# Introduction

This report describes the design and implementation of an Inertial Measurement Unit (IMU).

# Design

The designed was done after specifications to follow the specifications from the Xsens Xti-10 series as close as possible.

It was decided to use the Atmel AT90CAN128 microcontroller as MCU, because of the students familiarity of it from earlier work. The gyroscopes chosen were Analog Devices ADXRS450, which gives a digital output via SPI communication. The chosen accelerometers were Analog Devices ADXL325, which gives an analog output.

For the communication between the IMU and the rest of the world it was chosen to use serial via USB, using FTDIs UART-to-USB converting chip FT232RL.

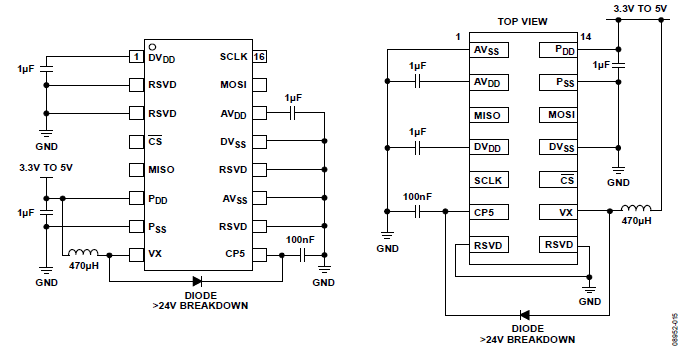
The power supply to the IMU can be switched between power from the USB and external power. Both of these are filtered and regulated to 3.3 V. Meaning that the entire card is driven at 3.3V (except for the FT232RL which takes 5V directly from the USB and 3.3V to drive its IO to the microcontroller).

For the programming interface to the microcontroller both the ISP and the JTAG pins were put on the card as pin headers, for further work it could also be possible to write a bootloader and thereafter program the MCU via USB.

## In depth: Gyroscope

Since the gyroscopes measures one axis per chip, three chips are needed. This model comes in two packages, one lying down, SOIC\_CAV, (to measure rotation around the z-axis) and one standing up, LCC\_V, (to measure rotation around either the x- or y-axis depending on orientation).

According to the specifications in the datasheet they need a few external components to work properly, these are shown in Figure 1.



Figur Application circuits