Systems Design Exercise

Group 3:
Saluki Hadahiro
Kim-Long Vo
Grace Livingston
Olivia Smith

June 16, 2018

Contents

1 Hardware Requirements **1**

2 Configuration and Failsafe Operation **2**

3 Fault Detection **2**
3.1 Controller Faults 2

4 Motor Control **2**
4.1 PWM Controller 2
4.2 Rotational Velocity 2
4.3 Temperature 3
4.4 Current Draw 3
4.5 Self-Test 4

5 Control Interfaces **4**
5.1 Local Interface 4
5.2 Remote Interface 4

6 Emergency Brake **5**

1 Hardware Requirements

In order to provide a platform that can manage the operation of the drive train and the ongoing operator interfaces and still have enough of a safety margin, a SoC with a PWM controller, 4 ADCs, 4 SPI interfaces, an ethernet interface, and several GPIOs should be available on the system. These are common on many platforms.

The CPU doesn't need to be exceptionally fast. The time for each revoloution of a 3600 RPM motor is 1.6 ms. With 12 sensors, even with a round robin scheduler, the microcontroller would have 133 s to service each sensor's data assuming each sensor needed servicing on each revolution. But even with this rotational rate, the actual driven system is likely going to move much slower, and if there's a hard response time of 100 ms from a fault detection to the application of the emergency brake, at 8MHz that's still 800,000 clock cycles, which is far more than most CPUs require to service an interrupt and change the state of a GPIO.

If we include the overhead of running the management software, and a small RTOS on the system, a 16MHz RISC system with 512MB of RAM would be more than enough to run everything. 4GB of flash storage for firmware, failsafe copies, data collection, and configuration would be enough to continue operation under most expected conditions while still providing reliable control and management with or without remote software.

2 Configuration and Failsafe Operation

The system's configuration should be validated on load and on change to prevent out-of-range operation from causing damage or harm. In order to keep the configuration as a whole adequate, configuration changes should be validated and then made to the entire configuration at once only after validation succeeds. Invalid configurations should maintain the last good configuration, or on startup should use a known-safe configuration.

3 Fault Detection

3.1 Controller Faults

The system should use ECC for RAM and flash. This will ensure that hardware data corruption is detected, and where possible, corrected. Statistics on failures should be maintained as part of the standard operation and management modes.

4 Motor Control

4.1 PWM Controller

The system is required to manage the motor speed and direction. To control both, a controller with a PWM peripheral connected to an H-Bridge will enable the system to provide positive control over both. The signals provided to this peripheral can then be used to compute the expected velocity, current draw, and temperature of the various drive train components given the environmental temperature. This can then be compared to the actual measurements to identify possible faults and to isolate them.

4.2 Rotational Velocity

To measure rotational velocity, a rotational Hall Effect Sensor on the motor shaft can be used to determine both. When the speed or direction doesn't match those expected, the controller can signal a motor fault and stop operation.

These sensors normally communicate on an I^2C bus. These are very common on standard SoCs and should be standard on most controllers considered for this application. Ideally the output of the transmission would also have a corresponding sensor allowing the controller to compare the input and output velocities to catch drive train faults.

Since they also have no moving parts and no sensors that can be occulded by dust, the maintenance requirements are exceptionally low compared to other options. They also require less power and calibration than other options, and device faults are much easier to detect when they happen.

4.3 Temperature

The controller also is required to monitor the motor temperature. This can be done with a series of thermocouples. Each would monitor the temperature of a different part of the drive system. The motor housing itself, depending on its size, should have at least two, one for the drive spindle end and one on the opposite side of the housing. The spindle and transmission should also each have at least one to monitor the condition of the drive train.

For these, the thermocouples can be monitored with one or more ADCs. To save on peripheral input requirements, multiple thermocouples can be multiplexed together, and a set of GPIO pins can then be used to select between them. The multiplexor will have an effect on the ADC readings, but at the same time, the accuracy of these readings doesn't need to be high precision so much as the long-term trends and the low/nominal/high temperature bands would need to be identifiable.

Having one or more measuring the ambient temperature would also provide a stronger ongoing calibration signal for the others. This will allow compensating for both drift over time and more ephemeral environmental changes, and also provide more information on the presence of sensor faults.

4.4 Current Draw

Another operational parameter to measure is the motor current draw. The current draw can be monitored with another Hall Effect sensor, and read via an ADC channel or over an I^2C bus, depending on the kind of current sensor. The higher the resolution of data available, the easier it would be to identify test signals added to the PWM output that wouldn't affect the motor velocity but would still be detected by the current probe.

On its own the current can provide much of the information the other sensors would also provide, such as heavy load, no load, short circuits, and so on. When combined with the velocity and thermal information, the current draw

can provide much more accurate data on the system status, and when there is a current fault the other sensors can then help isolate where the fault is and what kind of fault it is.

For example, if the drive train is disconnected by damage or by other maintenance, the motor current curve will be much different, but will still correlate similarly to the velocity curve. The thermal data will not show as much of an increase as normal, however, and the peak current will also be much lower, and this would then isolate the fault to this specific category. Adding in the drive train velocity and temperature data would then show no corresponding motion on the output of the drive train, or would even possibly show no data from the drive train sensors if they're disconnected automatically with the drive train itself, and would prevent a potentially dangerous fault where the motor starts while exposed to people working on it in the absence of other housing fault sensors.

4.5 Self-Test

All of these sensors introduce several opportunities for faults unrelated to the actual drive train operation. However, by coordinating the inputs, and by performing test readings from each on start-up and while under operation, sensor faults can then also be identified. For example, even if the motor is locked, sending a short series of high-frequency pulses out of the PWM should be reflected in the current monitor convolved with the inductance of the motor's coils, showing as an average of the pulse duty cycle. If there's a wiring fault in the motor, this will change or prevent these pulses from propagating through to the current sensors.

5 Control Interfaces

5.1 Local Interface

For the local interface, we can use a capacitive touch screen with programmable buttons. These are reliable, work in a range of enironments, and are inexpensive because of the proliferation of smart phones. They usually interface with UART for the programmable buttons, and over a memory-mapped or SPI interface for the display itself.

The programmable buttons provide a good interface for things like emergency stop controls, immediate configuration changes, and other user inputs that can't be lost and that hardware buttons could lose over time from wear and from damage.

The display also can provide context-sensitive information to the user, like the current operational status, any warnings from the self-test or other sensors, and especially any faults detected. This information can be provided without needing users to navigate to it, since the controller would know exactly what's most urgent, or can otherwise display the current system state. It also can

provide graphical depictions of the location of the fault to users on the system itself to aid in troubleshooting and repair.

5.2 Remote Interface

Using an Ethernet interface would allow the system to integrate with common control protocols such as SNMP, and present a control interface using a web browser. It can also be configured to send messages via e-mail. For more advanced networks it can even integrate with the local authentication system.

6 Emergency Brake

The system should have a brake that's engaged when there's no power, or when an emergency stop button is engaged. This stop button, as well as other hardware failsafe devices, should engage the brake and notify the controller via an interrupt that it's been engaged. The controller should also be able to engage this brake as part of normal operation or in the event of an emergency condition detected by the various sensors.