Fault tolerant Steer-By-Wire road wheel control system

Bing Zheng, Cliff Altemare, and Sohel Anwar

Abstract— **This paper describes a fault tolerant steer-by-wire road wheel control system. With dual motor, dual microcontroller architecture this system has the capability to tolerate single point failure without degrading the control system performance. The arbitration bus, mechanical arrangement of motors, and the developed control algorithm allow the system to reconfigure itself automatically in the event of a single point fault, and assure a smooth reconfiguration procedure. The experimental result illustrates the effectiveness of the proposed system.**

I. Introduction

WITH electronically controlled steering systems, there is a great potential for enhancing safety due to fast and precise intervention that is in contrast to the limited reaction time of the driver[1]. At the same time, Active steering provides the potential for the improvements of driver's comfort, handling, and robustness of the driver-vehicle system with respect to uncertain operating conditions [2] - [4].

One technology under investigation for electronically controlled steering systems is steer-by-wire [5], [6]. Since there is no mechanical connection between steering wheel and road wheel for steer-by-wire, active steering can be easily implemented by inserting an electronic control system between driver's input from steer wheel and road wheel drive system. This electronic control system employing sensors, communication channels and microcontrollers ties the steering wheel and road wheels to each other. But due to the fact that there is no mechanical linkage between the steering wheel and the road wheels for steer-by-wire system, a fault from a sensor, actuator or microcontroller that form the control system may result in unwanted steering effects, if not handled quickly in a fault-tolerant manner. Thus, a fault tolerant control system is safety critical in steer-by-wire vehicles, requiring highly dependable sensors and actuators, quick fault detection and identification algorithms and a means for maintaining reliable vehicle control in the event of a fault.

To overcome the above-mentioned potential fail-safe issue, a dual motor, dual microcontroller control system architecture for steer-by-wire road wheel operation is proposed in this paper. With this architecture, each motor uses a smart motor controller (SMC) to form an inner motor torque control loop. This torque loop controls the motor torque output to track a given torque reference. The microcontroller forms an outer loop, performing road wheel position control. By using two microcontrollers, a master-slave structure is formed. The master carries out all the steer-by-wire road wheel control operation while slave provides the system redundancy. In the event that a fault occurs within the master controller, slave takes over the control operation. If a fault occurs at one local motor loop, the corresponding SMC will shut down the torque control loop. In this case, the steer-by-wire control system switches to single motor operation automatically without any intervention from master or slave microcontrollers.

This paper is organized as follows. Section 2 discusses the system architecture. Section 3 describes the system dynamic and control system design. Control system implementation is outlined in Section 4. Section 5 illustrates the test results, and the summary of the paper is contained in section 6.

.

II. SYSTEM ARCHTECUTRE

The overall system architecture is shown in Fig. 1. The actuator assembly uses two brushless motors, M1, and M2. The shafts of both motors are connected to a worm gear, which drives a dual-pinion rack steering mechanism. The worm gear is connected with pinion 1, while pinion 2 is used as mount for an absolute angular sensor. The motor has a built-in resolver for motor commutation purposes. Additionally, this resolver is used for relative pinion angle measurement. Each motor has its own SMC that carries out the local closed-loop torque control and motor fault detection.

Manuscript received September 9, 2004. This work was supported by the Chassis Advanced Technology Department of Visteon Corporation.

B. Zheng is with the Chassis Advanced Technology Department, Visteon Corp., Dearborn, MI 48126 USA (e-mail: bzheng@visteon.com).

C. Altemare was with the Chassis Advanced Technology Department, Visteon Corp., Dearborn, MI 48126 USA. He is now with Sumitomo Corporation, Columbus, OH, USA.

S. Anwar was with Chassis Advanced Technology Department, Visteon Corp., Dearborn, MI 48126 USA. He is now with the Mechanical Engineering Department, Purdue School of Engineering & Technology, IUPUI, Indianapolis, IN 46202 USA (phone: 317-274-7640; fax: 317-274-9744; e-mail: soanwar@iupui.edu).

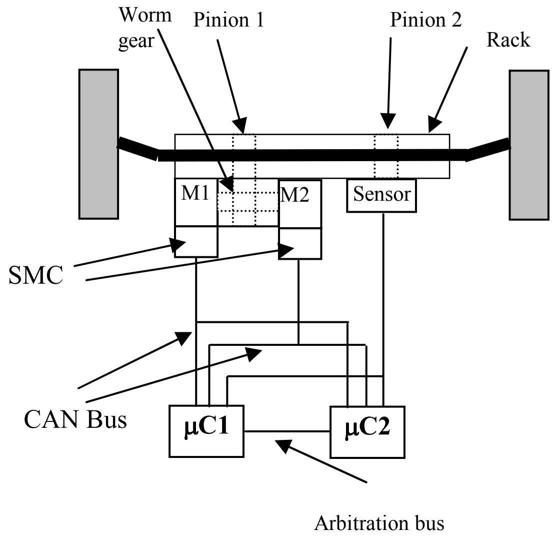

Fig. 1. Fault tolerant SBW control system.

Two microcontrollers (μCs) are used to control the road wheels. Microcontrollers are connected to each other through an arbitration bus as shown in Fig. 2. The value for the ***MySts*** digital output line from each microcontroller is an electrical LOW during system initialization after power up or after reset. When microcontroller loses power, its ***MySts*** digital output line is also LOW by design via external electrical hardware.

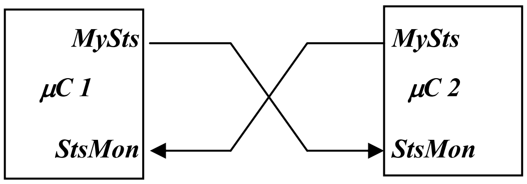

Fig. 2. Master-slave arbitration

III. CONTROL SYSTEM DESIGN

Control system consists of two loops: an inner motor torque control loop and an outer pinion angle control loop. The purpose of the inner motor torque loop is to control motor to generate the required torque as specified by a torque reference, and the purpose of outer pinion control loop is to track commanded steer angle with minimal error.

Fig. 3 illustrates the block diagram of the inner motor torque control loop. The prefilter converts the desired torque T_d into the desired current components i_{dd} and i_{dq} in d-q coordinates. At the same time, the actual motor current components in d-q coordinates, i_d and i_q are calculated using measured motor phase current i_a and i_b as well as motor speed ω. A PI control algorithm is used to calculate the control commands based on the current error between i_{dd}, i_{dq} and i_d, i_q. The command in PWM form is then feed into power electronics (PE) that in turn drives the motor to generate the desired torque. Neglecting the transient response, the equivalent input-output relation of the local torque control loop is described by the following equation

$$T_e = T_d \qquad (1)$$

where T_e is motor output torque.

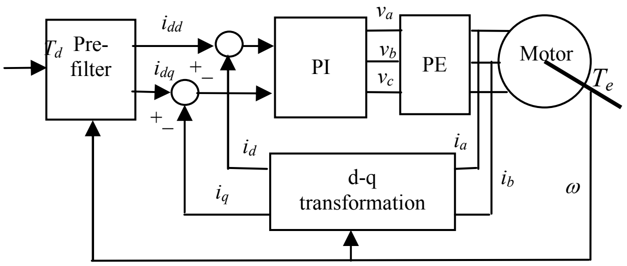

Fig. 3. Smart Motor Controller

Assume that two motor control loops have the same dynamic characteristics and that motor inertia and friction are negligible when compared with the road wheel assembly, the differential equation describing the pinion angle dynamics is then as follows

$$J\ddot{\theta} + b\dot{\theta} + F_c \operatorname{sgn}\dot{\theta} + k_a \tau_a = mu \qquad (2)$$

where

θ: Pinion angle
J: The moment of inertia of the assembly
b: Viscous damping
F_c: Coulomb friction
k_a: Scaling factor
τ_a: Tire self-aligning moment
u: Torque
m: Number of motors

To design the outer loop, the dynamic equation (2) is rearranged as follows

$$J\ddot{\theta} + b\dot{\theta} = mu - \Delta(\theta) \qquad (3)$$

where

$$\Delta(\theta) = F_c \operatorname{sgn}\dot{\theta} + k_a \tau_a \qquad (4)$$

$\Delta(\theta)$ is considered as predictable disturbance.

Based on the equation (3), the outer control loop is partitioned into two parts: a nominal feedback part and a feedforward part. The nominal feedback loop is designed based on the following equation.

$$J\ddot{\theta}+b\dot{\theta}=mu \qquad (5)$$

It can be easily seen that this is the pinion angle system dynamic model (3) with disturbance being truncated. A second order lead-lag compensator as described by (6) is used for the nominal feedback loop. The compensator parameters a_1, a_2, b_0, b_1, b_2 and g are selected based on the nominal model using bode plots. The compensated nominal system has a minimal phase margin of 45 degree and 10 dB gain margin for both single motor operation and dual motor operation.

$$G(s)=g\frac{s^2+a_1s+a_2}{b_0s^2+b_1s+b_2} \qquad (6)$$

The purpose of feedforward loop is to compensate the disturbance as specified in equation (4). For the feedforward compensation, F_c is obtained through experiment using system identification technique [7], [8]. The aligning moment in the equation (4) is directly approximated as the following empirical function of tire slip angle [9]

$$\tau_a=\bar{\tau}_a(\alpha_f) \qquad (7)$$

where α_f is the tire slip angle. Using the vehicle bicycle model [10], the above aligning moment can be converted as the function of the pinion angle with given vehicle speed. Define

$$f(\theta)=F_c\operatorname{sgn}\dot{\theta}+k_a\tau_a \qquad (8)$$

then we have the overall pinion angle control signal be represented by the following equation

$$u=G(s)(\theta_d-\theta)+F(s)\theta \qquad (9)$$

where *F(s)* is a function in complex domain which approximates the relation defined in equation (8).

The block diagram of the overall closed-loop steer-by-wire road wheel angle control system is illustrated in Fig. 4. In the diagram, *P* represents the plant of the road wheel assembly, θ_d. the desired pinion angle. Also, we have neglected the motor loop dynamics due to its fast response when compared with the outer angle control loop.

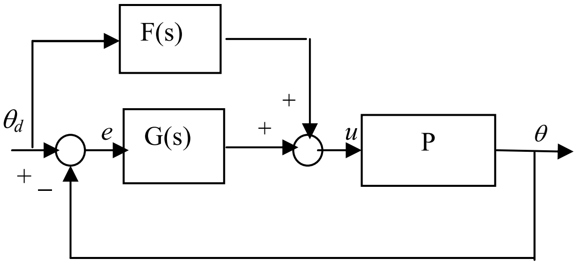

Fig. 4. Equivalent control system

IV. SYSTEM IMPLEMENTATION

Fig. 5 illustrates the implementation of the controller. The inner motor control loop is implemented in SMC and outer road wheel angle control loop is in microcontrollers. Both SMCs share the same program code, and both microcontrollers share the same program code except that one has a longer initialization time than the other.

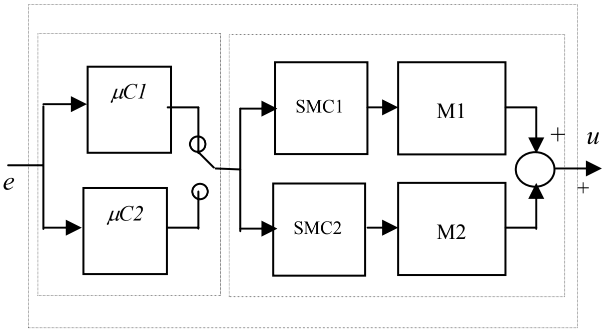

Fig. 5. Control system implementation

A. Master-slave operation

After initialization, each microcontroller checks the ***MySts*** signal output line from the other through the ***StsMon*** digital input port as shown in Fig. 2**.** The microcontroller that reads a LOW signal from others becomes the master. The master then pulls its ***MySts*** digital output line HIGH, and proceeds with control operation. The microcontroller that reads a HIGH signal becomes the slave. As a slave, the micro-controller keeps its ***MySts*** digital output LOW. The slave also calculates its own control command, and compares the calculated result with master's result received through the CAN communication network. Whenever the difference in the comparison of the two control commands is larger than a predefined threshold, the slave will use the control command from the master to reset its own calculation. At the same time, the slave controller continuously monitors the ***MySts*** digital output from the master. As soon as ***MySts*** line from the master is detected LOW for a specified time period, the slave will switch to master operation mode. As master, it will toggle its ***MySts*** digital output line HIGH and proceeds with outer angle control operation.

Each microcontroller is monitored by its own internal *WatchDog*. When a fault occurs within the microcontroller, and the fault is beyond the capability of the exception handler, the *WatchDog* will reset the µC, which will bring ***MySts*** to LOW.

B. CAN bus operation

The messages sent out from the SMCs include resolver values and local diagnostic information. The resolver values are sent with a 1ms period for closed-loop control purpose. The diagnostic information is sent at 10 ms period, and is used by microcontrollers to evaluate the health of the motor control loops.

Only the master sends control command to the SMCs through CAN buses during normal operation. Both SMCs receive the same control commands. The slave also receives the control command from the master and uses it to synchronize its own control calculation with master. If master becomes faulty, the slave takes over the control task by sending its control command to the SMCs through the CAN buses.

Whether the control signal is sent by master or slave, the CAN message ID for the control command is the same. There is no message ID conflict on the bus since master and slave never send control commands at the same time.

C. Sensor Operations

All three angular sensors described in the system architecture, one absolute, and two relative (resolver), are used to measure pinion angle. Upon power on, the absolute sensor value is captured. This captured value is then added to the resolver signal that has a zero output right after power on. With this added value, the resolver functions as an absolute angular sensor whose output can be represented by the following equation

$$\theta_{cal} = \theta_{rel} + \theta_{abs0} \qquad (10)$$

where θ_{rel} is raw resolver signal value, θ_{abs0} is captured absolute sensor value upon the power on, and θ_{cal} is calibrated revolver sensor value.

During normal operation, the readings from all three sensors are constantly compared among each other. A weighted averaging algorithm is used to process the readings as described by the following equation

$$\theta_{flt} = \alpha\theta_{cal1} + \beta\theta_{cal2} + \gamma\theta_{abs} \qquad (11)$$

where

θ_{flt}: Filtered pinion angle
θ_{cal1}: Calibrated pinion angle from resolver 1
θ_{cal2}: Calibrated pinion angle from resolver 2
θ_{abs}: Measured pinion angle from absolute sensor

The value of α ,β, and γ is between 0 and 1. At the same time, the values of α ,β and γ are constrained by the following equation

$$\alpha + \beta + \gamma = 1 \qquad (12)$$

In most cases, more weights are added to the calibrated resolver readings due to its higher resolution and linearity. When the difference among the sensor readings is larger than a specified threshold, the reading from the sensor that has the largest difference from the others will be omitted, i.e. the corresponding coefficient from the above equation becomes zero. Nevertheless, this sensor reading will be used again for the weighted averaging during the next calculating cycle. But, if this large difference lasts longer than a specified period, the sensor will be considered faulty and the corresponding readings will be discarded for the rest of the operation.

V. TEST RESULTS

The control system developed has been implemented in a Steer-By-Wire test bench at Visteon Chassis Advanced Technology. The fault tolerance test of the system was embedded into the road wheel assembly's accelerated durability test, and the test results are illustrated in Fig. 6 and Fig. 7.

For the accelerated durability test, a reference pinion angle signal is applied to the control system reference input. This signal has a frequency of 0.115 Hz, and amplitude of 125 degree. At the same time, an active load of 8896 Newton is applied to the tie-rod of the assembly. This active load has a waveform varying from a constant to a 5 Hz sinusoid shape or vise versa, depending on the stage of the durability test.

Fig. 6 illustrates the test result for the motor loop fault-tolerance capability. Fig. 6a and 6b show the phase currents of the motor, and Fig. 6c shows the comparison between measured and desired pinion angle. During the durability test, fault was injected into the inner loop of motor 2. The corresponding SMC detects the fault via a diagnostic code and shuts the inner loop down. The overall control system switched automatically to the single motor control mode. It can been from Fig. 6a that there was an phase current increase for the motor 1 during the switchover due to the shutdown of motor 2 which has zero phase current after switchover. But, no pinion angle tracking degradation was observed in the captured pinion angle data which is illustrated in Fig. 6c. With this test, the active load is a constant.

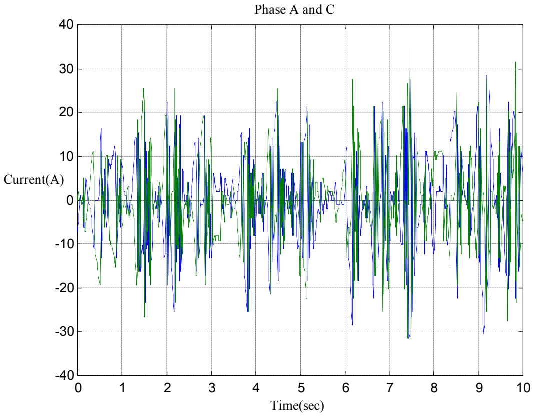

Fig. 6a. Phase current of motor 1.

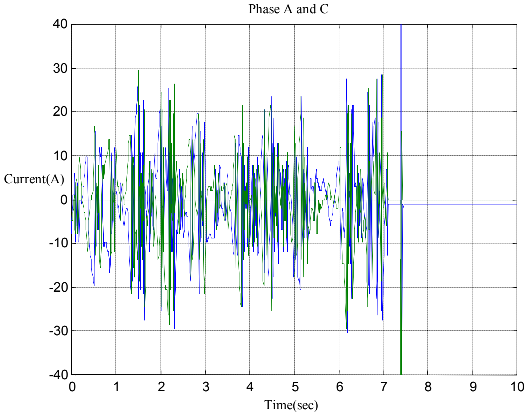

Fig. 6b. Phase current of motor 2.

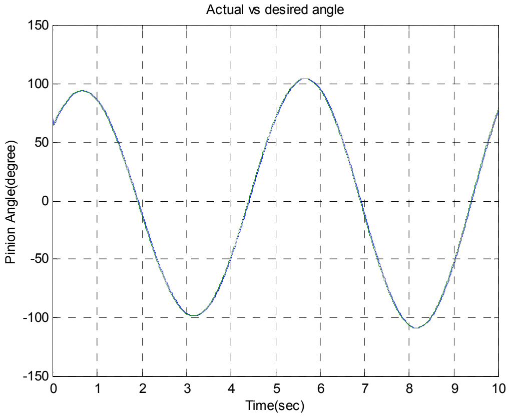

Fig. 6c. Pinion angle: actual vs. desired.

The result for microcontroller switchover is shown in Fig. 7. As in Fig. 6, the phase currents, and the comparison between measured and desired pinion angle shown in the corresponding Fig. a, b, and c. A fault is injected into master microcontroller during the test. The slave controller detects the MySts change from master immediately, and switched into master operating mode right after the detection. Fig. 7d shows the status change from the prospect of slave microcontroller. Again, no pinion angle tracking degradation was observed in the captured pinion angle data which is illustrated in Fig. 7c. The high frequency oscillation appeared on the measured pinion angle curve reflects the fact that active load is now in its 5 Hz sinusoid wave form.

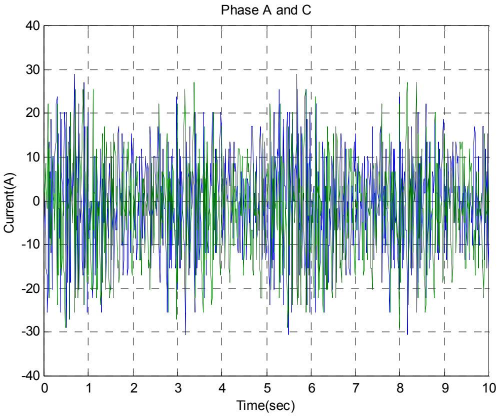

Fig. 7a. Phase current of motor 1.

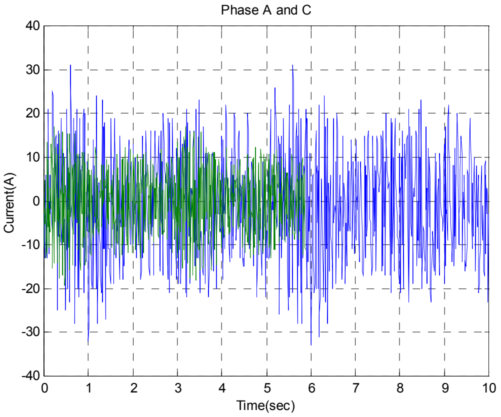

Fig. 7b. Phase current of motor 2.

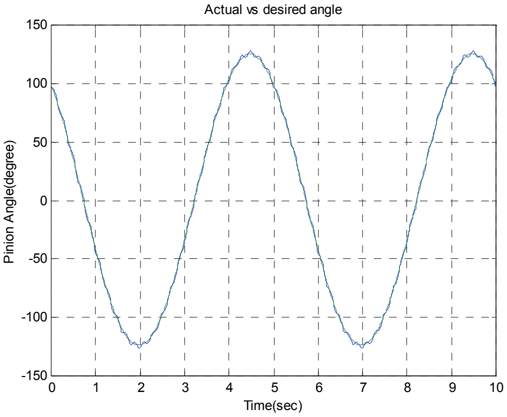

Fig. 7c. Pinion angle: actual vs. desired.

VI. CONCLUSION

The proposed control architecture has shown the benefit of fault tolerance capability. In the event of single point fault, either caused by one motor control loop, or by one microcontroller, the system continues to provide the road wheel steering functions without degradation of tracking performance.

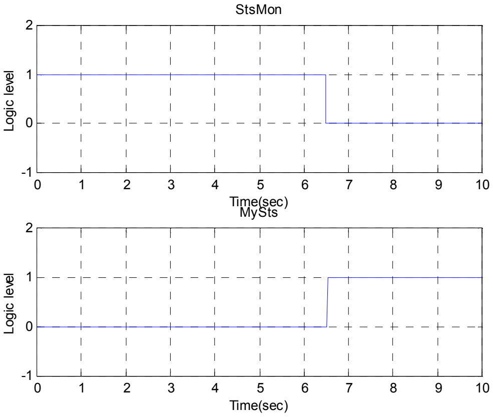

Fig. 7d. Arbitration bus signal.

Since the total power needed for steering the vehicle can be contributed by both motors, the maximum power output of each motor can be reduced by half, which in turn allows the usage of the power electronics with lower power consumption. This will reduce the complexity of the power electronics design with respect to the noise, EMI/EMC, as well as thermal release.

Finally, since both microcontrollers share the same program code, the cost related with the software development, debug, and maintenance can be greatly reduced.

ACKNOWLEDGEMENT

The authors wish to acknowledge Mr. Matt Mikesell, senior manager of Chassis Advanced Technology at Visteon Corporation, for his review, comments and suggestions on the paper.

REFERENCE

[1] J. Ackermann and T. Buente, "Automatic car steering control bridges over the driver reaction time", *Kybernetika*, Vol. 33, No. 1, pp 61-74, 1997.

[2] J. Kasselmann and T. Keranen, “Adaptive steering”, *Bendix Technical Journal*, Vol. 2, pp 26-35, 1969.

[3]. J. Ackermann, “Yaw disturbance attenuation by robust decoupling of car steering”, *In proceedings of the IFAC World Congress*, San Francisco, CA, 1966.

[4] K. Huh and J. Kim, “Active steering control based on the estimated tire forces”, *Journal of Dynamic Systems, Measurement and Control*, Vol. 1223, pp 505-511, 2001.

[5] M. Segawa, K. Nishizaki and S. Nakano, “A Study of vehicle stability control by steer by wire system”, *Proceeding of the International Symposium on Advanced Vehicle Control*, Ann Arbor, MI, 2002.

[6] B. Zheng et al, “Active steering control with front wheel steering”, *American Control Conference*, pp 1475-1480, 2004.

[7] B. Zheng and Ning-show Hsu, “SISO system identification via block-pulse function”, *International. Journal of Systems Science*, Vol. 13, No. 6, pp 697-702, 1982.

[8] B. Zheng and Ning-show Hsu, “Analysis and parameter estimation of bilinear system via block-pulse function”, *International. Journal of Control*, Vol. 36, No. 1, pp 53-65, 1982.

[9] H. Pacejka et al, “Tyre modelling for use in vehicle dynamics studies”, *SAE technical paper,* No. 870421.

[10] P. Riekert and T. Schunck, "Zur Fahrmechanik des gummibereiften Kraftfahrzeugs", *Ingenieur Archieve*, Vol. 11, pp 210-224, 1940.