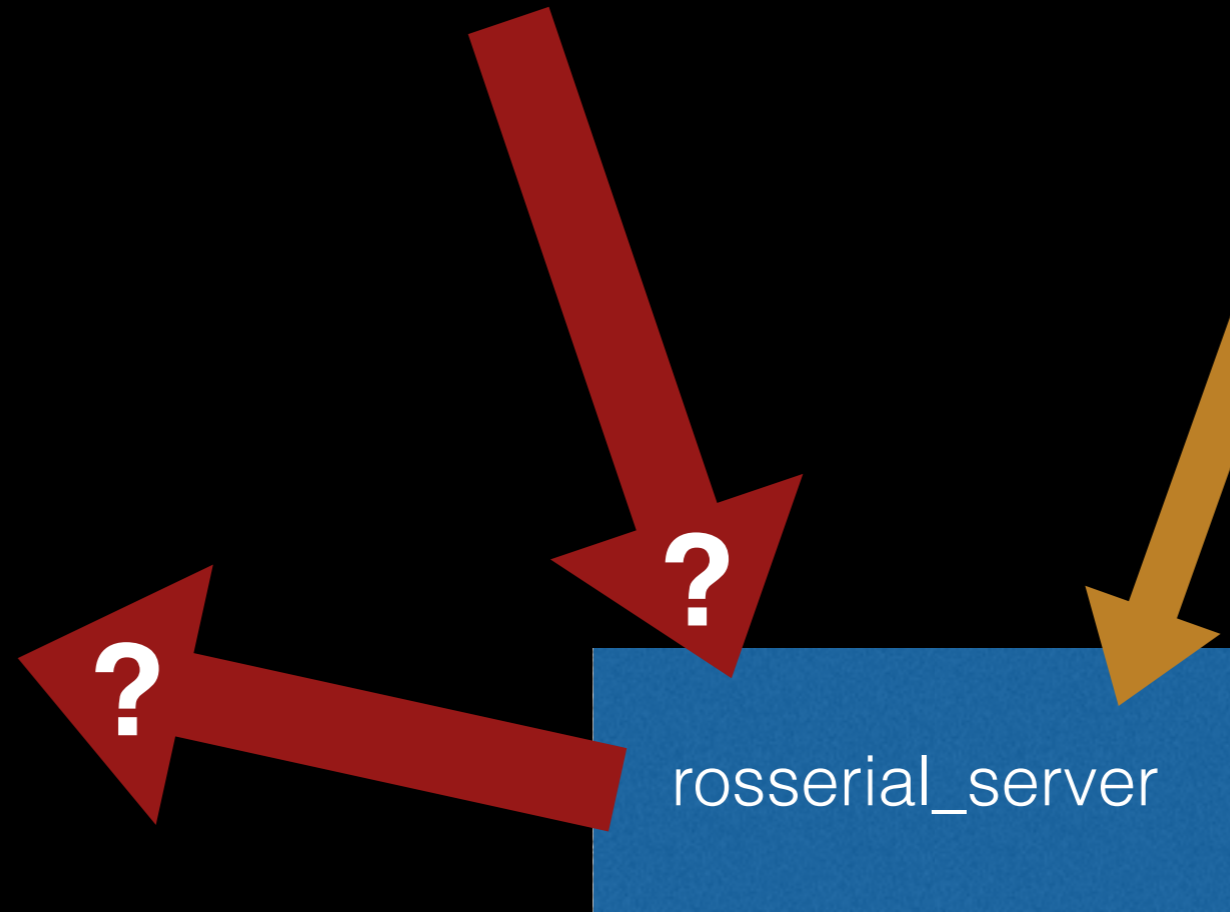


rosserial_server



topic_tools::ShapeShifter

msgtype
MD5



```
sudo apt-get install ros-indigo-rosserial-server
```



```
#include "jackal_msgs/Feedback.h"
```

```
jackal_msgs::Feedback feedback_msg_;  
feedback_msg_.header.frame_id = FRAME_ID;  
feedback_pub_ = new ros::Publisher("feedback", &feedback_msg_);  
nh_.advertise(*feedback_pub_);
```

```
while(true)  
{  
    if (feedback_periodic_.expired(freertos::time()))  
    {  
        control_task_.fillFeedbackMsg(&feedback_msg_);  
        feedback_msg_.header.stamp = nh_.now();  
        feedback_pub_->publish(&feedback_msg_);  
    }  
}
```

jackal_firmware/

CMakeLists.txt



package.xml

test

firmware/

CMakeLists.txt

3rdparty/

include/

src/

```
cmake_minimum_required(VERSION 2.8.3)
project(jackal_firmware)

find_package(catkin REQUIRED COMPONENTS roslint roserial_client ...)

catkin_package()

roserial_generate_ros_lib()
roserial_configure_client(DIRECTORY firmware)

add_dependencies(${PROJECT_NAME}_ros_lib
  jackal_msgs_genpy
  sensor_msgs_genpy
  geometry_msgs_genpy
  nmea_msgs_genpy
)

roserial_add_client_target(firmware mcu.bin ALL)
```



```
catkin_make jackal_firmware_firmware_mcu.bin
```



```
cd j_ws/build/jackal_firmware/firmware
```

```
cmake j_ws/src/jackal_firmware/firmware  
-DROS_LIB_DIR=j_ws/build/jackal_firmware/ros_lib  
-DEXECUTABLE_OUTPUT_PATH=j_ws/devel/share/jackal_firmware
```

```
cmake --build j_ws/build/jackal_firmware/firmware -- mcu.bin
```

```
sudo service jackal stop
```

```
sudo apt-get update
```

```
sudo apt-get install ros-indigo-jackal-firmware
```

```
roslaunch jackal_firmware flash_firmware
```

jackal_firmware/

CMakeLists.txt

package.xml

test



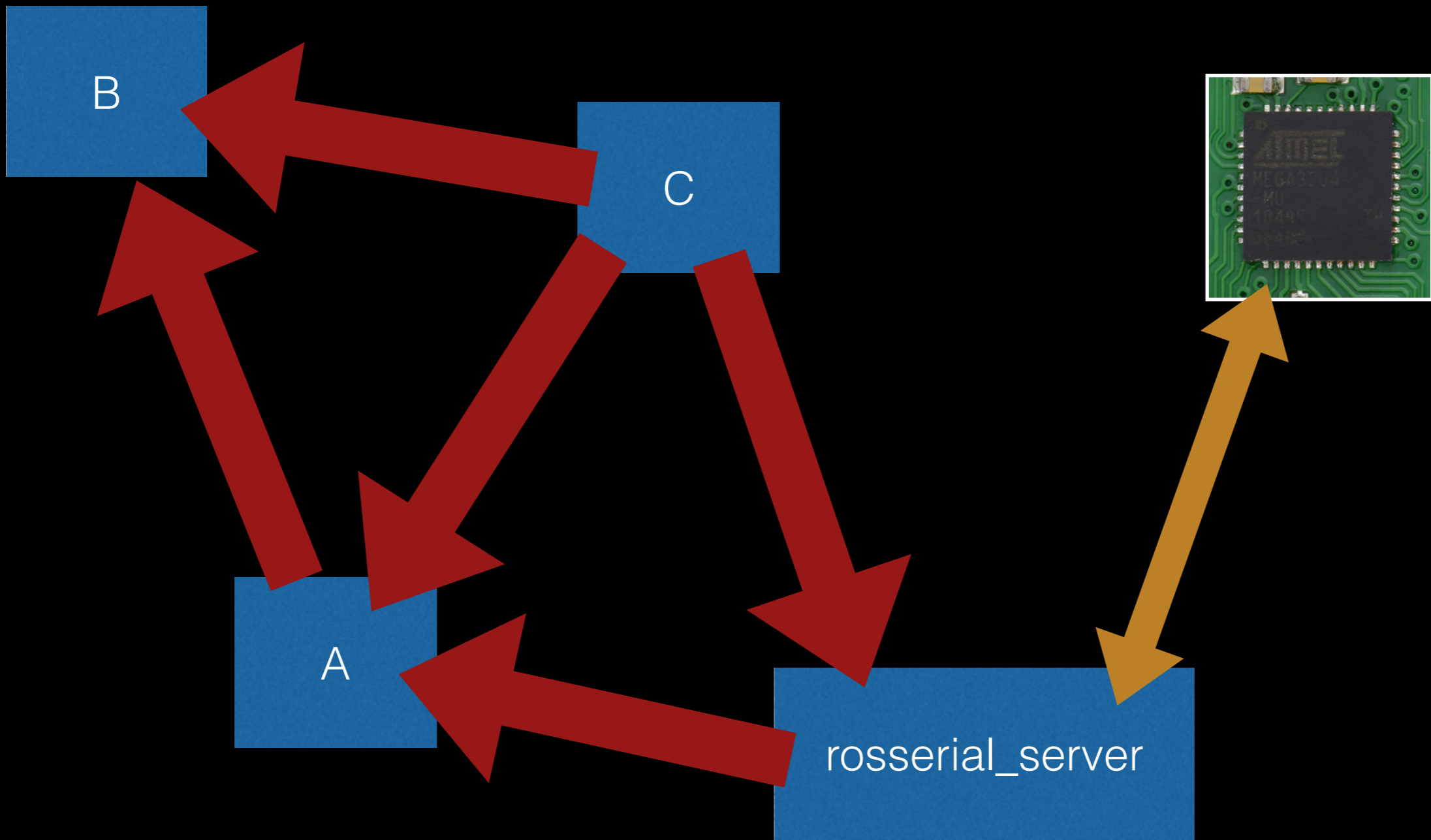
firmware/

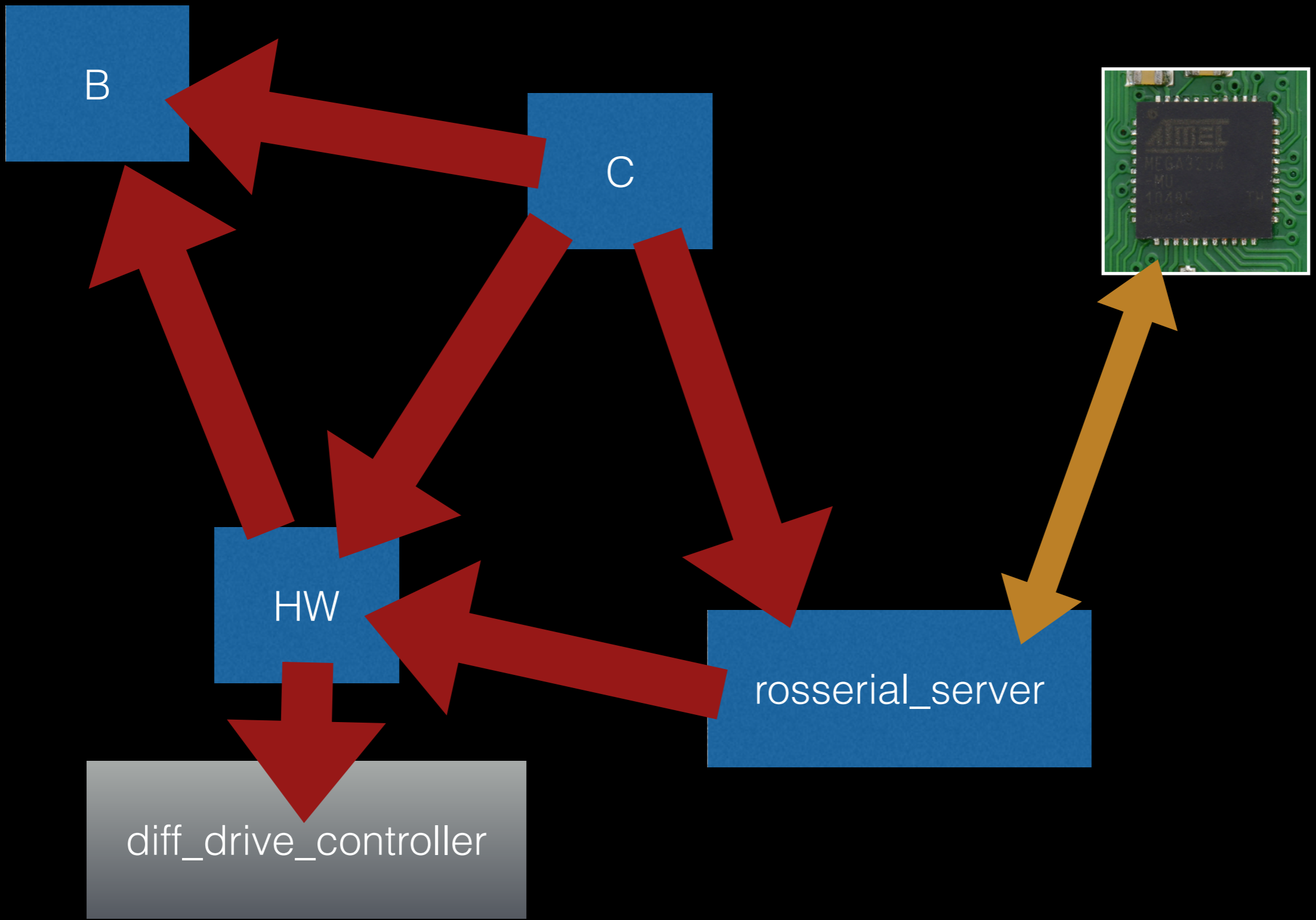
CMakeLists.txt

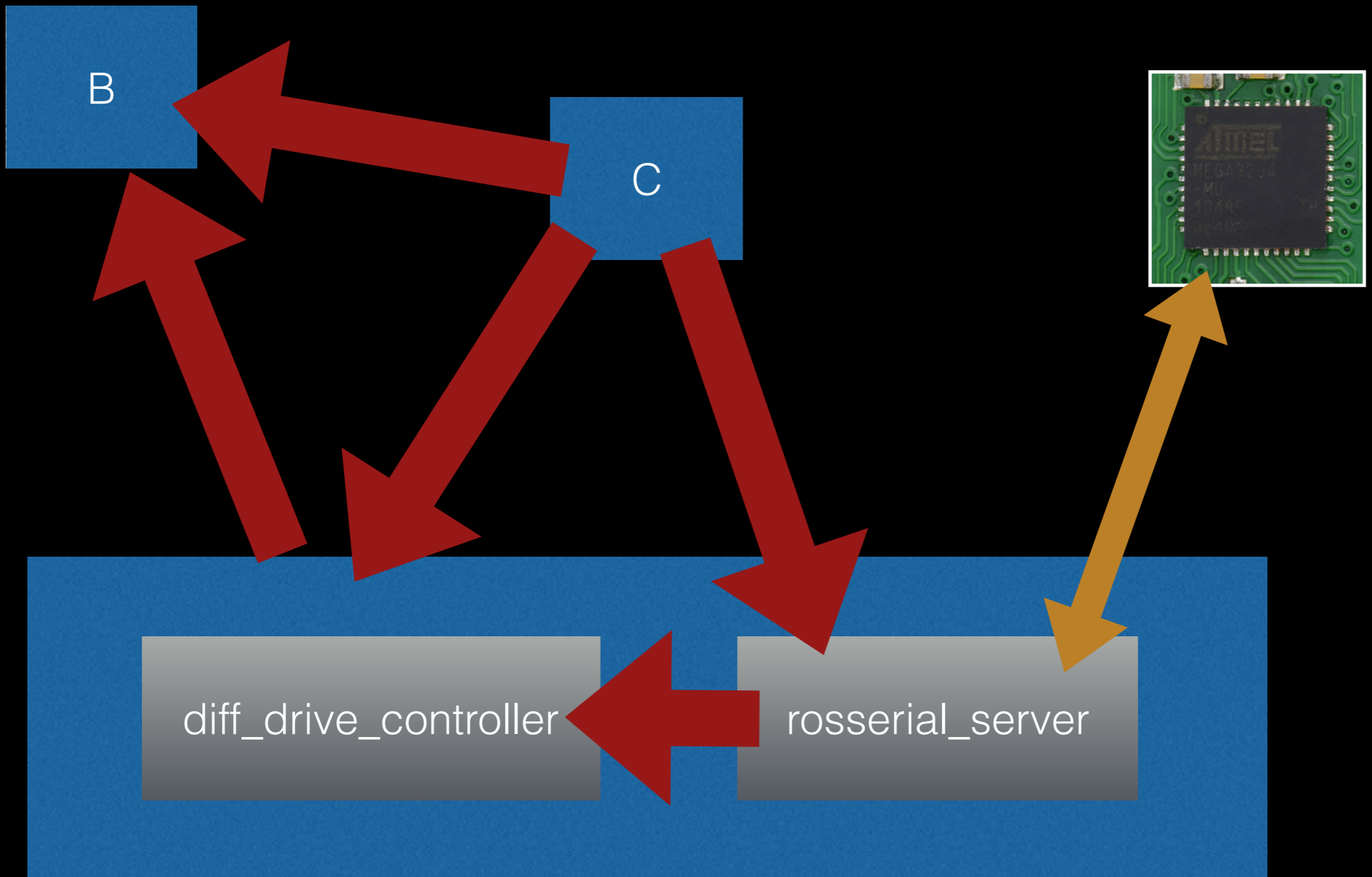
3rdparty/

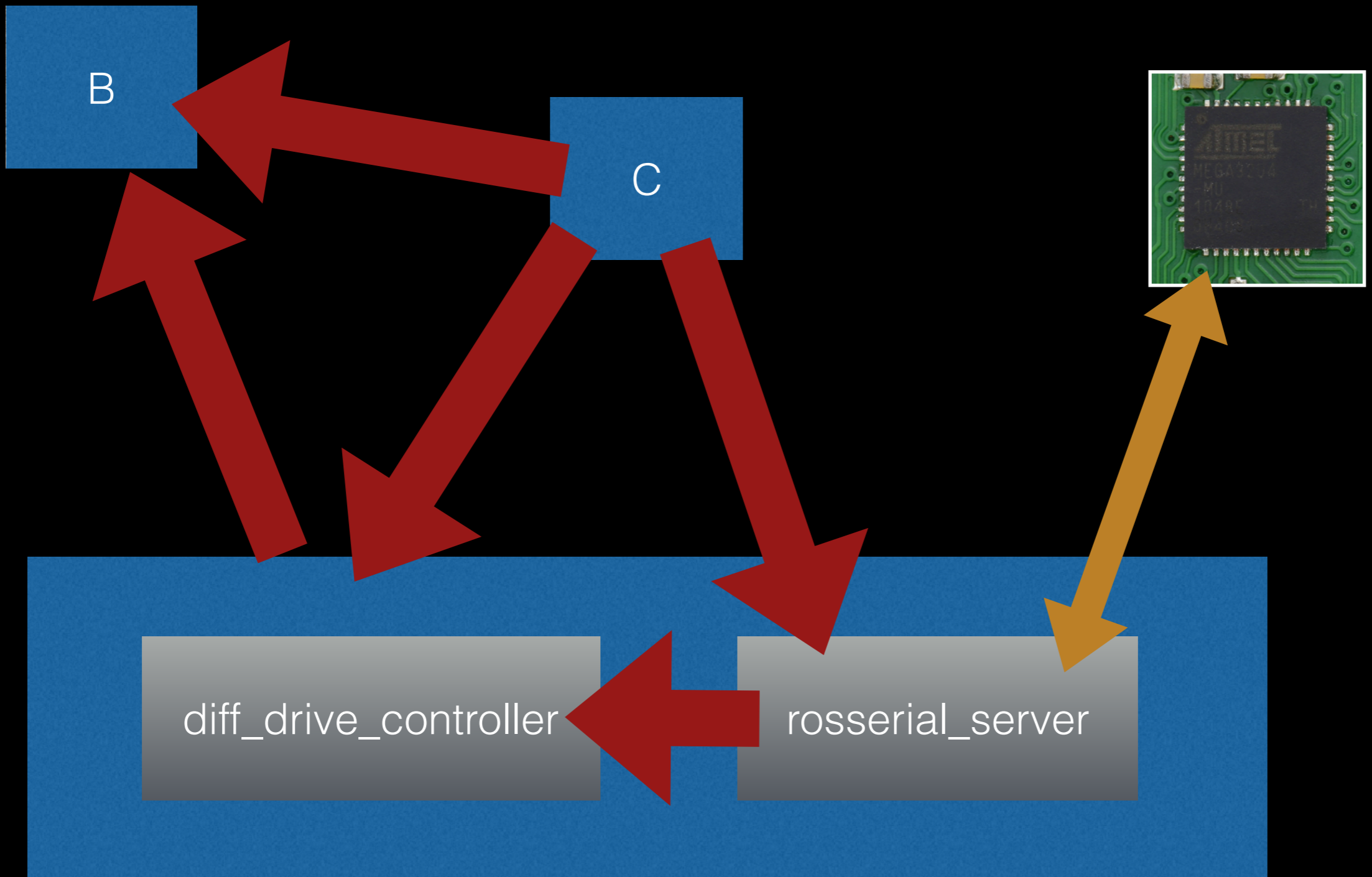
include/

src/









`ros::spin()`



`controller_manager::update()`

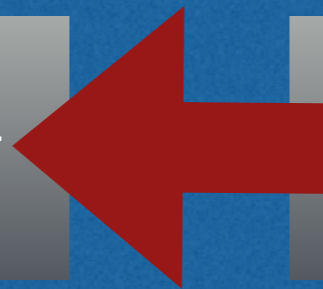


`boost::asio::io_service()`



`diff_drive_controller`

`rosserial_server`



```
new roserial_server::SerialSession(io_service, port, 115200);  
boost::thread(boost::bind(&boost::asio::io_service::run, &io_service));  
  
controller_manager::ControllerManager cm(&jackal, controller_nh);  
boost::thread(boost::bind(controlThread, ros::Rate(50), &jackal, &cm));  
  
ros::spin();
```



diff_drive_controller

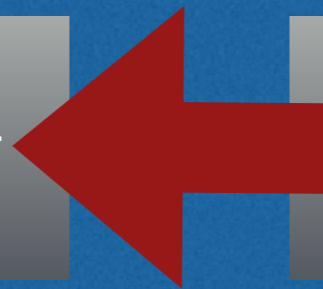
roserial_server

```
void controlThread(ros::Rate rate, JackalRobot* robot,
controller_manager::ControllerManager* cm)
{
    ros::Time last_time;

    while (1)
    {
        ros::Time this_time = ros::Time::now();
        robot->copyJointsFromHardware();
        cm->update(this_time, this_time - last_time);
        robot->publishDriveFromController();
        last_time = this_time;
        rate.sleep();
    }
}
```

diff_drive_controller

rosserial_server



Now using other ros-control controllers!



```
graph LR; A[rosserial_server] --> B[diff_drive_controller]; A --> C[joint_state_controller];
```

diff_drive_controller

joint_state_controller

rosserial_server

DDS

serialization changes

catkin nested builds stay awesome

catkin testing stays awesome

integrating with ros-control stays awesome

PLUGS

`teleop_twist_joy`

`interactive_marker_twist_server`

`roslint`

ROS Manufacturers BoF

