Titan Electronics Projects

Titan is a large project, with several subsystems for electronics.

# Vision System – Critical

This is critical. There is a need for 3 displays, one primary view for each rider with a backup monitor for the forward-facing rider that should be entirely independent of other electronic systems to minimize complexity.

The system for these revolves around Raspberry Pis (rpi), a Raspberry Pi Camera is used as the video input to the Raspberry Pi. It is then displayed over HDMI to a monitor to be displayed, the primary monitors for each rider have an overlay applied with important information to each rider such as speed.

There is an option of removing the rpi from the backup system to simplify it, this would require a camera that directly outputs video in one of the formats on the

Tasks for this are:

* Writing code for the display
  + Heavily software/programming oriented
  + Write code that will display
* Design hardware for the display
  + Hardware intensive
  + Monitors need

Skills needed/learned:

* Configuring rpis
* Python / C++ coding
* Basic circuitry
* Potential for circuit board design

# Sensor System – Important

This is an auxiliary system to the rpis, probably based around an Arduino of some kind (likely an Nano). The purpose of this system is to collect real-time sensor data for the vision system to query when needed. The reason for moving this off the rpis is to ensure data is collected in real-time by not having the sensor system multitask and easier hardware interfacing due to the nature and open support of Arduinos for these purposes. The data needed in a rough order of descending importance is:

* Wheel speed
* Ambient air pressure
* GPS
* Accelerometer
* Wheel temperature
* Brake temperature
* Wheel pressure
* Ambient air temperature / humidity

Skills needed/learned:

* Micro controller (C/C++) implementation and coding
* Breadboard prototyping
* Circuit board design

# Circuit board design

Pretty self explanatory, design a circuit to connect all the hardware in the bike. Goes pretty hand in hand with the sensor system, however there is a need for other systems to be present on this board, and different versions of the board to be made for the different rider positions so it is considered it’s own task. The systems that will need to be implemented on boards:

* 5V regulator (for each battery used)
  + Needs to output atleast 2.5A per rpi fed
* Sensor system
* Communication circuits
  + Between rpis (if needed)
  + Telemetry (if researched)
* Emergency signal
  + Likely an LED in the tail of the bike that is turned on when help is needed. Should be a super simple ciruit

Note, last year we used a single board that had all the features we needed. Although this made assembly easier and repairs/substitutions easier, it was very inconvenient due to only a fraction of the board being used on the ‘slave’ systems. Going forward we should put effort into making boards designed specifically for each rpi system.

Skills needed/learned:

* Circuit design
* Using PCB design software
* Circuit assembly
* Circuit debugging

# ANT+ System – Moderate

This system is meant to find a way to get data from the off-the-shelf systems used to monitor rider data: the heartrate monitor and power pedals. The data needed is power and cadence, heart rate would be a nice addition. These devices are commonly implemented with ANT+ communication, we should get an ANT+ receiver and start experimenting with it.

Skills needed/learned:

* Coding (python/C++)

# Raspberry Pi Communication – Normal

This covers communication between the two rpis for each rider. The purpose is to allow basic information to be passed between riders’ rpi’s such as cadence or power.

This will likely make use of an ethernet cord connecting the rpis in a ‘crossover’ configuration. This will be a coding-oriented project.

Skills needed/learned:

* Coding
* Potential hardware/circuit design testing

# Logging – Low

Implement some logging system of information on the data collected by the bike systems. This is split into two parts, video recordings and data records.

This is entirely coding oriented.

Skills needed/learned:

* Coding on rpi (python/C++)

# Telemetry – Low

Send numerical data over long distances (>100m) from within the vehicle to another system, likely a chase vehicle, e.g. speed. This will need to be able to work wirelessly and ideally from any direction relative to the vehicle. The baseline required bandwidth is 10 kBd (kilobaud) at 100m with clear line of sight between transceivers. Reliability is nice but not critical if the connection can be recovered quickly.

More of a research project but implementing it will probably need some help.

Skills needed/learned:

* Coding on rpi
* Coding on microcontrollers
* Others TBD depending on telemetry system selected

# Voice communication – Minimal

Develop a system to broadcast audio between our electronic system. The focus of this will be improving the bandwidth of our telemetry system and developing sound recording/replication systems at each end.

Independent research project in reality.

Skills needed/learned:

* TBD based on direction taken