Motion Controlled Robot: Dr. RoomBotNect

Daniel Clifford, Xi Guo, Hasan Jamal, Claire Maki, Joseph Romeo
drroombotnect@umich.edu
EECS 452 - Winter 2011

Kinect Interfacing

PrimeSense's OpenNI (Natural Interface) Module is used in conjunction with the NITE tracking and gesture recognition overlay to locate and isolate a users hand. Once the hand has been recognized with the proper gesture, the tracking of the approximate center of the hand is then transmitted at a rate of 30 coordinates per second. This is accomplished by having the data sent at the same time the video feedback from the Kinect's infrared sensor is refreshing on the screen for the user to view.This center coordinate consists of all three dimension locations, (x, y and z), for future processing and interpretation.

iRobot Interfacing

We initially used a serial/telnet software for interacting with the iRobot. We wrote scripts to describe the behavior of the iRobot. This included controlling the various actuators on the iRobot including its wheels, speakers and LEDs. We are using the RXD serial input pin on the DE-9 port of the robot to transmit serial commands wirelessly to the iRobot. These commands are being received via the Xbee receiver.

Wireless Communication

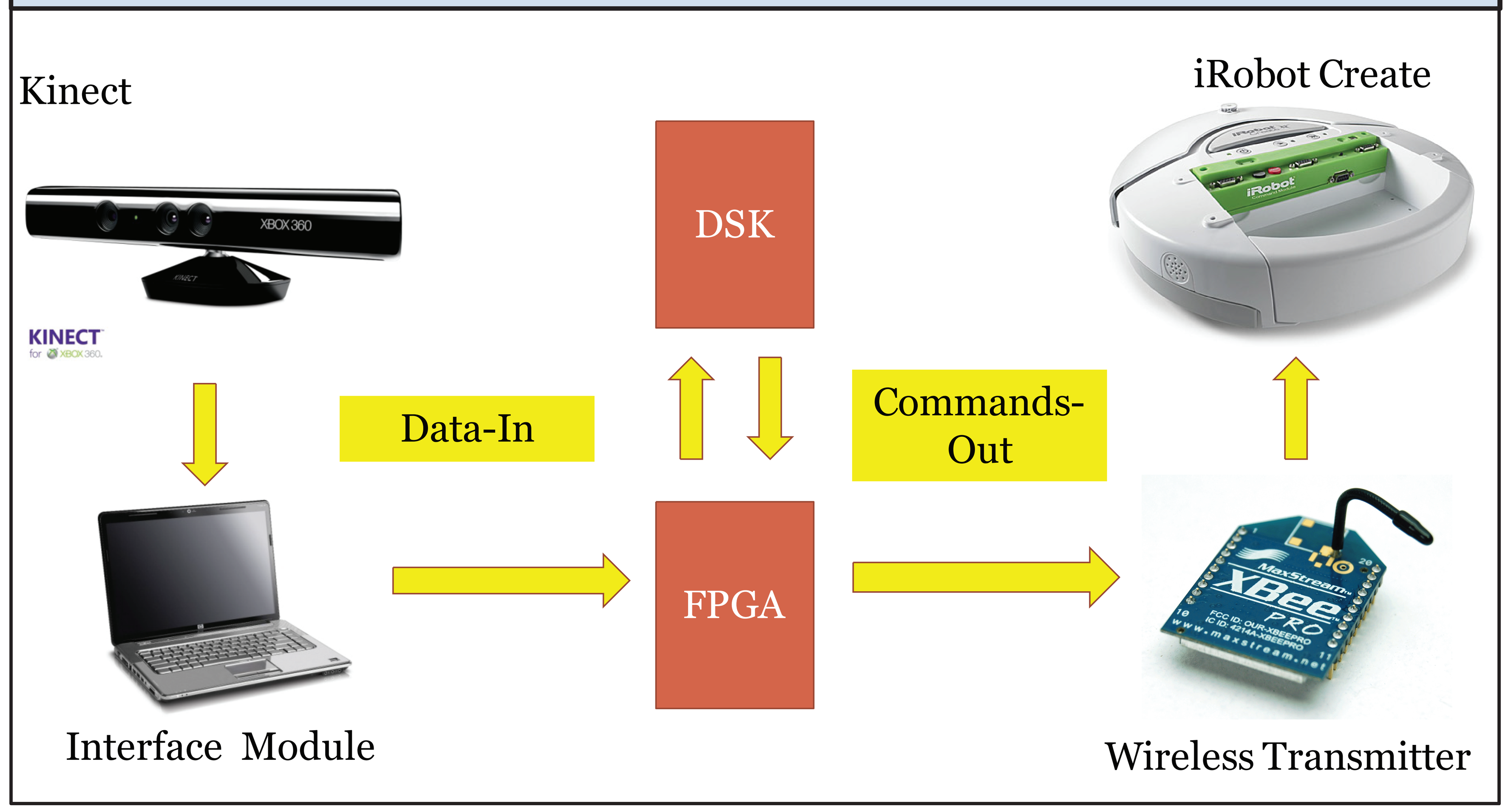
Wireless serial communication to the iRobot is provided by the Xbee transmitter and receiver pair. To establish communications between the transmitter and receiver, the Xbees had to be assigned to the same personal area network. Once set up, the Xbee transmitter sent serial commands output by the FPGA to the Xbee receiver located on the iRobot.

Introduction

There has been a trend in computing to make interfacing to devices more intuitive. One of the most recent innovations in video game technology has been the Microsoft Kinect device which enables control though a natural user interface. It allows users to "be the controller" by tracking the user's motion and using that data to play different games. There are many benefits to motion control. For example, it is much more intuitive for users to move their body in a certain way than learning how to interact with a joystick or other controller. Easier learning translates to a better user experience.

The idea behind our project is to have a programmable robot receive instruction via human body motions instead of an in-hand hardware interface. Using the Microsoft Kinect's abilities to form a 'depth-map', we will use hand motions to control the basic operations of the robot.

Design Flow



Acknowledgements

We would like to acknowledge the invaluable assistance from:
Dr. Kurt Metzger
Dr. Alfred Hero
Benjamin Schwartz

Serial Communication

We receive x, y and z coordinates from the Kinect at a rate of 30 frames per second with one point per frame. The coordinates are then transmitted serially via USB to the FT232 Breakout board which interfaces with the FGPA. The coordinates are transmitted one character at a time. Once received, each character is sent to the DSK where the actual floats are reassembled one character at a time. The DSK then processes these numbers accordingly.

Signal Processing

Data from the Kinect is processed to interpret gestures performed by the user. The commands are defined as follows:

Hand Movement Gestures

Hand Movement	iRobot Movement
Right	Turn right and drive straight
Left	Turn left and drive straight
Push Forward	Drive straight
Circular Motion	Spin based on speed of hand
Return to center	Stop motion
Drop hand, keep stationary	Stop tracking, end session

When the hand is initially detected by PointViewer, the hand tracking software records the first x,y,z coordinate and stores it as a center point. Future points are compared to the center point to determine direction of motion. If the hand stays still, the robot remains stationary. If the hand moves outside that zone to the right or left, it responds with the correct motion.

When the hand leaves the center zone vertically, it is assumed that circular motion is happening. To determine the speed of rotation, 60 points of data are collected to calculate the centroid of the circle. This centroid point is used to convert all future points to polar coordinates. The angle of each point is compared to the previous point to determine speed of rotation. The speed is mapped to three speed levels on the iRobot. After performing circular motion, the user must keep their hand stationary for 2 seconds to re-center the robot before performing linear commands.

To end a user session, the user must drop their hand down by their side and keep it stationary for about 1.5 seconds. The iRobot sings the Michigan Fight Song to signal the end of a session.