ECEN 489: Laser Rangefinder Final Report

Clayton Crawford Jeff Terrell Sam Carey John Boyd
Taahir Ahmed Ashton Jackson Doug Maunder Tengyan Wang

6 December 2013

1 Project Overview

This project attempted to design and construct a low-cost laser rangefinder suitable for use in robotics, safety, and mapping applications. To measure distance to the target, a fast signal (on the order of 100 MHz) is modulated on an infrared laser diode. The light reflected off of the target drives a phototransistor, and the reflected signal is compared to the transmitted signal to determine the phase difference introduced by the laser transmission path. Separately, inertial position tracking is used to track the movement of the emitter through space, to enable mapping of objects or environments. While the device produced during the project period does not work properly, we believe that the concept is sound and could be realized over a longer timeframe.

2 System Breakdown

The overall system breaks down into six modules – three hardware and three software. The *distance measurement* hardware module generates the modulated beam and compares the returned signal to the transmitted signal to perform distance determination. The *beam scanner* uses an angled mirror on a motor with coupled encoder to sweep the beam in a plane and report the current beam angle to the user. The *inertial measurement unit* uses an integrated circuit accelerometer and gyroscope to report linear and angular acceleration to the software pipeline. On the software side, the *pose integrator* takes the reported accelerations and produces the current 3D pose of the device. The *pose transformation* module takes the reported ranges and angles and produces measurement points in the reference frame produced by the pose integrator. Finally, the *visualizer* displays the points using OpenGL. Figure 1 shows the overall information flow between the project modules.

The microcontroller used to coordinate the hardware modules and communicate with the attached computer is an Atmel AT90USB1286, chosen for its high clock rate and built-in USB stack. Direct USB communication has several advantages over the typical USB-serial or serial communication, the most compelling of which is the ability for the device to uniquely identify itself using vendor and device ids that are recognized by a driver on the computer. In addition, the USB specification has an add-on specification (a USB *class*) called the USB Test and Measurement Class, which defines message formats for generic USB-attached measurement equipment; the Linux kernel has a built-in understanding of Test and Measurement devices, allowing the rangefinder to be identified and used with little effort.

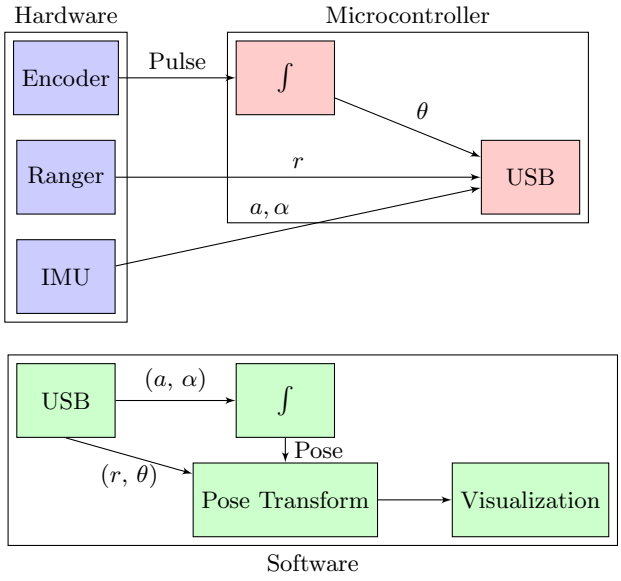

Figure 1: Module representation of the project, with information flow indicated.

2.1 Hardware – Ranger

Jeff Terrell, John Boyd, Taahir Ahmed, Doug Maunder

The heart of the range measurement module is the AD8302 phase/gain measurement chip. On the device, we have a high frequency oscillator that drives an infrared laser diode. This modulated light beam travels out to its target and bounces back to a phototransistor mounted coaxially with the laser diode. The phase difference between the transmitted and received signal is proportional to the path length of the laser, with a small constant offset introduced by the amplification circuitry. The microcontroller takes the phase measurement from the AD8302 and scales it according to its calibration to produce a range. A block schematic of our range detection circuit is shown in Figure 2.

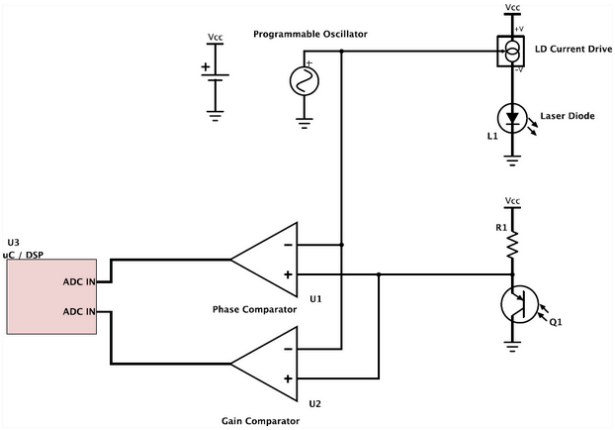

Figure 2: Block diagram of the ranger circuitry. The current drive block is not an integrated circuit; rather, it is a custom drive circuit.

2.2 Hardware – Inertial Measurement Unit

Ashton Jackson, Sam Carey

The IMUs accelerometer records coordinate points in an egocentric frame. The gyroscope reads units in degrees per second, and the accelerometer reads them in g. First, the I2C communication is initialized, and the accelerometer is set to measure mode with a range of +/- 4G and scaled accordingly. The I2C clock speed is set at 50 kHz. As the most recent Eulers angles (roll, pitch, and yaw) are read, they are stored in x, y, and z variables. This occurs in an infinite loop with a delay of 100 microseconds. The microcontroller then transfers the angles from the hardwares egocentric frame to a common world frame.

2.3 Hardware – Beam Director

Sam Carey, Doug Maunder

The laser diode and phototransistor are installed parallel to each other on the mainboard, pointing forward. They look into a two-inch mirror angled at 45 degrees and mounted on the tip of a motor shaft pointing back coaxially at the mainboard. A 3D-printed frame holds the motor in this position by a thin armature. The 3D-printer was also used to mount the mirror to the motor shaft. As the motor shaft and mirror rotate, the laser beams reflection is redirected at a constant angular frequency, circularly scanning a plane perpendicular to the shaft and the fixed beam from the mainboard. As the laser sweeps across various surfaces in the environment, its backscatter is detected through the mirror by the phototransistor. A small slice of the plane is obscured from the lasers scan by the motors supporting armature. Taking advantage of this, the blocking effect triggers a reset in the beams present angle counter, thereby calibrating the starting angle relative to the mainboard. Intermediate angles, between armature scans, are tracked by the motors built-in encoder.

2.4 Software – Pose Integration

Taahir Ahmed

The pose integrator takes linear and angular acceleration data from the IMU and produces a 3D pose for use by the pose transformation unit. Each integration step is modeled as an instantaneous rotation due to the angular acceleration, followed by a linear offset from the linear acceleration. This is a common spatial integration strategy that produces accurate results provided the underlying integration scheme is accurate and updates are closely spaced. There are two separate integration schemes used. The angular integration is handled by a Lie group integration scheme in which angular acceleration, velocity, and position are modeled as unit quaternions. A Lie group integration scheme (as opposed to simple Euler or even Runge-Kutta integration) respects the unit-length constraint on the quaternions, giving a very accurate integration. The linear integration is handled by a standard runge-Kutta scheme using batched updates.

After integration, the pose consists of a unit quaternion (orientation) and a 3-vector (position). This representation is converted to a 4-by-4 affine transform matrix for use in the pose transformation stage.

2.5 Software – Pose Transformation

Clayton Crawford, Tengyan Wang

This module takes the current pose from the pose integrator and uses it to transform measurements from the rangefinder and beam director from the hardware's egocentric reference frame to a common world frame for display. The overall world-frame result is given by

$$p_w = ABp_e, \tag{1}$$

where A is the affine transform matrix produced by the pose integrator, B is the affine transform matrix that represents the angle recorded by the beam director, and p_e is the egocentric range vector $[r\ 0\ 0\ 1]^T$. In order to construct the matrices and perform matrix operations, the Armadillo linear algebra library is used. The JSONcpp library is used to send the resulting data, after matrix multiplication, to the visualization stage.

2.6 Software – Visualization

Tengyan Wang, Clayton Crawford

The visualization receives the data set after pose transformation and then uses these data for OpenGL display. Original data are sets of 3D point with x,y,z variables collected by IMU, pose transformation transforms the data from their local coordinate systems to a global coordinate system. Since all the points are in the same coordinate system, they can be directly displayed using OpenGL functions. We are using freeglut3 library and include 'glut.h' in our visualization program. The typical OpenGL display uses the form glutDisplayFunc(function) in the main function, the 'function' within parentheses is used for plotting 3D points in the display window. Besides display function, other functions like rotation, zoom in/zoom out viewing and change viewing point are applied, all of these functions involves real-time 3D motion.

The test part of visualization is based on random or manipulated data with Json output and input. The input data are generated by a test program, which can write random/manipulated data in the form of Json object to a certain file, acting like it's the data received from pose transformation. Then the visualization program also creates the same form of Json object and read Json from input file, at last displays 3D points with x,y,z variables. An example simulated scan of a pipe with a deformity is shown in Figure 3.

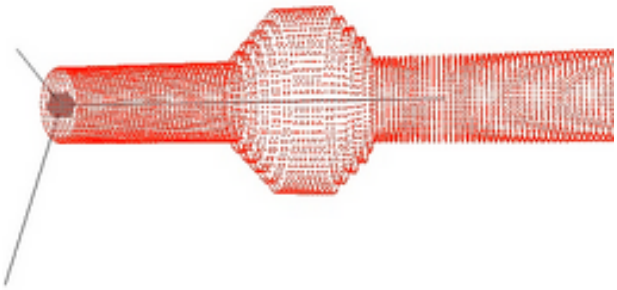

Figure 3: Output of pose transformation and visualization pipeline for simulated data of a pipe with a circular deformity.

3 System Status

At the end of the project period, the rangefinder system is incomplete. The software pipeline is completed and tested (with the exception of the pose integrator), as well as the IMU and beam director. The primary defect is in the rangefinder unit; specifically, the modulation driver circuitry. The current design uses an op-amp to directly drive the laser diode. However, the slew rate required in this configuration is beyond the capability of any reasonably-priced op-amp. To fix this, the circuit shown in Figure 4 is proposed. It uses the op-amp in an inverting configuration to drive the gate of a MOSFET. The MOSFET then drives the laser diode in a current-drive configuration. Instead of directly trimming to its output, the op-amp trims to the voltage across a small current-sense resistor, which ensures that the current through (and the power emitted by) the laser diode follows the modulation function well.

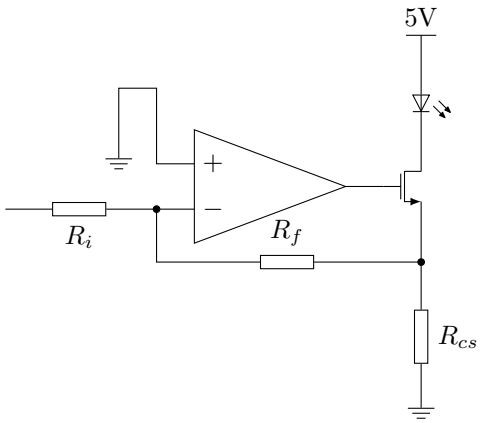

Figure 4: Proposed current-sense driver for modulating laser power. Drive amplitude is not limited by op-amp slew rate.