6

Control system implementation

6.1 Introduction

This chapter describes the selection and design of the various components required to implement the suspension control system for the experimental research vehicle. The experimental single electromagnet suspension uses a subset of the vehicle suspension components.

The full set of vehicle control algorithms developed in Chapters 3, 4 and 5 is computationally complex. Therefore, in order to provide a flexible experimental system which is capable of being freely modified for current and future research work, a digital signal processing approach was chosen. Additional benefits of this approach include the elimination of drift and offsets in the signal processing, and the ability to implement readily nonlinear functions. The disadvantages of digital processing are primarily those due to the time and amplitude discretisation of the signals.

The implementation of the experimental control system is described in five sections. First, the system requirements for the transducers, converters, and signal processors are identified. Suitable industrial feedback sensors are then selected and an electromagnet current controller is designed. Next, the signal processing, signal conversion and data communication subsystems are designed. The design and configuration of the software which implements the control algorithms is then described, and finally, conclusions are drawn about the system implementation.

6.2 System requirements

The experimental control system performs three basic tasks. It measures the air gap and acceleration of each electromagnet, it calculates the electromagnet current demands according to the control algorithms developed in Chapters 3, 4, and 5, and it adjusts and maintains the electromagnet currents at the demanded levels.

In order to perform these tasks, the control system is functionally decomposed into five subsystems, namely feedback sensors, analogue to digital converters, signal processors, digital to analogue converters, and finally, electromagnet current controllers. The last subsystem is implemented in the analogue domain to reduce the digital signal processing load. The critical requirements for each subsystem are specified in terms of their bandwidth, range, resolution, and accuracy. These specifications are calculated by first considering the required signal bandwidths and sampling rates, and then the required signal ranges, resolutions, and accuracies.

6.2.1 Bandwidths and sampling rates

The suspension system has been designed, and the control algorithms developed, using continuous-time methods. However, since discrete-time, digital signal processing is to be used, it is necessary to determine an acceptable maximum time interval between the iterations of the controller. Shannon's sampling theorem states that a sample rate of twice the highest frequency component of a signal is theoretically sufficient to describe that signal completely.[86] However, for real-time, closed-loop control applications, the feedback loop delay introduced by Shannon's sampling rate would severely disturb the location of the closed-loop poles, and adversely affect the closed-loop response. In order to reduce the adverse effects of the feedback loop phase delay and hence obtain a good response, the sample interval generally needs to be 1/5 to 1/10 of the time constant of the dominant system pole.[87] This results in a sampling rate which is 15 to 30 times Shannon's theoretical sampling rate.

In addition to determining an acceptable delay for discrete-time sampling, an acceptable phase delay for the air gap and acceleration sensors must also be determined. The phase lag due to the electromagnet current controller is already incorporated in the design of the suspension control algorithms, so it requires no further consideration.

In order to produce a control system design which makes efficient use of the various subsystem components, a range of acceptable phase delays is calculated, and this is then apportioned between the feedback sensors and the discrete-time controller.

Table 4.4 lists the location of the dominant poles and zeros of the closed-loop position control system for the single electromagnet suspension. For the purpose of this analysis, the poles and zeros of the vehicle mode position controllers can be considered to be the same as those of the single electromagnet suspension. The suspension response is dominantly third-order, with a highest pole frequency of around 300 rad/s for both the unloaded and fully loaded cases, and the calculated minimum phase delay at this frequency is 3.0 rad (172°). By using the sample rate guideline identified earlier, an additional phase delay of 10-20% for the position controller processing and sensor delays is assumed to be acceptable. This permits an additional phase delay of 0.3-0.6 rad (17-34°) at 300 rad/s, which can also be considered as a time delay of 1.0-2.0 ms.

Figure 3.4 shows the location of the open-loop poles for the electromagnet at the worst case operating point. The unstable pole has a worst case value of 167 rad/s, at which the total open-loop phase delay is 3.4 rad (195°). Using the same assumption as before, this permits additional force controller and sensor delays of 0.34-0.7 rad (20-39°) at 167 rad/s, which can be considered as a time delay of 2.0-4.0 ms.

The phase and time delay requirements for the position controller are more exacting than those for the force controller, so they are used for the baseline requirements. The approximate equality between the delays occurs because the suspension control algorithm parameters were designed to make maximum use of the available electromagnet force actuation bandwidth. It also requires the same overall control system bandwidth to be used for both the suspension position control algorithm and the electromagnet force control algorithm.

The phase delays contributed by the air gap and acceleration sensors occur concurrently, whilst the signal processing time delay follows sequentially. Since attaining high digital sampling rates is more costly than high sensor bandwidths for this application, the signal processing should be allocated a larger share of the available time delay than the feedback sensors. Consequently, the signal processing is allocated roughly two thirds of the allowable phase delay, with the remaining one third left for the feedback sensors.

The phase delay generated by the first-order phase lag inherent in suitable low-cost industrial air gap sensors is given by:

$$\theta_{sensor} = \tan^{-1}\frac{\omega}{\omega_{sensor}} \approx \frac{\omega}{\omega_{sensor}} \qquad since \qquad \omega < \omega_{sensor}$$

$$\therefore \quad \omega_{sensor} \approx \frac{\omega}{\theta_{sensor}} \qquad \textbf{6.1}$$

where θ_{sensor} is the sensor phase delay, ω_{sensor} is the sensor bandwidth, and ω is the signal frequency. The air gap sensor bandwidth required to generate a phase delay of no more than 0.1-0.2 rad at a signal frequency of 300 rad/s is therefore 1500-3000 rad/s. This is equivalent to a sensor pole time constant of 0.3-0.7 ms.

Suitable low-cost industrial accelerometers use a sprung mass as the sensing element and have a second-order low-pass response. Therefore, in order to generate the same total phase delay as the air gap sensor, the accelerometer must have a natural resonant frequency of 3000-6000 rad/s.

The time delay allowed through the signal processors is given by the allowable processing phase delay (0.2-0.4 rad) divided by the highest closed-loop pole frequency (300 rad/s). This gives an allowable signal processing time delay of 0.7-1.3 ms. Assuming that the signal processing takes about 80% of the sampling interval, the average time delay due to the discrete-time signal conversion and digital processing is 1.3 times the sample interval, and so a sampling interval of 0.5-1.0 ms is required.

The assumptions made about acceptable levels of phase delay were verified using a detailed simulation of the full closed-loop suspension system (see Appendix C). The additional phase delay of 10-20% had little impact on the closed-loop response, whilst phase delays greater than 80% of the original value caused marginal instability problems at the worst case operating point. Phase delays of less than 5% had a negligible impact on the suspension response.

Table 6.1 summarises the bandwidth and sampling interval requirements that have been calculated for the feedback sensors and the digital control algorithms. These figures are guidelines between which trade-off adjustments can be made as necessary.

Table 6.1 Bandwidth and sampling interval specifications feedback sensors

Component	Parameter	Value
Accelerometer	natural frequency	480-960 Hz
Air gap sensor	bandwidth	240-480 Hz
Electromagnet force controller	sampling interval	0.5-1.0 ms
Suspension position controller	sampling interval	0.5-1.0 ms

6.2.2 Range, resolution and accuracy

The normal operational envelope for the suspension system (see Chapter 3) covers an electromagnet air gap range of 1-5 mm. However the air gap range required for the air gap sensors is 0.5-5.5 mm in order to permit operation up to the mechanical limit stops. Whilst the normal operational acceleration levels are very low, higher levels are required for testing the suspension position controller. In addition, since the absolute velocity and position are calculated from the acceleration measurement, it is imperative that the acceleration signal is not allowed to saturate. Therefore, to accommodate step response tests, a maximum acceleration measurement range of about ±1 g is required. Operation of the electromagnets over their full operational envelope requires the current to be controlled over the range 0-20 A.

The resolution and hysteresis of the transducers and the associated analogue-digital converters affect the amplitude of the limit cycle oscillations of the control system.[88] Since acceleration is the primary quality control factor for the suspension system, an acceleration limit cycle amplitude of no more than about 20% of the target comfort threshold of 0.4 m/s^2 (see Chapter 4) is considered desirable. This calls for an acceleration resolution of 80 mm/s^2. Assuming an integration interval of 1 ms for the velocity and position integrators, and sufficient numerical precision, this results in a velocity resolution of 80 μm/s, and a position resolution of 80 nm.

The required force actuation resolution is calculated next because it is a useful reference value for calculating the required air gap and current resolutions. The force resolution is related to the acceleration resolution by:

$$\Delta force = mass\ \Delta acceleration \qquad \textbf{6.2}$$

where $\Delta force$ is the force resolution and $\Delta acceleration$ is the acceleration resolution. At the maximum load of 50 kg per electromagnet, an acceleration resolution of 0.08 m/s^2 requires a force resolution of 4 N per electromagnet.

The required air gap signal resolution is determined by considering the force resolution and the air gap feedback gain. The relationship between the air gap resolution and the force resolution is given by:

$$\Delta airgap = \frac{\Delta force}{air\_gap\_gain} \qquad \textbf{6.3}$$

The air gap feedback gain for the electromagnet force controller is determined by the open-loop electromagnet stiffness. This gives rise to a worst case, full load gain of 950 N/mm (see Table 3.2). Since the required force resolution calculated above is 4 N, this requires an air gap signal resolution of 4 μm. This resolution calls for an expensive, precision air gap sensor. However, the electromagnet stiffness at the nominal operating air gap is only one third of the worst case stiffness. The nominal operating point therefore requires an air gap resolution of 12 μm which is close to that obtainable by standard industrial sensors which cost significantly less than precision sensors. Therefore, in order to reduce costs, the lower resolution is preferred and a slight degradation of the response at small air gaps is anticipated. The use of multiple air gap sensors to improve the air gap measurement accuracy over the electromagnet's full length, would also increase the effective sensor resolution, as well as providing sensor redundancy. However, in order to reduce the cost and complexity of the experimental vehicle, only one air gap sensor per electromagnet is employed.

The air gap feedback signal also forms the very low frequency component of the calculated track position which is filtered and fed into the suspension position controller. However, since the position feedback gain per electromagnet is 250 N/mm (see Chapter 4), this presents a less stringent resolution requirement than that identified above. The position resolution of 80 nm, calculated previously from the acceleration resolution and integration interval, is clearly satisfactory.

The required current control resolution is determined in a similar manner to the air gap signal resolution. In this case, the worst case open-loop electromagnet force/current ratio is 400 N/A (see Table 3.2), which requires a current resolution of 10 mA.

The acceptable accuracy for the air gap and current transducers is determined by considering a simplified equation for the electromagnet force and the corresponding error equation. These are given by:

$$f = k\,\frac{i^2}{g^2} \qquad \therefore \qquad f_{err} = k_{err} + 2i_{err} + 2g_{err} \qquad \textbf{6.4}$$

where f is the lift force, i is the coil current, g is the air gap and k is the constant of proportionality. The error of the nonlinear force controller model is represented by k_{err}, which is about ±7% (see Section 2.3.7). By assuming a current controller tolerance of about ±2% and an air gap sensor tolerance of around ±3%, the resultant force controller accuracy is expected to be about ±17%.

The accuracy required for the accelerometers is calculated by determining an acceptable deviation in the characteristic response of the position controller. Under normal operating conditions, a load change from full load to no load increases the natural undamped frequency, ω, and damping ratio, ζ, (see Equation 4.7) by a factor of around 50%. Therefore, a further discrepancy of about ±10% to allow for transducer errors is considered acceptable. Since both the velocity and position signals are calculated from the acceleration signal, an error in acceleration measurement gain is modelled as an equivalent error in the position, velocity and acceleration feedback gains. The suspension parameters and their worst case error equations are approximately given by:

$$\omega \propto \sqrt{k_{pos}}, \qquad \zeta \propto \frac{k_{vel}}{\sqrt{k_{pos}}} \qquad \therefore \qquad \omega_{err},\ \zeta_{err} = \frac{acc_{err}}{2} + \frac{f_{err}}{2} \qquad \textbf{6.5}$$

where k_{pos} and k_{vel} are the position and velocity feedback gains, ω_{err} and ζ_{err} are the parameter errors, and acc_{err} and f_{err} are the acceleration measurement and force actuation errors respectively. Since the expected force actuation error is ±17% (see above), an accelerometer accuracy of ±3% is required to achieve a total suspension parameter deviation of ±10%.

Table 6.2 summarises the range, accuracy and resolution required to measure, convert, and process the acceleration, air gap, and current signals. The accuracy is an aggregate figure which should be achieved on an end-to-end basis. In practice, the very high accuracy of analogue-digital converters and digital signal processing means that the accuracy figures can be used as transducer requirement specifications.

Table 6.2 Range, accuracy and resolution specifications

Signal	Range	Accuracy	Resolution
Acceleration	±10 m/s^2	± 3%	0.08 m/s^2
Air gap	0.5-5.5 mm	± 3%	12 μm
Current	0-20 A	± 2%	10 mA

6.3 Transducers

The suspension position control and the electromagnet force control algorithms require the measurement of the electromagnet air gaps and accelerations. The control signal generated by the control algorithms is a current demand, and this is actuated by a high-gain, closed-loop current controller. In order to complement the benefits of the electromagnetic suspension, the feedback sensors should not make contact with the track.

The transducer requirements identified in the preceding section are now used to select commercial devices for the feedback sensors, and the design of a special purpose electromagnet current controller is described.

6.3.1 Accelerometer

A large range of industrial accelerometers is available which measure absolute acceleration. Most accelerometers employ a mass constrained by some stiffness and damping, and measure deflection of the mass to determine the acceleration. The sensitivity of such a device is generally a function of the sprung mass divided by the stiffness. Since the natural frequency is also a function of the sprung mass divided by the stiffness, the natural frequency and hence the bandwidth of these devices is closely related to the sensitivity.

The selection of a suitable accelerometer is thus largely determined by the sensitivity/bandwidth constraint, and this appears to be somewhat more exacting for electromagnetic suspension control applications than it is for many other industrial requirements. Other important device parameters include cross-axis sensitivity, thermal

stability, resolution, linearity and hysteresis. The last two factors are particularly important because the acceleration signal is double integrated within the control system to calculate the absolute velocity and position.

The basic performance requirements for the accelerometer are the measurement range and resolution, the natural frequency and the general accuracy, and these are ±10 m/s^2, 0.08 m/s^2, 480-960 Hz and ±3% respectively (see Table 6.1 and Table 6.2). These requirements are met most cost effectively by an industrial micro-machined silicon sensor, which incorporates temperature compensated signal conditioning circuitry on the silicon substrate. The device has a full scale range of ±20 m/s^2, a resolution better than 0.04 m/s^2, a natural frequency of 600 Hz, and a maximum error of ±3%. The full specification is listed in Appendix D.

6.3.2 Air gap sensor

Non-contacting air gap measurement is used extensively in manufacturing processes where the measurement target is moving. Capacitive, inductive and eddy-current measurement techniques form the basis of most industrial non-contacting air gap sensors. Capacitive and inductive devices use the measurement target as part of a capacitor or inductor and determine the air gap from the measured capacitance or inductance. Eddy-current devices measure the power loss from a coil due to the power dissipation associated with eddy-currents circulating in the flux-coupled target.

The maximum performance requirements for the air gap sensor are determined by the electromagnet force controller which calls for a measurement range of 0.5-5.5 mm, a resolution of 12 μm, a bandwidth of around 240-480 Hz, and a basic accuracy of ±3% (see Table 6.1 and Table 6.2). A standard industrial eddy-current loss air gap sensor has been selected which has a useable air gap measurement range of 3-10 mm, a resolution of 15-30 μm, a bandwidth of 190 Hz, and a basic accuracy of ±3% when suitably calibrated. The full specification for the device is listed in Appendix D. The specification is slightly lower than desired, but precision devices are typically 3-5 times the price of the standard industrial devices. Since the overall system performance is affected by both the air gap sensor and the accelerometer, the marginal performance of the air gap sensor is partly offset by the high resolution and bandwidth of the accelerometer.

Eddy-current loss devices may not be appropriate for use on a production system because the output is dependent on the target material resistivity which varies with target temperature and material. Rapid movement of the sensor along the track may also cause reduced accuracy due to eddy current losses induced by the motion of the sensor along the track.

6.3.3 Electromagnet current controller

The electromagnet force control algorithm developed in Chapter 3 incorporates a closed-loop electromagnet current controller, with a loop gain of 100. The implementation of the current controller is now described, and it is based on the design parameters derived in Chapter 3.

The primary dynamic characteristic of the current controller is determined by the current feedback gain, which has already been calculated. The secondary dynamic characteristic is the current slew rate, which is determined by the power supply voltage and the characteristics of the electromagnet. Therefore, a power supply voltage for the current controller must be calculated which gives an acceptable minimum current slew rate.

Since the electromagnet is a component of a system designed to provide a comfortable suspension ride, a force slew rate of 10% of the maximum load force per force actuation time constant ($T_{force} \approx 3.2$ ms) is considered to be ample. The worst case operating point in terms of the force slew rate occurs at the minimum air gap, since it suffers the highest eddy current lag time constant. At this operating point (1 mm, 500 N), the electromagnet requires an excitation current, I_{magnet}, of 4 A. Generation of a ±10% force change requires only a ±5% change in current due to the square law relationship between the electromagnet current and the force. The required current change, ΔI_{magnet}, is therefore ±0.2 A. The required supply voltage, which is the sum of the voltage needed to slew the current plus the steady-state coil voltage, is given by:

$$V_{supply} = L_{coils} \frac{\alpha_{phase\_lead} \, \Delta I_{magnet}}{T_{force}} + I_{magnet} \, R_{coils} \qquad volts \qquad \mathbf{6.6}$$

where α_{phase_lead} is time constant ratio of the eddy current phase-lead compensator, and R_{coils} is the resistance of the electromagnet coils. The maximum required supply voltage is therefore:

$$V_{supply} = 0.111\,H\,\frac{3\times 0.2A}{0.0032\,s} + 4A\times 0.8\,\Omega = 24\,V \qquad \textbf{6.7}$$

The supply voltage required at an air gap of 5 mm with a full load is, surprisingly, about the same as that required at 1 mm. At the 5 mm operating point, the reduced voltage demand due to the smaller eddy current lag time constant, is offset by the increased voltage demand due to the higher leakage flux and the higher steady-state current.

The supply voltage required for a given implementation is slightly higher due to the additional resistance of the power controlling devices, which are connected in series with the electromagnet. A suitable technique for amplifying the low power control signal to the high currents required for the electromagnet is now considered.

If a linear mode (class A) power amplifier design were employed, the maximum theoretical efficiency would be about 33% since the electromagnet would have an average voltage of 8 V across it, with the remainder dropped across the series current regulator. This would produce an average amplifier dissipation of 16 V x 10 A = 160 W, and require a power supply rating of 24 V x 20 A = 480 W. Alternatively, the use of a switch mode (class D) power amplifier typically enables an efficiency of about 85% to be achieved,[89] thus reducing the average amplifier dissipation to about 12 W, and the power supply rating to about 380 W.

A switch mode current controller is therefore desirable and since no suitable commercial units were available, an electromagnet current controller was designed.[90] Figure 6.1 shows the schematic diagram of the controller which incorporates electrical isolation between the low power signal circuitry, and the high power circuitry.

The current is sensed using an electrically isolated precision current sensor with an accuracy better than 1%, and a response time of about 1 μs. The measured current is compared with the reference current demand and amplified to form the current error signal. This analogue signal is fed into an industry standard pulse width modulator (PWM) integrated circuit, which drives an electrically isolated H-bridge power switch.[91] Since only a uni-directional current drive is required, two active switches are used in the bridge configuration,[92] with passive devices used in the two remaining legs.

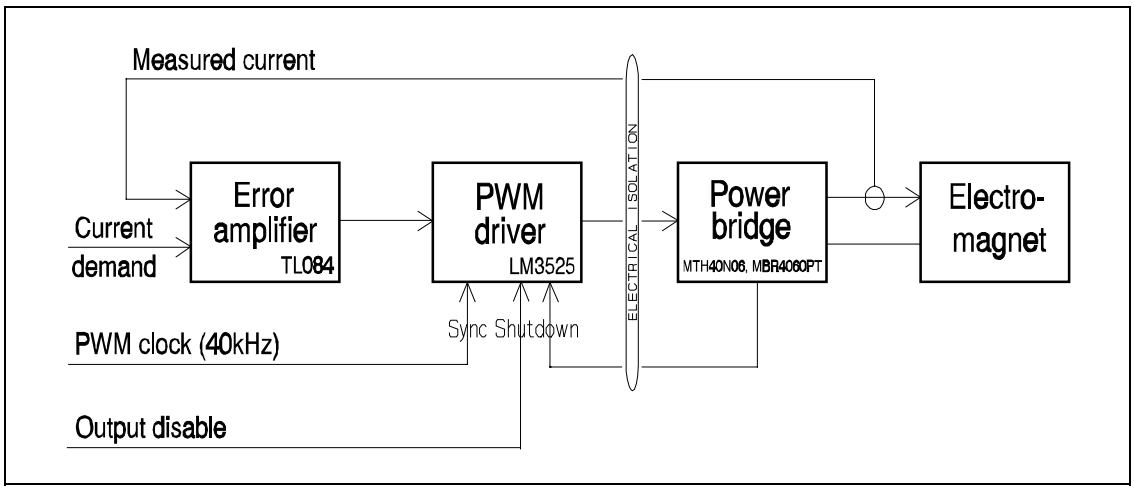

Figure 6.1 Schematic design of the electromagnet current controller.

The time constant of the closed-loop current response is around 1 ms, so a sampling (ie. PWM switching) frequency of at least 10 kHz is desirable. In order to introduce a negligible phase delay due to sampling, and also to make the switching inaudible, a PWM frequency of 40 kHz is used. This switching frequency favours the use of MOSFET devices rather than thyristors or bipolar transistors for the bridge switches.[93] The selected MOSFET devices have a maximum, temperature derated, drain-source resistance of 50 mΩ, and are switched in 100-200 ns by a MOSFET driver integrated circuit. This results in a total power dissipation of about 16 W at the nominal current rating of 10 A. The sensor accuracy and closed-loop gain combine to give the current controller an accuracy better than ±2%.

Each current controller module incorporates an on-board PWM oscillator which permits independent operation of the module. However, for the experimental vehicle which has four electromagnets, the slight difference in frequency between each of the PWM oscillators would generate low frequency beat noise. This would be undesirable since the noise would be within the bandwidth of the feedback control signals. Therefore, for the vehicle configuration, a single oscillator source is used to ensure synchronous operation of the four current controller modules, and hence prevent any low beat frequency noise.

The current controller and electromagnet are also protected by two safety features. First, the current controller receives an output disable control which is activated if necessary by a system watchdog timer. In addition, a current limiter set to a level of 21 A is incorporated in the design.

The high power supply for the current controllers is provided by commercial, mains-powered switch mode supplies rated at 27 V and 15 A. The circuit diagrams for the electromagnet current controller are listed in Appendix D.

6.4 Signal processing, conversion and communication

To ensure sufficient flexibility to investigate control algorithms of increased complexity in the future, a scalable processing resource is desirable. A further requirement for flexibility and future research is for a distributed processing system. In addition to these system features, good language support and development system support for scalable and distributed processor systems are required.

Researchers in the fields of real-time control and simulation[94,95,96,97] have found that parallel processing architectures can offer significant performance and modularity advantages over single processor architectures.

Three microprocessor development systems were available for use with this work. The target microprocessors and co-processors were the Intel 80386/80387, the Motorola 68030/68881, and the Inmos T800 transputer. In a study[98] of these and other processors, the transputer came out most favourably in terms of processor performance, connectivity, parallel language support, and multi-processor development support.

Academic and industrial researchers[99,100,101,102] have also found the transputer to be particularly suitable for implementing parallel processing architectures. Initial work using one transputer to perform the signal processing for an experimental single electromagnet suspension[103] confirmed that a T800 transputer based system was capable of meeting the processing requirements at sampling rates around 1 kHz.

6.4.1 Transputers

The Inmos transputer family[104] is a range of microprocessors designed for use as parallel processing components. The family includes 16 bit integer processors (T2xx), 32 bit integer processors (T4xx), and 32 bit processors with a floating point unit (T8xx). The T800 transputer (Figure 6.2) contains the following system components:

- 32 bit central processing unit (12.5 MIPS at 25 MHz);
- 64 bit floating point processor (1.5 MFLOPS at 25 MHz);
- 4 Kbytes internal memory;
- 2 timers;
- 4 bi-directional 5-20 Mbits/sec inter-processor serial communication links.

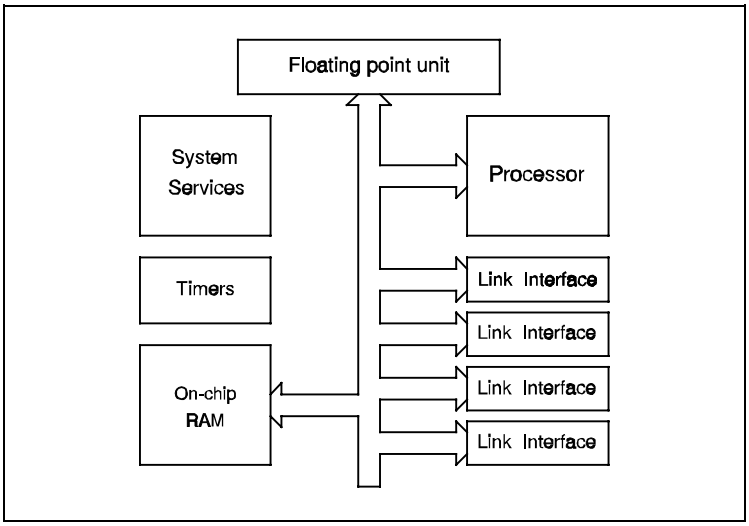

Figure 6.2 T800 transputer architecture.

The inter-processor communication links facilitate the use of distributed processor systems with distributed memory. Unlike conventional bus systems with shared memory, the inter-process communication bandwidth rises linearly with the number of processors. Memory and input/output facilities can be readily extended using external devices.[105] Data transfers using the inter-processor links are performed concurrently with process execution using direct memory access (DMA). This results in a low performance degradation even when all links are running at full capacity. The communication links can be connected directly between processors on a single card or in a chassis unit. For longer distances, twisted pair, coaxial cable or fibre-optic links can be simply and effectively used. Communication links between transputers and links to external devices can be configured using a 32-way cross-point switch if necessary.

In order to provide flexibility when configuring software processes onto transputers, a built-in hardware scheduler is provided which multi-tasks parallel processes on a single processor.[106] This enables algorithms to be coded in parallel without constraining them to a particular parallel processor configuration. The scheduler performs a task switch in about 1-4 microseconds which is extremely fast, thus permitting multiple parallel threads to be efficiently used even with sampling intervals of approximately 1 ms.

Transputer development systems are available for a range of computer platforms. For this work, an IBM compatible PC/AT is used with a plug-in card which accommodates the development host processor. The development system software and application software use the PC as a terminal and file server. Development system support for many programming languages is available, most of which have parallel programming extensions. For this work, occam is used because it provides a flexible and robust environment, with particular regard to the data security issues of parallel processing.

6.4.2 Control system hardware structure

A modular hardware system has been developed to run the vehicle controller and the single electromagnet rig controller. The system functions are partitioned and implemented using six cards which can be freely configured. The functional specification for each card is listed in Table 6.3. The TRAM motherboard takes standard Inmos TRAM modules which consist of a transputer processor plus random access memory (RAM).

The cards for the vehicle control system are partitioned between two chassis. One contains the feedback sensor signal conditioning, the analogue to digital converter and three TRAM motherboards. The other contains the digital to analogue converter and the four electromagnet current controller modules. For the single electromagnet control system, all the cards and modules are located in a single chassis.

In order to prevent earth-loop and other noise problems, the chassis and the development computer system are all electrically isolated from each other. Digital fibre optical connections operating at 10 Mbit/s are used for all inter-unit transputer link communications. Rack input and output interface cards provide the optical input and output facilities between equipment units. Since the optical link connections are short, optical signal power loss is not a problem, and so plastic optical fibre cable is used which is inexpensive.

The issues affecting the design of the analogue signal conditioning and conversion cards, and the TRAM motherboard and optical interface cards are discussed next. The chassis configurations and circuit diagrams for all the cards are detailed in Appendix D.

Table 6.3 Control system hardware function specifications.

CARD	FUNCTION
SAP	Sensor Analogue Processor: - Current to voltage conversion for 4 air gap sensors. - Filtering & amplification for 4 accelerometers.
ADC	Analogue to Digital Converter: - 8 analogue input channels. - Anti-alias filters. - Multiplexer selection. - 12 bit, bipolar ADC, 10 μs conversion time. - 16 bit, 10 MIPS integer processor (T212). - Front panel switch input.
TMB	Tram MotherBoard: - Motherboard with 4 size1 TRAM sites.
DAC	Digital to Analogue Converter: - 4 buffered analogue output channels. - 12 bit unipolar DACs. - 16 bit, 10 MIPS integer processor (T212). - Watchdog timer. - Front panel switch input.
IFIN	InterFace INput: - Optical chassis input interface. - 1 bidirectional transputer data link. - Transputer reset & analyse control inputs. - Transputer error control output.
IFOUT	InterFace OUTput: - Optical chassis output interface. - 1 bidirectional transputer data link. - Transputer error control input. - Transputer reset & analyse control outputs.

6.4.3 Analogue signal conditioning and conversion

The schematic design of the analogue to digital converter card is illustrated in Figure 6.3. Eight analogue input channels are provided, each of which is buffered and filtered to remove high frequency noise which would otherwise be aliased[107] down to the frequency bandwidth used by the control system signals. The anti-alias filters have a critically damped, second-order response, with a corner frequency of 1.6 kHz. The

electromagnet current controllers switch a total of up to 80 A at a frequency of 40 kHz. They can therefore be expected to be a source of noise which will be inductively and capacitively coupled to the feedback signals. Such noise (at >40 kHz) is reduced by 56 dB by the anti-alias filters. The anti-aliased signals are selected by an analogue multiplexer which routes the selected signal to the 12 bit analogue to digital converter, where it is sampled and converted to the digital domain. A parallel input port is also provided which is used solely to read the state of a switch mounted on the front panel of the card.

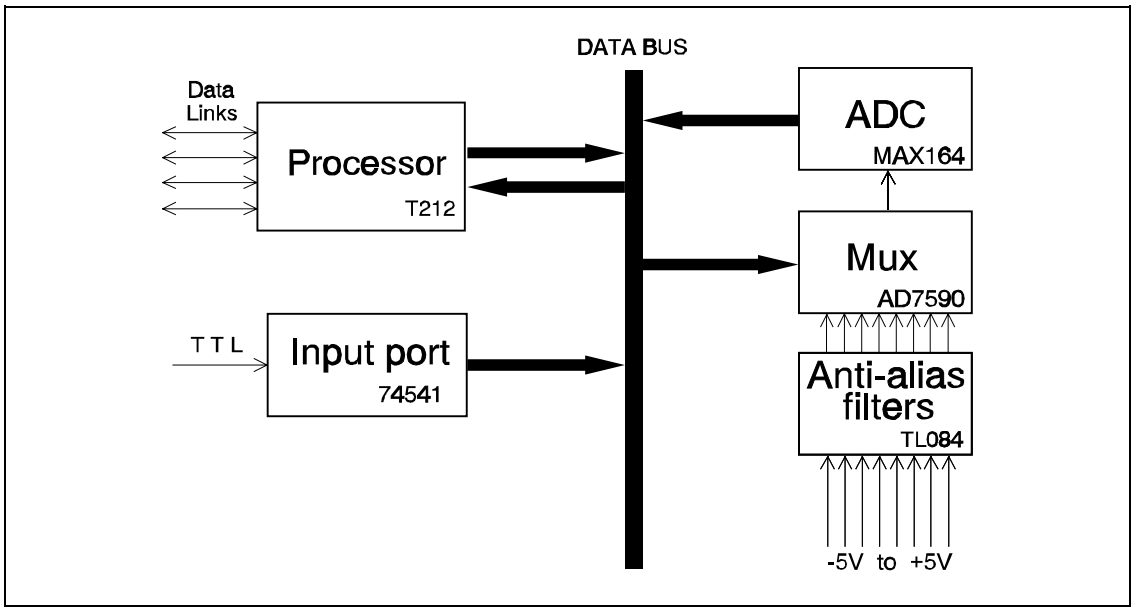

Figure 6.3 Schematic diagram of the analogue to digital converter.

The sensor analogue processor card applies the appropriate signal conditioning and amplification to the sensor signals to give a full scale deflection of 0-5 V for the air gap sensor and ±5 V for the accelerometer. In addition, the accelerometer signals are filtered by a first-order high pass filter with a corner frequency of 0.1 Hz to remove the 1 g measurement offset due to gravity.

The 12 bit analogue converter produces a channel quantisation amplitude of 2.5 mV which in turn gives an air gap resolution of 5.0 μm and an acceleration resolution of 5 mm/s^2.

The digital to analogue converter card employs four 12 bit converters to drive four electromagnet current controllers. The schematic design of this card is illustrated Figure 6.4. In addition, a parallel input port is used to monitor a panel mounted switch. This card also carries the system watchdog timer which can disable the electromagnet

current controllers. If the watchdog is not triggered at least once every 10 ms, all connected current controllers will shut down immediately. This feature helps to provide fail-safe operation and a controlled system startup. The card also includes a 40 kHz clock generator which is used to synchronise the PWM switching of the electromagnet current controllers. An optical interface is provided on one of the transputer's links to provide electrical isolation for the vehicle chassis containing the DAC card and the electromagnet current controllers.

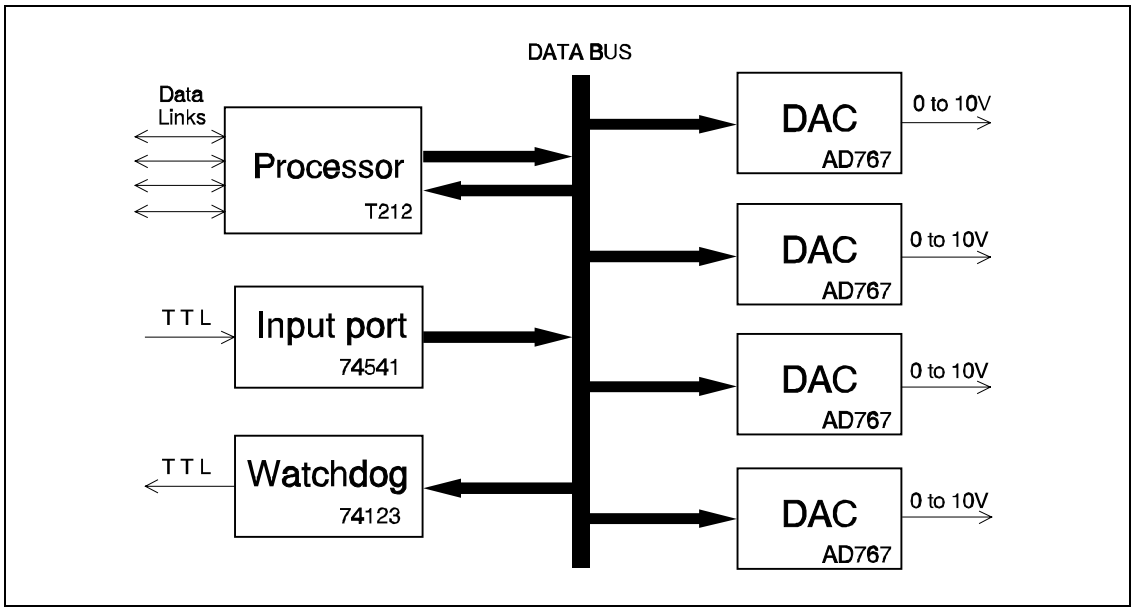

Figure 6.4 Schematic diagram of the digital to analogue converter.

The 12 bit digital converters are configured for a unipolar, full scale deflection of 10 V which produces an output quantisation level of 2.5 mV, and gives a current resolution of 5 mA. The conversion resolutions for the air gap, acceleration and current signals are thus at least twice the values specified in Table 6.2 for the end-to-end system signal resolutions.

6.4.4 Digital signal processing and communication

The TRAM motherboard carries commercial plug-in transputer modules which perform the signal processing functions. The ADC and DAC cards are provided with 16 bit integer processors in order to facilitate intelligent analogue interfaces, rather than to perform signal processing. Figure 6.5 illustrates the schematic design of the TRAM motherboard. It simply connects four TRAM sites in a pipelined configuration, and

provides buffered interfaces between the processor control signals and the chassis backplane.

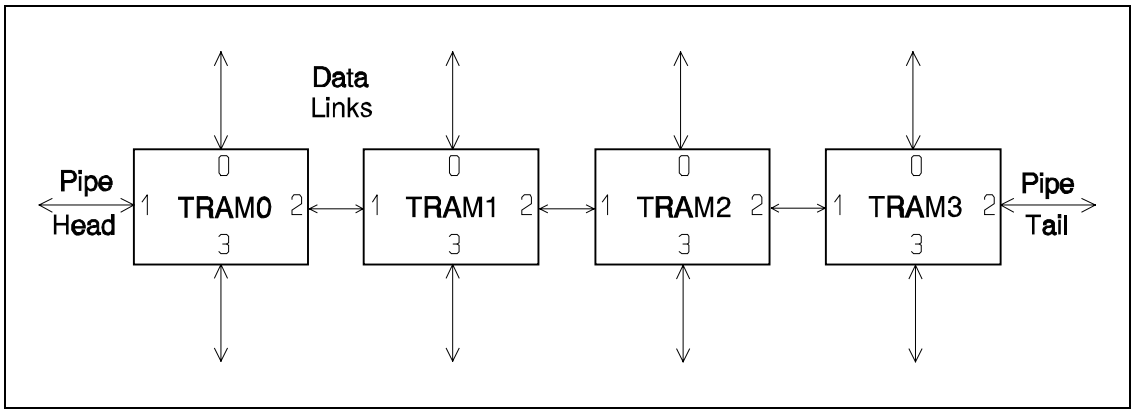

Figure 6.5 Schematic diagram of the TRAM motherboard.

To provide optical fibre connections between the two chassis and the PC, a pair of optical interface cards are used. One provides a chassis input interface, and the other is configured as a chassis output interface. Each card provides optical interfaces for one bi-directional transputer data link, and the processor reset, error and analyse control signals.

6.5 Software design

The design of the control system software is described in five main parts. The reasons for using occam are explained first, and some salient features of the language are outlined. The high-level structure of the control system software is then described. Next, a method for converting the continuous-time algorithms to the discrete time domain is selected. The level of numerical precision required for the signal processing is then calculated. Finally, the configuration of the constituent processes of the control system software onto the hardware is described.

6.5.1 Occam

The occam programming language[108] is a high level language, designed to express the sequential and concurrent components of algorithms, and their configuration on a network of processors. Since the transputer has a built-in multi-tasking scheduler, parallel algorithms can be run on a single processor as well as being distributed over a network of processors. The strength of this facility is that a parallel algorithm can

be directly coded into a parallel program, with little regard to the processor configuration. The program can then be configured to execute on one or more processors, parallelism and communication permitting.

Occam has been used for this work for three main reasons. Firstly, the use of concurrent processing hardware introduces high level concurrency within the software whilst control systems typically contain low level concurrency as well.[109] Occam is a rare example of a programming language specifically designed to implement concurrency at all structural levels in a natural and efficient manner. Secondly, when using concurrent algorithms, occam can provide a degree of security unknown in conventional sequential programming languages. Finally, the transputer reflects the occam structural model and may be considered an occam machine. Programming in occam is thus almost as efficient as using assembly language on conventional processors.

The features which differentiate occam from conventional languages are outlined in Appendix E prior to the listings of the control system software programs.

6.5.2 Control system software structure

The structure of the control system software has been designed using a combination of functional decomposition as advocated by Wirth,[110] and functional partitioning to minimise the data flow between processes as proposed by DeMarco.[111] Figure 6.6 illustrates the structure of the system in terms of tasks which are connected via channels providing synchronised communication of data signals, control messages and exception reports. The real-time data flows through the input signal interface, the state calculators, the control algorithms and the output signal interface. These components are supported by the sample scheduler, the exception handler, the data monitor and the user interface.

Table 6.4 gives a functional overview of the software tasks for the vehicle control system. The specification for the single electromagnet suspension is the same except for the use of only one electromagnet rather than four.

The control algorithms block depicted in Figure 6.6 is partitioned into two sub-blocks, which perform the suspension mode force calculation and the electromagnet current calculation. Figure 6.7 illustrates this arrangement for the single electromagnet

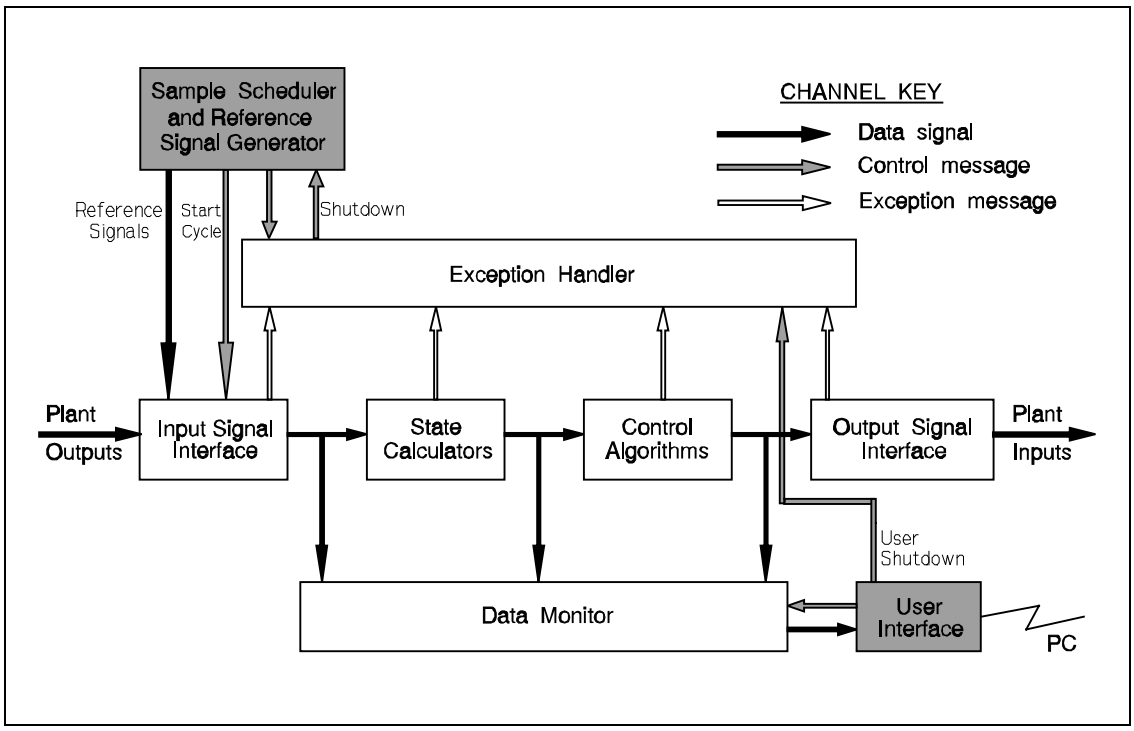

Figure 6.6 Control system task structure, and data and control flows.

suspension system. The air gap and acceleration signals are supplied by the input signal interface and the current demand is fed to the output signal interface. The parallelism within the state calculation and the control blocks is fine grained, and is therefore not well suited to parallel processing using transputers.

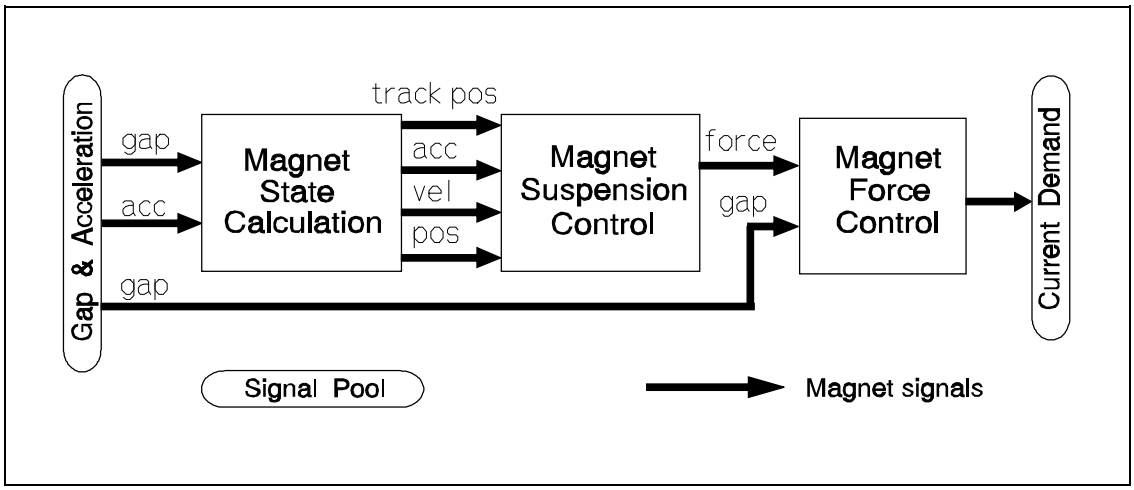

Figure 6.7 Single electromagnet controller data flow

The vehicle suspension system illustrated in Figure 6.8 uses four independent suspension controllers, one for each of the vehicle motions, heave, pitch, roll and torsion. Each suspension controller, plus its associated transformation and state calculation, involves a significant amount of computation. This results in a

Table 6.4 Vehicle control software functional overview

Input tasks:
- Read in acceleration and air gap sensor signals using ADC.
- Scale signals, check and report errors.

State calculation:
- Calculate velocities and positions from acceleration signals (Chapter 4).
- Calculate track position from magnet positions and air gaps (Chapter 4).

Control tasks:
- Apply the vehicle suspension control algorithm (Chapter 5) and magnet force control algorithm (Chapter 3) to the input and calculated data sets to produce a set of current demands for output. Check signals and report errors.
- Send user selected data to PC via asynchronous monitor process.

Output tasks:
- Scale current demand signals and output to the DACs.
- Check signals and report errors.
- Reset watchdog timer if the system is operational.

Miscellaneous:
- Permit user modification of controller parameters (including sample rate).
- Provide user selectable test reference signals (d.c., sine wave, square wave).
- Generate smooth startup and shutdown under normal and error conditions.

computational granularity which is sufficiently large to utilise parallel processing effectively using one transputer per suspension controller. After the vehicle motion force demands have been calculated, they are transformed to electromagnet force demands which are fed to the electromagnet controllers. The four independent electromagnet force controllers involve a level of computation similar to that of the vehicle suspension controllers, and so parallel processing using one transputer per controller is again efficient.

For the sake of convenience in terms of monitoring and comparing vehicle mode signals, the input and output decoupling transformations also normalise the signal amplitudes for the vehicle pitch, roll and torsion modes. This normalisation is described in Chapter 5, and it converts the angular signals and torques to the equivalent linear signals and forces at the electromagnet centres. Normalising the signals also permits a single data monitoring configuration to handle signals for suspension control algorithms using different decoupling transformations.

The sequencing of the operations of the vehicle control system can be summarised in terms of the following eight major phases:

1. Delivery of all magnet air gap and acceleration signals to each vehicle section.
2. Transformation of magnet signals to vehicle mode motions.
3. Calculation of unmeasured vehicle motion states.
4. Computation of each vehicle mode force demand.
5. Distribution of all vehicle mode forces to each magnet control section.
6. Transformation of vehicle mode force demands to magnet force demands.
7. Computation of each magnet current demand.
8. Delivery of each magnet current demand to the current demand signal pool.

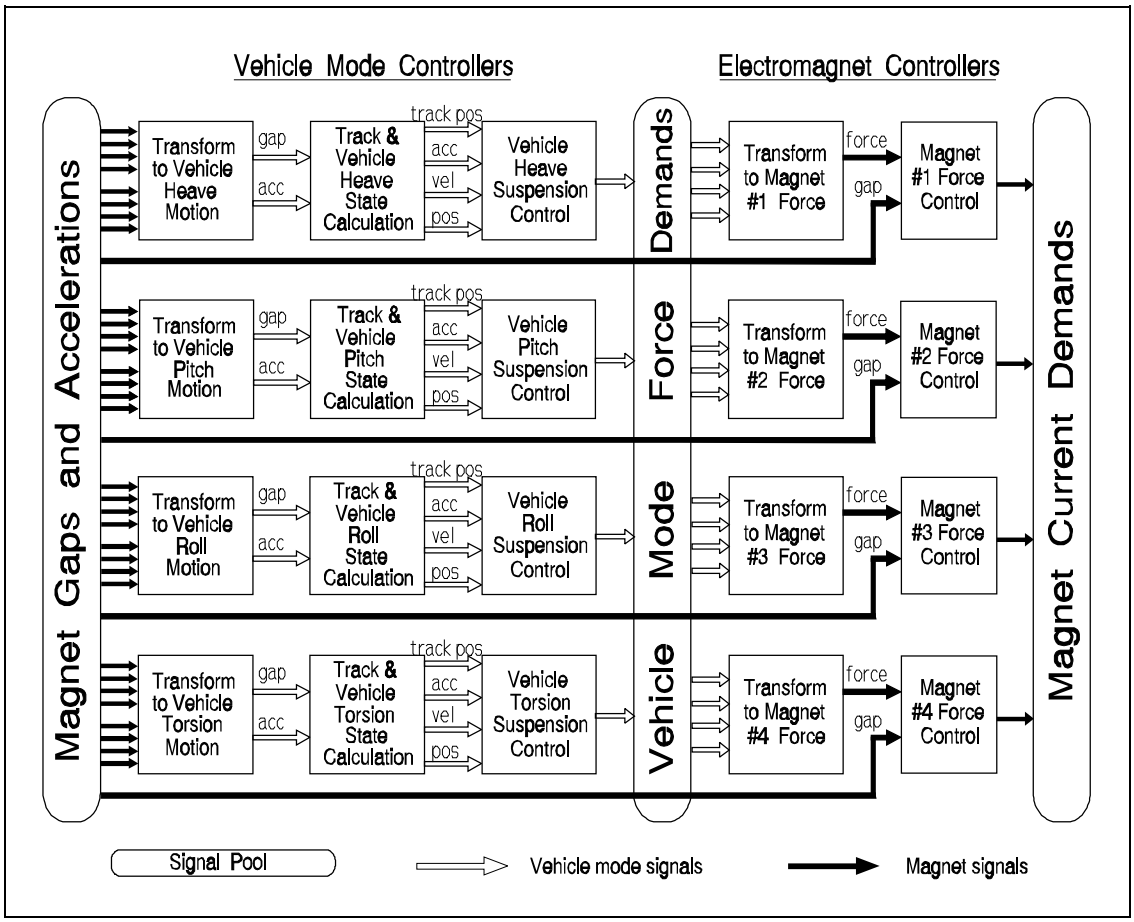

Figure 6.8 Vehicle suspension controller data flow

These phases occur irrespective of the number and configuration of processors used to run the control system. For a balanced processor load distribution, either 1, 2 or 4 processors can be used, each computing either 4, 2 or 1 vehicle motion sections followed by the same number of electromagnet control sections. The number of processors required depends on the individual processor power, the algorithm complexity and the required speed of execution. The multi-processing overheads for

the transputer, including the inter-processor communication overheads are small which gives a high multi-processor utilisation.

6.5.3 Discrete time domain integration

The control algorithms developed in Chapters 3, 4 and 5 have been designed in the continuous time domain. Therefore, they must be converted to discrete time domain representations before they can be implemented in software. The selection of a suitable technique for implementing discrete-time integrators and filters is considered next.

Filters are used by the state calculation algorithm, the suspension mode control algorithm and the electromagnet force control algorithm. The first two algorithms use low pass and high pass filters, with time constants ranging from 1.6 s to 40 ms, whilst the electromagnet force control algorithm uses a phase lead compensator with a pole time constant of 1.5 ms. The time constant of the phase lead compensator pole is thus close to the required control system sampling interval of 0.5-1.0 ms. The discrete-time representation is therefore a fairly crude approximation to the continuous-time response. However, the effect of this misrepresentation is to increase the effective phase lead slightly which is not detrimental to the location of the closed-loop system poles. The simulation model listed in Appendix C was used to verify this assumption. The high pass filter and phase lead compensator are formulated in terms of a first-order low pass filter as described in Table 6.5.

Table 6.5 Filter formulations

$$LowPass(s,T) = \frac{1}{1 + sT}$$

$$HighPass(s,T) = \frac{sT}{1 + sT} = 1 - LowPass(s,T)$$

$$PhaseLead(s,T,N) = \frac{1 + NsT}{1 + sT} = N - (N-1)\,LowPass(s,T)$$

Various integration algorithms are available for approximating the continuous time domain in the discrete time domain and each method has its own merits.[112] A primary requirement for this development is that the approximation algorithm must be cascadable so that each independent functional block within the control system can be designed and implemented independently. The algorithm must also produce an accurate

d.c. gain, and map stable continuous-time poles to stable discrete-time poles. These requirements are effectively met by the Tustin algorithm (trapezoidal integration) and by the first difference algorithm (Euler integration). Since all the filters except for the phase lead compensator have time constants which are orders of magnitude greater than the sample interval, the simpler first difference algorithm is sufficiently accurate. The lower quality approximation that results for the phase lead compensator is acceptable.

Prewarping[113] of the filter time constants is unnecessary for the normal control system parameter settings since the time constants are much larger than the sampling interval, and so their pole placement accuracy is normally very good. However, frequency prewarping is required for the phase lead compensator time constant since it is so small. It is also employed for the other time constants to allow for experimentation with extreme settings for the filter frequencies and the sampling interval. Table 6.6 lists the discrete time algorithms which are used for integration and low-pass filtering, and the prewarping correction factor.

Table 6.6 Discrete-time integrator and filter implementation

Integrator:	$y_k = y_{k-1} + x_k \, T_{sample}$
Low pass filter:	$y_k = y_{k-1} \left(1 - \frac{T_{sample}}{T_{cutoff}} \right) + x_k \frac{T_{sample}}{T_{cutoff}}$
Prewarp correction:	$\left[\frac{T_{sample}}{T_{cutoff}} \right]_{corrected} \approx 1 - \exp\left(-\frac{T_{sample}}{T_{cutoff}} \right)$

6.5.4 Numerical accuracy

Having determined the discrete time integration algorithm, the level of numerical accuracy required to implement the control algorithms is now calculated. A floating-point representation is assumed for all operations in order to have precision independent of scaling. This eliminates the implementational overhead associated with the use of fixed-point representations where signals must be re-scaled as necessary to maintain sufficient precision.

The resolution required for the transducer signals calls for the use of 12 bit analogue-to-digital and digital-to-analogue converter hardware. The calculation of each current demand signal involves a total of about 80 mathematical operations, and in general, a 32 bit floating-point representation (which has a 23 bit mantissa) provides sufficient numerical accuracy for 12 bit data. However, loss of accuracy can occur where two values which differ by many orders of magnitude are added or subtracted.

The worst case of this behaviour occurs in the state integration filters for velocity and position, where the required sample interval of 0.5-1.0 ms can result in very small values being summed onto the very much larger value of the state integrator.[114]

Table 6.7 shows the calculation of the relative size of the input and integrator values for the velocity and position state calculators. The sampling interval, T_{sample}, is assumed to be 0.5 ms, and the integration filter corner frequency of 0.1 Hz gives a filter time constant, T_{filter}, of 1.6 s. The decay product term is neglected for this analysis since it is approximately unity. Acc_{min} is set to half the required quantisation amplitude for the acceleration measurement and vel_{min} represents the corresponding velocity quantisation. Vel_{max} and pos_{max} represent the maximum practical values that the velocity and position signals can have.

Table 6.7 Signal magnitude calculation for the state integration filter

Euler integration for velocity integration filter (worst case scenario) gives:

$$\begin{aligned} vel &= vel_{max} + acc_{min} \cdot T_{sample} / T_{filter} \\ &= 10^{-1}\ m/s + 5x10^{-2}\ m/s^2 \cdot 5x10^{-4}\ s / 1.6\ s \\ &= 10^{-1}\ m/s + 1.6x10^{-5}\ m/s \end{aligned}$$

Relative size: $10^{-1}/1.6x10^{-5} = 6.4x10^{3}$

Euler integration for position integration filter (worst case scenario) gives:

$$\begin{aligned} pos &= pos_{max} + vel_{min} \cdot T_{sample} / T_{filter} \\ &= 5x10^{-3}\ m + 1.6x10^{-5}\ m/s \cdot 5x10^{-4}\ s / 1.6\ s \\ &= 5x10^{-3}\ m + 5x10^{-9}\ m \end{aligned}$$

Relative size: $5x10^{-3}/5x10^{-9} = 10^{6}$

The position integration filter provides the most exacting requirement since the maximum accumulated position signal can be 10^6 times larger than the minimum added value. A 32 bit floating-point representation with a 23 bit mantissa (plus the sign bit)

has a precision of just under 7 decimal digits. Under the worst case scenario of maximum position signal and minimum velocity, the input value would be summed into the integrator with a precision of less than one decimal digit. This would cause severe pole placement inaccuracy, and a significant loss of the summed signal accuracy. The state integration filters therefore use a 64 bit floating-point representation which has a 52 bit mantissa giving a precision of about 15.5 decimal digits.

Table 6.8 Signal magnitude calculations for the guideway following filter

Euler integration for guideway filter (worst case scenario) gives:

$$\begin{aligned} trk &= trk_{max} + gap_{min} \cdot T_{sample} / T_{filter} \\ &= 5\text{x}10^{-3}\ m + 5\text{x}10^{-6}\ m \cdot 5\text{x}10^{-4}\ s / 4\text{x}10^{-2}\ s \\ &= 5\text{x}10^{-3}\ m + 6.25\text{x}10^{-8}\ m \end{aligned}$$

Relative size: $5\text{x}10^{-3}/6.25\text{x}10^{-8} = 8\text{x}10^{4}$

And similarly for the second (cascaded) guideway filter gives:

Relative size: $5\text{x}10^{-3}/7.8\text{x}10^{-10} = 6.4\text{x}10^{6}$

A similar argument applies to the guideway following filter which consists of two cascaded low-pass filters. Table 6.8 shows the magnitude calculations for the guideway filter where gap_{min} is set to half the required air gap signal quantisation and trk_{max} is the maximum air gap measurement. Once again, the large relative size between the two filter terms, almost 7 decimal digits, requires the use of 64 bit floating-point numbers for the guideway following filters.

The fast time constant used by the phase lead compensator used in the magnet force control algorithm results in 32 bit floating-point accuracy being sufficient. The remaining computation for the mode and force transformations and the suspension and electromagnet control algorithms can all be satisfactorily performed using 32 bit floating-point arithmetic.

6.5.5 Process configuration

Having determined the requirements for the control system software, the final stage is to partition the software tasks onto a suitable transputer configuration. For the single

electromagnet suspension controller, this is simple since a single 20 Mhz T800 processor is sufficient to provide a minimum sampling interval of 0.8 ms. Experimental responses showed this sampling rate to be more than adequate. Responses were therefore obtained with larger sampling intervals, and they showed that the response started to become unacceptable for values larger than about 2 ms.

For the vehicle control system software, the execution time for all of the tasks identified in Figure 6.8 was measured for a single 25 MHz T800 processor. This achieved a minimum sampling interval of about 2.3 ms, compared with the target range of 0.5-1.0 ms, and experimental responses showed it to be unsatisfactory. It was estimated that the use of two processors would reduce the sample interval to about 1.2 ms, whilst employing four processors would reduce it down to about 0.65 ms.

For economic reasons, a two processor configuration was investigated. This configuration executed with a measured sampling interval of 1.25 ms, and good experimental responses were obtained. Figure 6.9 shows how the software tasks are partitioned between the two processors. This configuration and sampling interval is used for all of the experimental vehicle responses presented in this dissertation. The lower processing load on the signal processor without the data monitor and its associated real-time buffer permits the use of a 20 Mhz processor.

A fully controlled vehicle with four suspension electromagnets and four guidance electromagnets could therefore be controlled by four transputers, since the additional overheads are minimal. The code listings for all of the software described in this chapter are listed in Appendix E.

6.6 Conclusions

This chapter has shown that the electromagnetic vehicle suspension strategy developed in the earlier chapters can be implemented using readily available parallel processing hardware. The use of transputers for the signal processing and intelligent converter interfaces enabled a highly modular and inexpensive real-time processing platform to be constructed. This was complemented by using the occam programming language which enabled a fairly complex software implementation to be developed very rapidly.

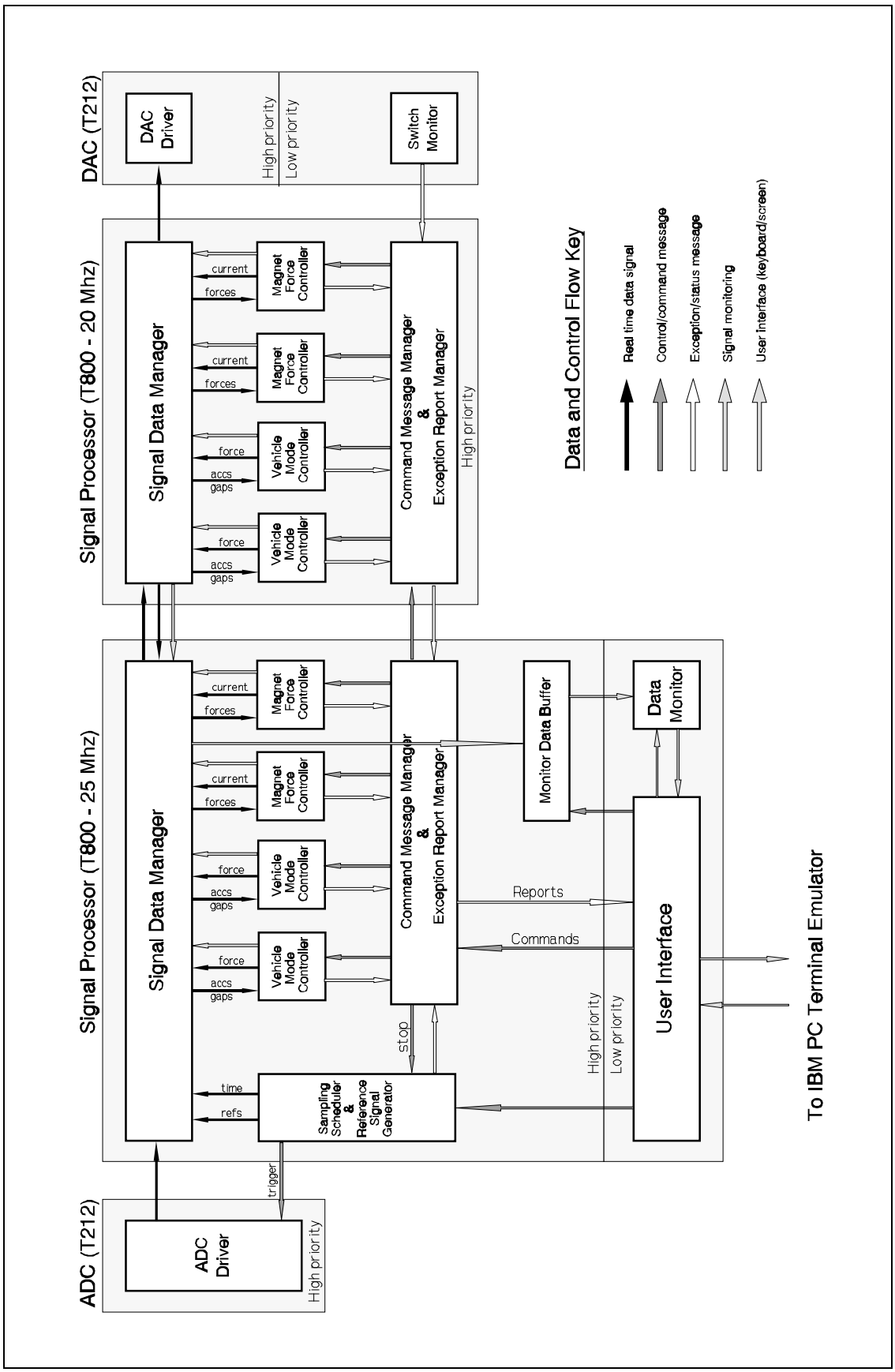

Figure 6.9 Vehicle control system process configuration