

All the code and demo videos are in [EE292S_Lab2](#)

code files guide (the number at the end of the file name is the number of bits in the prbs sequence):	
code	description
plot_cross_corrs_5.py plot_cross_corrs_8.py	Live plot of 7 cross correlations (one for each sense line), each with up to 5 drive peaks. The correlation is done with our PRBS, and each drive line outputting an offset version of this PRBS.
part2_live_5.py part2_live_8.py	Live plot of the 5x7 heatmap of our touch sensor. These 35 points represent the 7 cross correlations (one for each sense line), each with up to 5 drive peaks, giving us 35 spatial points.
part2_fit_5.py part2_fit_8.py	Live plot of the 5x7 heatmap of our touch sensor, augmented with a centroid marking and a gaussian ellipse fit.
part2_get.py	part2_fit_8.py, modified to save up to 1000 centroid samples.
part2_variance.py	Postprocessor for part2_get.py, calculating the variance in x-coordinate, y-coordinate, and x-y position of the centroid from a given set of samples.
part2_gesture.py	Live plot of the centroid location and velocity, using Kalman filtering.
demo videos guide (the number at the end of the file name is the number of bits in the prbs sequence):	
demo	description
prbs5_correlations.mov prbs8_correlations.mov	Live plot of 7 cross correlations (one for each sense line), each with up to 5 drive peaks. The correlation is done with our PRBS, and each drive line outputting an offset version of this PRBS.
prbs5_heatmap_ellipse_fit.mov prbs8_heatmap_ellipse_fit.mov	Live plot of the 5x7 heatmap of our touch sensor, augmented with a centroid marking and a gaussian ellipse fit. The 35 points on the heatmap represent the 7 cross correlations (one for each sense line), each with up to 5 drive peaks, giving us 35 spatial points.
gestures.mp4	Live plot of the centroid location and velocity, using Kalman filtering. The dot shows the centroid. The arrow shows the speed.