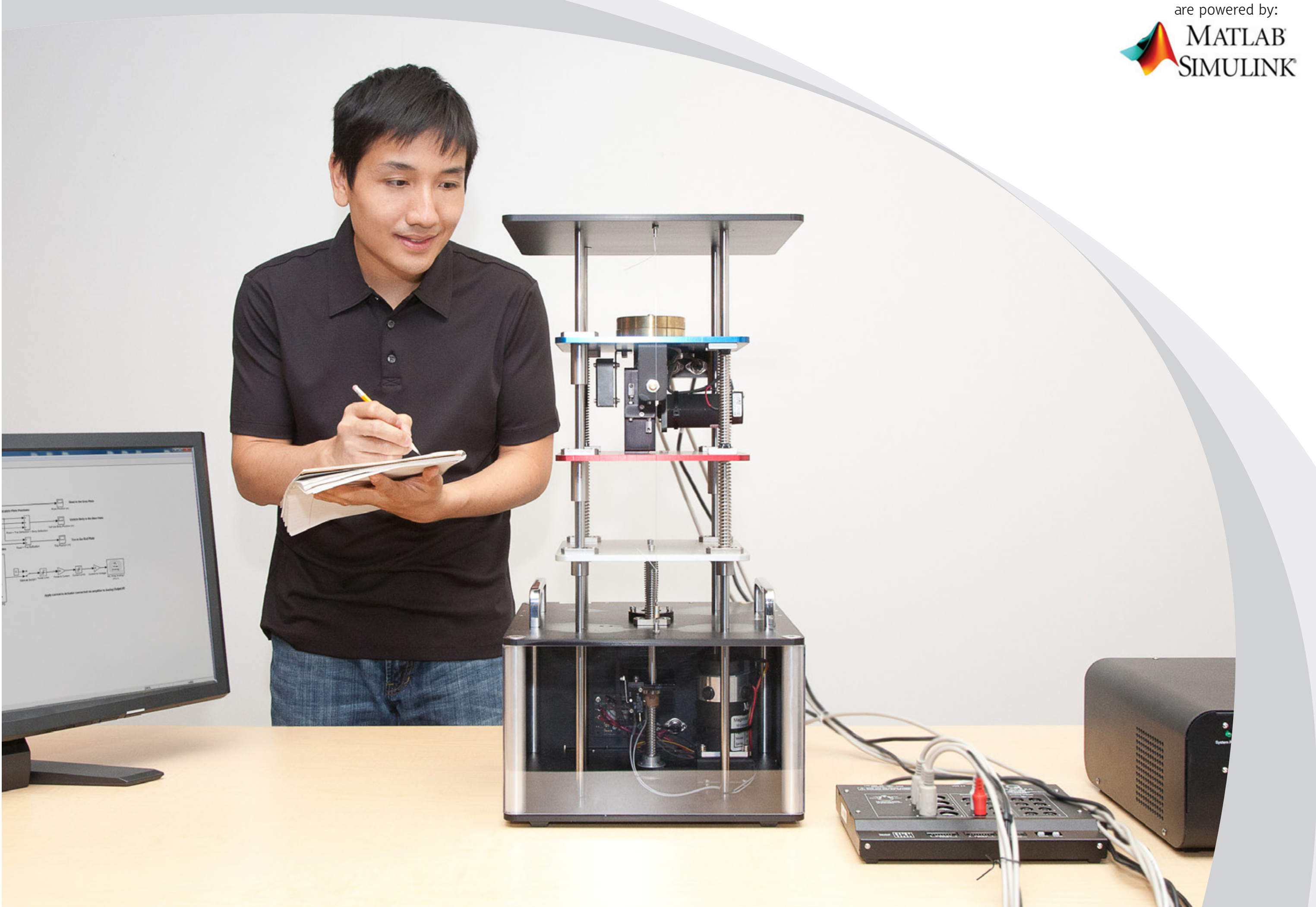
QUANSER
INNOVATE. EDUCATE.

# LABORATORY GUIDE

## Active Suspension Experiment for MATLAB®/Simulink® Users

Developed by:
Jacob Apkarian, Ph.D., Quanser
Amin Abdossalami, M.A.SC., Quanser

Quanser educational solutions are powered by:
MATLAB SIMULINK



CAPTIVATE. MOTIVATE. GRADUATE.

Quanser Inc.
119 Spy Court
Markham, Ontario
L3R 5H6
Canada
info@quanser.com
Phone: 1-905-940-3575
Fax: 1-905-940-3576

Printed in Markham, Ontario.

For more information on the solutions Quanser Inc. offers, please visit the web site at:
http://www.quanser.com

# CONTENTS

# 1 INTRODUCTION

## 1.1 Description

The purpose of the Active Suspension is to design and implement a state-feedback controller for a quarter-car model. This system consists of two masses, each supported by a spring and a damper. The *sprung mass*, $M_s$, represents the mass of the vehicle body while the *unsprung mass*, $M_{us}$, represents the tire in the quarter-car model. This system is fourth order because there are four independent storage elements, the two masses and the two springs. The spring $K_s$ and the damper $B_s$ support the body weight over the tire. The spring $K_{us}$ and the damper $B_{us}$ model the stiffness of the tire in contact with the road. An LQR controller can be used to optimize a variety of performance parameters in the quarter-car model. In this approach, the performance criteria are formulated into a mathematical model. This mathematical representation is then optimized while considering the control actuator limitations. The performance measures that have to be minimized are listed below:

- *Ride Comfort* is related to vehicle body motion sensed by the passengers. A measure for the Ride Comfort is the acceleration of the sprung mass in the quarter car model.
- *Suspension Travel* refers to relative displacement between the vehicle body and the tire and is constrained within an allowable workspace. In the quarter car model, relative displacement between the sprung mass and the unsprung mass represents Suspension Travel.
- *Road Handling* is associated with the contact forces between the road surface and the vehicle tires. These forces provide the necessary friction between the road and the tires in a real car. The contact forces between the road and the tires depend on the *tire deflection*. In a quarter car model, relative displacement between the unsprung mass and the road represents the tire deflection.

## 1.2 Topics Covered

The following topics are covered in this manual:

- How to mathematically model the Active Suspension plant, using, for example, force analysis on free body diagrams.
- How to obtain a state-space representation of the open-loop system and to do open-loop analysis
- How to obtain different transfer functions for the Active Suspension Experiment as a MIMO system.
- How to use the obtained Active Suspension state-space representation to design a Linear Quadratic Regulator (LQR).
- To simulate the Linear Quadratic Regulator (LQR) controller using the developed model of the plant and to ensure the controller performance specifications are met without any actuator saturation.
- To implement an LQR-based state-feedback controller in real-time and evaluate its actual performance.
- To observe and investigate the disturbance response of the suspension system in response to chirp and pulse shape road disturbances.

# 2 BACKGROUND

## 2.1 Modeling

### 2.1.1 Dynamics

In this section, the general dynamic equations of the Active Suspension System will be derived. The Free Body Diagram method is used to obtain the dynamics of the system as a double mass-spring damper model. This diagram is illustrated in Figure 2.1. In this approach, the two inputs to the system are considered to be active suspension control command $F_c$ and the road surface position $z_r$. Furthermore, it is reminded that the reference frames in Figure 2.1 are used to choose the generalized coordinates, i.e. $x_1$ and $x_2$. The generalized coordinate $x_1$ represents the tire displacement (usnprung mass in quarter car model) and $x_2$ represents the vehicle body displacement (sprung mass in the quarter car model) all with respect to the ground. The positive directions are upwards.

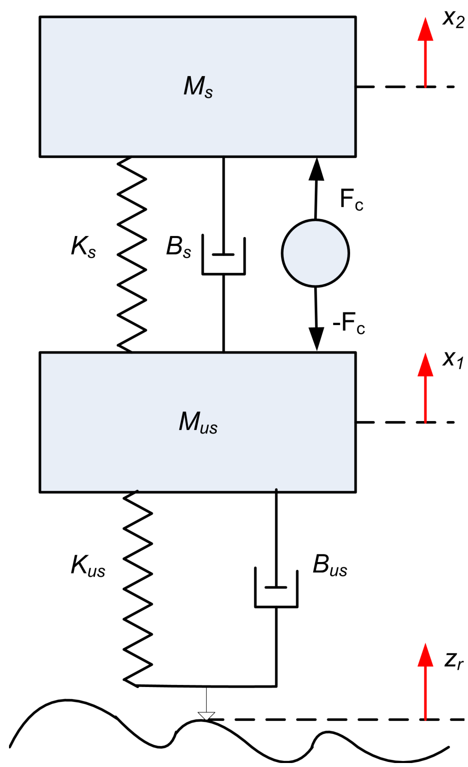



Figure 2.1: Double Mass-Spring-Damper used to model Active Suspension

To find out equations of motion (EOM) for this system, The free body diagram for each mass should be determined. There are two masses in the system and the forces applied to each mass should be drawn on the diagrams. There will be two equations of motion. All the initial conditions are assumed to be zero. The free body diagram for $M_s$ looks like Figure 2.2. The forces applied to the $M_s$ are due to the spring force, damping force, active suspension force, and gravity.

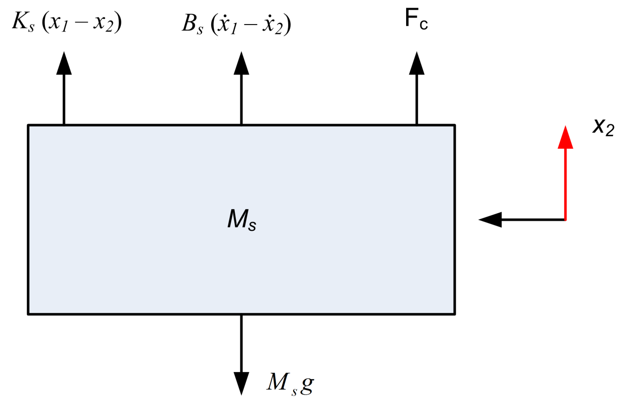



Figure 2.2: The free body diagram for $M_s$

The EOM for $M_s$ will be as follows

$$\ddot{x}_2 = -g + \frac{F_c}{M_s} + \frac{B_s\dot{x}_1}{M_s} - \frac{B_s\dot{x}_2}{M_s} + \frac{K_s x_1}{M_s} - \frac{K_s x_2}{M_s} \tag{2.1}$$

The free body diagram for $M_{us}$ looks like Figure 2.3. The forces applied to the $M_{us}$ are the springs forces, damping forces, active suspension force, and gravity.

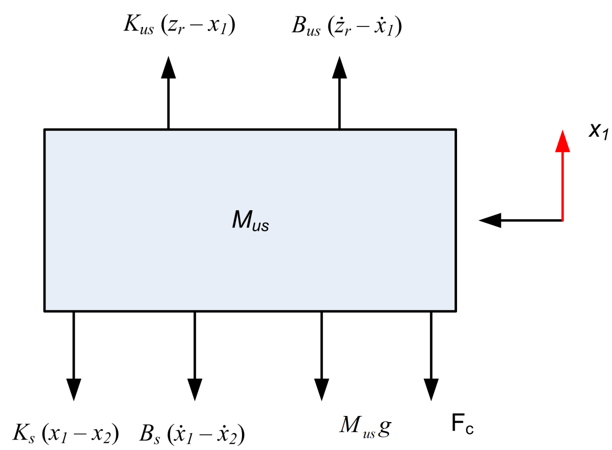



Figure 2.3: The free body diagram for $M_{us}$

The EOM for $M_{us}$ can be derived as follows

$$\begin{aligned}\ddot{x}_1 = &-g - \frac{F_c}{M_{us}} - \frac{(B_s + B_{us})\,\dot{x}_1}{M_{us}} + \frac{B_s\dot{x}_2}{M_{us}} + \frac{B_{us}\dot{z}_r}{M_{us}} \\ &- \frac{(K_{us} + K_s)\,x_1}{M_{us}} + \frac{K_s x_2}{M_{us}} + \frac{z_r K_{us}}{M_{us}}\end{aligned} \tag{2.2}$$

### 2.1.2 Eliminating Gravity Force from EOM

The objective of this section is to prove mathematically that the gravity force only changes the equilibrium points in the Active Suspension EOM and it does not affect the dynamics of the system. At the static equilibrium point, i.e. $x_1 = x_{eq1}, x_2 = x_{eq2}$, all the derivatives of $x_1$ and $x_2$ of any order are zero. Also, the road surface $z_r$ and all its derivatives and the control force $F_c$ are zero. Substituting these changes in equations 2.1 and 2.2 will give the following results

$$
\begin{aligned}
K_s x_{eq1} - K_s x_{eq2} + M_{us} g + x_{eq1} K_{us} &= 0 \\
M_s g + K_s x_{eq2} - K_s x_{eq1} &= 0
\end{aligned}
\tag{2.3}
$$

As a result, the equilibrium points due to gravity will be

$$x_{eq1} = -\frac{g\,(M_s + M_{us})}{K_{us}} \tag{2.4}$$

$$x_{eq2} = -\frac{g\,(M_s K_{us} + K_s M_s + K_s M_{us})}{K_{us} K_s} \tag{2.5}$$

In order to remove gravity forces from the equations of motion we apply the following change of variables to the equations of motion:

$$
\begin{aligned}
x_1 = z_{us} - \frac{g\,(M_s + M_{us})}{K_{us}}, &\qquad x_2 = z_s - \frac{M_s g}{K_s} - \frac{g\,(M_s + M_{us})}{K_{us}} \\
\dot{x}_1 = \dot{z}_u, &\qquad \dot{x}_2 = \dot{z}_s \\
\ddot{x}_1 = \ddot{z}_{us}, &\qquad \ddot{x}_2 = \ddot{z}_s
\end{aligned}
\tag{2.6}
$$

By substituting the variable changes in 2.6 into Active Suspension EOM we get the following equations where the effect of gravity has been eliminated from the equations. In other words, the relaxed position of the springs, i.e. $z_{us} = 0$, $z_s = 0$ , will be the equilibrium point of the system.

$$M_{us}\ddot{z}_{us} = -B_s\dot{z}_{us} - B_{us}\dot{z}_{us} - F_c + B_s\dot{z}_s + B_{us}\dot{z}_r - (z_{us} - z_s)\,K_s - (z_{us} - z_r)\,K_{us} \tag{2.7}$$

$$M_s\ddot{z}_s = B_s\dot{z}_{us} + F_c - B_s\dot{z}_s - (z_s - z_{us})\,K_s \tag{2.8}$$

The only inputs to this system are the control force and the road surface.

### 2.1.3 State Space Representation

In order to design and implement a state-feedback controller for a system a state-space representation of that system needs to be derived. In this Section, a state space representation for the Active Suspension will be derived. It is reminded that state-space matrices, by definition, represent a set of linear differential equations that describe the system's dynamics. Since the two EOM of the Active Suspension system should already be linear and time-invariant, they can be written under the state-space representation as follows

$$
\begin{aligned}
\dot{x} &= Ax + Bu \\
y &= Cx + Du
\end{aligned}
\tag{2.9}
$$

The state space approach is a convenient way to model the quarter-car model with multiple inputs and outputs. The states can be defined such that they reflect the system performance parameters that have to be optimized. Moreover, this state space can be later used for state-feedback controller and observer design. Due to the existence of four energy storage elements in the quarter car model one should define four state for the system. The four state variables, the two inputs to the system, and the two outputs can be defined as follows:

$$x = \begin{bmatrix} z_s - z_{us} \\ \dot{z}_s \\ z_{us} - z_r \\ \dot{z}_{us} \end{bmatrix}, u = \begin{bmatrix} \dot{z}_r \\ F_c \end{bmatrix}, y = \begin{bmatrix} z_s - z_{us} \\ \ddot{z}_s \end{bmatrix} \tag{2.10}$$

Where the first state represents suspension deflection/travel. The second state is the vehicle body vertical velocity. The third state is the tire deflection which is a measure of road handling. The fourth state is the tire vertical velocity. The first input to the system is the road surface velocity. The second input is the control action that will be later designed. The first measured output of the system is the suspension travel. Assuming that the vehicle body is equipped with an accelerometer, the second measured output of the system will be the body acceleration. Using the equations of motions defined in one can calculate the matrices $A$, $B$, $C$, and $D$ as follows:

$$\begin{aligned} A &= \begin{bmatrix} 0 & 1 & 0 & -1 \\ -\frac{K_s}{M_s} & -\frac{B_s}{M_s} & 0 & \frac{B_s}{M_s} \\ 0 & 0 & 0 & 1 \\ \frac{K_s}{M_{us}} & \frac{B_s}{M_{us}} & -\frac{K_{us}}{M_{us}} & -\frac{B_s+B_{us}}{M_{us}} \end{bmatrix}, B = \begin{bmatrix} 0 & 0 \\ 0 & \frac{1}{M_s} \\ -1 & 0 \\ \frac{B_{us}}{M_{us}} & -\frac{1}{M_{us}} \end{bmatrix} \\ C &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ -\frac{K_s}{M_s} & -\frac{B_s}{M_s} & 0 & \frac{B_s}{M_s} \end{bmatrix}, D = \begin{bmatrix} 0 & 0 \\ 0 & \frac{1}{M_s} \end{bmatrix} \end{aligned} \tag{2.11}$$

### 2.1.4 System Transfer Functions

The state space representation that was derived in last section is an example of a multi-input and multi-output system (MIMO) with two inputs, i.e., road surface velocity and controller force Fc , and two outputs, i.e., suspension travel and body acceleration. The characteristic equation of the open-loop system can be expressed as shown below:

$$det(sI - A) = 0 \tag{2.12}$$

where $det()$ is the determinant function, $s$ is the Laplace operator, and $I$ is the identity matrix. Therefore, the system's open-loop poles can be seen as the eigenvalues of the state-space matrix A. The characteristic equation of the Active Suspension is as follows:

$$\begin{aligned} \det(sI - A) = \frac{1}{M_s M_{us}} (&s^4 M_s M_{us} + s^3 M_s B_s + s^3 M_s B_{us} + s^2 M_s K_{us} + B_s s^3 M_{us} \\ &+ B_s s^2 B_{us} + s B_s K_{us} + K_s s^2 M_{us} + K_s s B_{us} + K_s K_{us} + K_s s^2 M_s) \end{aligned} \tag{2.13}$$

The transfer functions for a MIMO system can be calculated as follows based on the system state space representation:

$$C(sI - A)^{-1} B + D \tag{2.14}$$

Below, the transfer functions for all the combinations of the two inputs and two outputs of Active Suspension are derived. The operator $\mathscr{L}$ is the Laplace operator.

$$\mathscr{L}\left(\frac{z_s - z_{us}}{\dot{z}_r}\right) = - sM_s\left(K_{us} + sB_{us}\right)/(s^4 M_s M_{us} + \left(M_s B_{us} + B_s M_{us} + M_s B_s\right)$$
$$s^3 + \left(M_s K_{us} + B_s B_{us} + K_s M_s + K_s M_{us}\right) s^2 + \left(K_s B_{us} + B_s K_{us}\right) s + K_s K_{us}) \tag{2.15}$$

$$\mathscr{L}\left(\frac{z_s - z_{us}}{F_c}\right) = \left(\left(M_{us} + M_s\right) s^2 + sB_{us} + K_{us}\right) /(s^4 M_s M_{us} + \left(M_s B_{us} + B_s M_{us} + M_s B_s\right)$$
$$s^3 + \left(M_s K_{us} + B_s B_{us} + K_s M_s + K_s M_{us}\right) s^2 + \left(K_s B_{us} + B_s K_{us}\right) s + K_s K_{us}) \tag{2.16}$$

$$\mathscr{L}\left(\frac{\ddot{z}_s}{\dot{z}_r}\right) = s\left(B_s s^2 B_{us} + \left(K_s B_{us} + B_s K_{us}\right) s + K_s K_{us}\right) /(s^4 M_s M_{us} + \left(M_s B_{us} + B_s M_{us} + M_s B_s\right)$$
$$s^3 + \left(M_s K_{us} + B_s B_{us} + K_s M_s + K_s M_{us}\right) s^2 + \left(K_s B_{us} + B_s K_{us}\right) s + K_s K_{us}) \tag{2.17}$$

$$\mathscr{L}\left(\frac{\ddot{z}_s}{F_c}\right) = s^2\left(s^2 M_{us} + sB_{us} + K_{us}\right) /(s^4 M_s M_{us} + \left(M_s B_{us} + B_s M_{us} + M_s B_s\right)$$
$$s^3 + \left(M_s K_{us} + B_s B_{us} + K_s M_s + K_s M_{us}\right) s^2 + \left(K_s B_{us} + B_s K_{us}\right) s + K_s K_{us}) \tag{2.18}$$

## 2.2 Control

There is a high demand for better ride comfort and handling of road vehicles. No suspension system can fully minimize all the performance measures introduced in Section 1 simultaneously. However, in comparison to passive suspension system, an actively controlled suspension, i.e., Active Suspension, can be exploited to come up with a better possible trade-off between the performance measures. In this Section, you will investigate the effects of Linear Quadratic Regulator (LQR) control. In this systematic approach, performance parameters as well as actuator limitations will be quantified in a quadratic measure which is later optimized. The optimal gain can then be used in a state feedback controller where it is assumed that the states are measurable.

In Section 2.1.3, we found a linear state-state space model that represents the Active Suspension system. This model is used to investigate the stability properties of the system in Section 2.2.1. The controllability notion of the system is presented in Section 2.2.2, Finally, in Section 2.2.3 the LQR controller is presented which will be later used to design a state-feedback gain to stabilize the system.

### 2.2.1 Stability

The stability of a system can be determined from its poles ([5]):

- Stable systems have poles only in the left-hand plane.
- Unstable systems have at least one pole in the right-hand plane and/or poles of multiplicity greater than 1 on the imaginary axis.
- Marginally stable systems have one pole on the imaginary axis and the other poles in the left-hand plane.

The poles are the roots of the system's characteristic equation. From the state-space, the characteristic equation of the system can be found using

$$\det\left(sI - A\right) = 0 \tag{2.19}$$

where *det()* is the determinant function, $s$ is the Laplace operator, and $I$ the identity matrix. These are the *eigenvalues* of the state-space matrix $A$.

### 2.2.2 Controllability

If the control input, $u$, of a system can take each state variable, $x_i$ where $i = 1 \dots n$, from an initial state to a final state then the system is controllable, otherwise it is uncontrollable ([5]).

**Rank Test** The system is controllable if the rank of its controllability matrix

$$T = \left[B\ AB\ A^2B \dots A^nB\right] \tag{2.20}$$

equals the number of states in the system,

$$\text{rank}(T) = n. \tag{2.21}$$

### 2.2.3 Linear Quadratic Regulator (LQR)

If (A,B) are controllable, then the Linear Quadratic Regulator optimization method can be used to find a feedback control gain. Given the plant model in Equation 2.23, find a control input $F_c$ that minimizes the cost function

$$J = \int_0^\infty x(t)'Qx(t) + RF_c(t)^2\, dt, \tag{2.22}$$

Where $x(t)$ contains the actual states of the system defined in Equation 2.10. The performance index $J$ penalizes the states of the system, i.e., suspension travel and tire deflection as the two performance measures, as well as the body velocity and tire velocity through the weighting matrix $Q$. It also reflects the control limitations by penalizing the control input through the weighting gain $R$. The weighting matrices affect how LQR minimizes the function and are, essentially, tuning variables.

In Active Suspension, the control action is $F_c$. Thus, the corresponding $A$ matrix and $B$ vector that correspond to this control action are as follows:

$$A = \begin{bmatrix} 0 & 1 & 0 & -1 \\ -\frac{K_s}{M_s} & -\frac{B_s}{M_s} & 0 & \frac{B_s}{M_s} \\ 0 & 0 & 0 & 1 \\ \frac{K_s}{M_{us}} & \frac{B_s}{M_{us}} & -\frac{K_{us}}{M_{us}} & -\frac{B_s+B_{us}}{M_{us}} \end{bmatrix}, B = \begin{bmatrix} 0 \\ \frac{1}{M_s} \\ 0 \\ -\frac{1}{M_{us}} \end{bmatrix} \tag{2.23}$$

This model was used to access the controlability of the system, and to derive appropriate feedback control gains.

The solution to the optimization problem in Equation 2.22 is $F_c = -Kx$. Given this control law, the state-space in Equation 2.23 becomes

$$\begin{aligned} \dot{x} &= Ax + B(-Kx) \\ &= (A - BK)x \end{aligned} \tag{2.24}$$

# 3 LAB EXPERIMENTS

## 3.1 Simulation

The state space representation of Active Suspension was derived in Equation 2.11. In this section, you will generate those equations and design a controller. The parameter values are outlined in the table below. These values have been derived using system identification techniques and they might not exactly match the nominal values presented in the Active Suspension User Manual.

| Parameter Symbol | Parameter Name | Parameter Value |
|---|---|---|
| $M_s$ | Sprung Mass | 2.45 kg |
| $M_{us}$ | Unsprung Mass | 1 kg |
| $K_s$ | Suspension Stiffness | 900 N/m |
| $K_{us}$ | Tire Stiffness | 1250 N/m |
| $B_s$ | Suspension Inherent Damping coefficient | 7.5 Nsec/m |
| $B_{us}$ | Tire Inherent Damping coefficient | 5 Nsec/m |

Table 3.1: Active Suspension Parameter Nomenclature

In this section we will use the Simulink diagram shown in Figure 3.1 to simulate the closed-loop control of the Active Suspension system. The system is simulated using the model summarized in Section 2.1. The Simulink model uses state-feedback control, with feedback gain $K$ found using the Matlab LQR command (LQR is described briefly in Section 2.2.3).

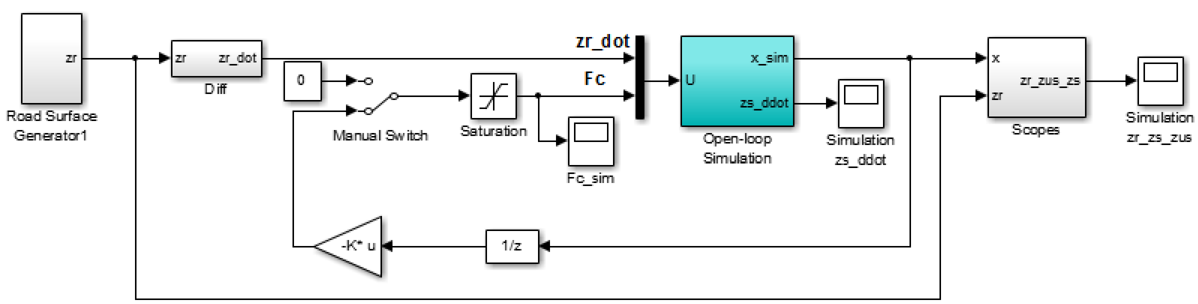



Figure 3.1: Simulink model used to simulate Active Suspension.

**IMPORTANT:** Before you can conduct these simulations and experiments, you need to make sure that the lab files are configured according to your setup. If they have not been configured already, then you need to go to Section 4 to configure the lab files first.

### 3.1.1 Procedure

Follow these steps to simulate the system:

1. Make sure the LQR weighting matrices in setup_as.m are set to

$$Q = \begin{bmatrix} 450 & 0 & 0 & 0 \\ 0 & 30 & 0 & 0 \\ 0 & 0 & 5 & 0 \\ 0 & 0 & 0 & 0.01 \end{bmatrix}$$

and $R = 0.01$.

2. Run the script to generate the gain

$$K = \begin{bmatrix} 24.66 & 48.87 & -0.47 & 3.68 \end{bmatrix}.$$

3. Open the plate position scope, *Simulation zr_zs_zus*.

4. The road input is a square shape signal with an amplitude of 0.01 m and frequency of 0.3 Hz

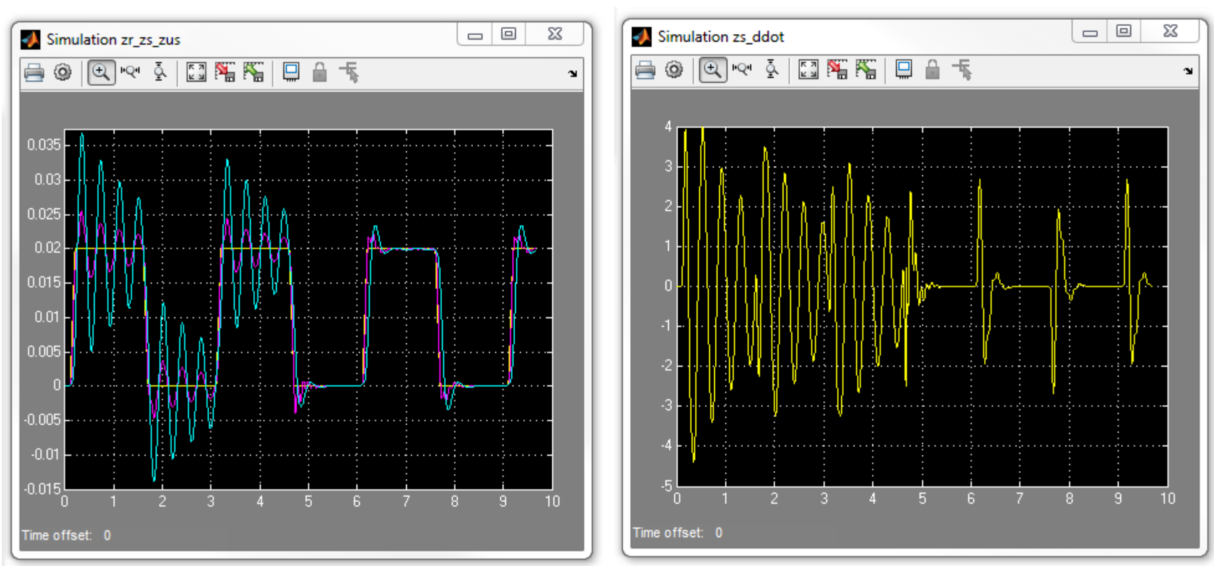
5. $Zr$ represents the bottom plate position which refers to the road. $Z_{us}$ represents the middle plate position which refers to vehicle tire. $Z_s$ represents the top plate position which refers to vehicle body.

6. In the Simulink diagram, go to QUARC | Build.

7. Click *Connect to Target* to connect to the real-time code, then Click on QUARC | Start to run the simulation.

8. The active damping control action can be enabled or disabled using the *Manual Switch* to observe both the controller performance and open loop response.

9. The scopes should be displaying a response similar to Figure 3.2. The closed loop controller is enabled 5 seconds into the response.




(a) Plate Positions

(b) Body Acceleration

Figure 3.2: Simulated closed-loop response.

### 3.1.2 Analysis

In the closed loop system the vehicle body and tire exhibit smaller oscillations in response to the road disturbances. The acceleration signal amplitude is also smaller in closed loop which indicates a better comfort measure in the quarter-car system. The tire oscillations are also dampened which indicates a better road handling measure.

## 3.2 Implementation

### 3.2.1 Closed-Loop Control

In this section we will use the Simulink diagram shown in Figure 3.3 to implement the closed-loop control of the Active Suspension system.

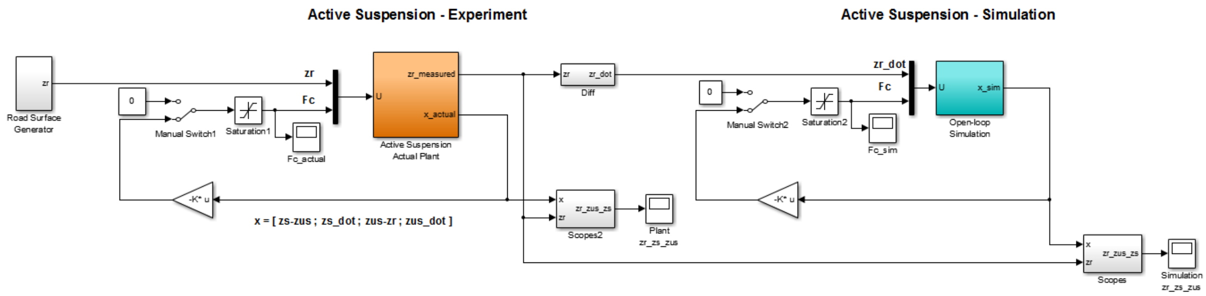
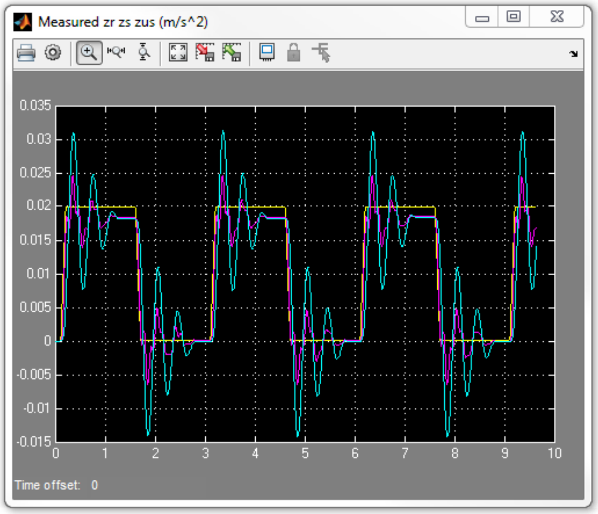


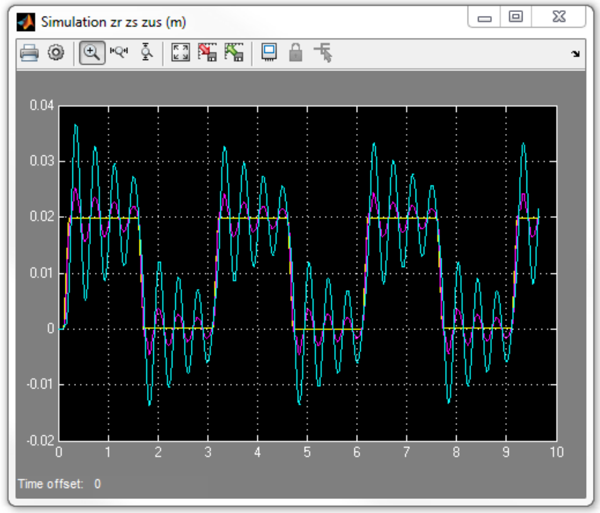

Figure 3.3: Simulink model used to run controller on the Active Suspension.

**IMPORTANT:** Before you can conduct these experiments, you need to make sure that the lab files are configured according to your setup. If they have not been configured already, then you need to go to Section 4 to configure the lab files first.

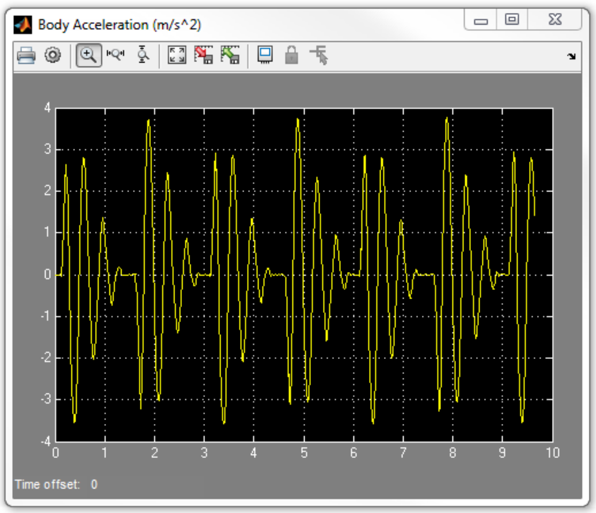
1. Run the setup_as.m script using the LQR weighting matrices that you used in the simulation in Section 3.1.
2. Open the simulated plate position scope, *Simulation zr zs zus (m)*, and the actual plate position scope, *Measured zr zs zus (m/s2)*.
3. In the Simulink diagram, go to QUARC | Build.
4. Click *Connect to Target* to connect to the real-time code, then Click on QUARC | Start to run the controller.
5. *Zr* represents the bottom plate position which refers to the road. *Zus* represents the middle plate position which refers to vehicle tire. *Zs* represents the top plate position which refers to vehicle body. A typical passive suspension, or open loop, response of the system are shown in Figure 3.4.




(a) Measured Open-Loop Plate Positions



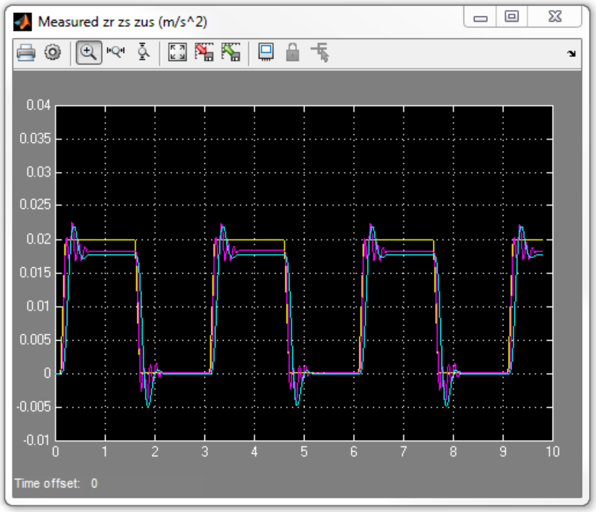

(b) Simulated Open-Loop Plate Positions



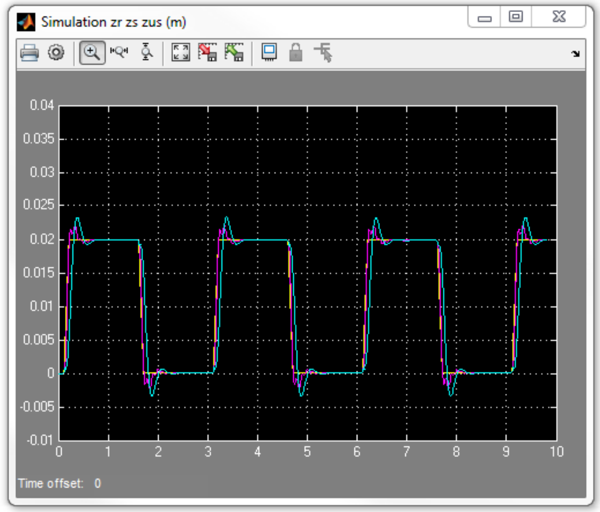

(c) Open-Loop Body Acceleration

Figure 3.4: Active Suspension closed-loop response.

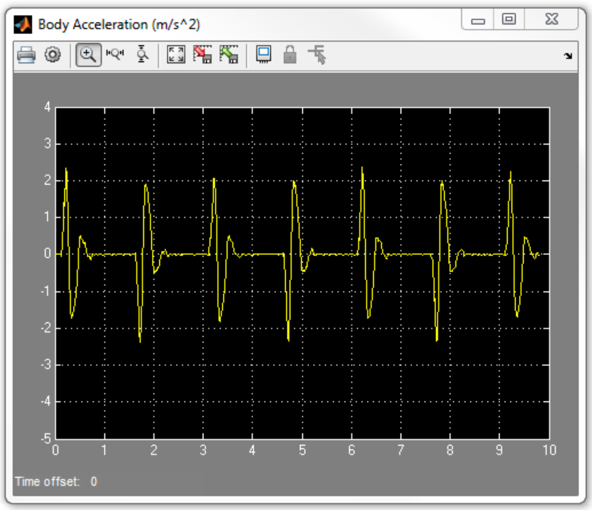
6. Switch between the open loop (passive suspension) and closed loop (active suspension) control using the *Manual Switch* blocks. The closed loop response of the system with the default gains looks like Figure 3.5.




(a) Measured Closed-Loop Plate Positions




(b) Simulated Closed-Loop Plate Positions




(c) Closed-Loop Body Acceleration

Figure 3.5: Active Suspension closed-loop response.

7. Manually tune the control gains if necessary based on the results.

### 3.2.2 Open-Loop Analysis

In this section we will use the Simulink diagram shown in Figure 3.6 to implement the open-loop frequency analysis of the Active Suspension system.

**IMPORTANT:** Before you can conduct these experiments, you need to make sure that the lab files are configured according to your setup. If they have not been configured already, then you need to go to Section 4 to configure the lab files first.

1. Open the q_as_ol.mdl model.
2. Run the setup_as.m script to setup the simulation state space parameters.
3. The road input is a chirp signal. It starts at a frequency of 1 Hz and an amplitude of 0.0015 m. It reaches a

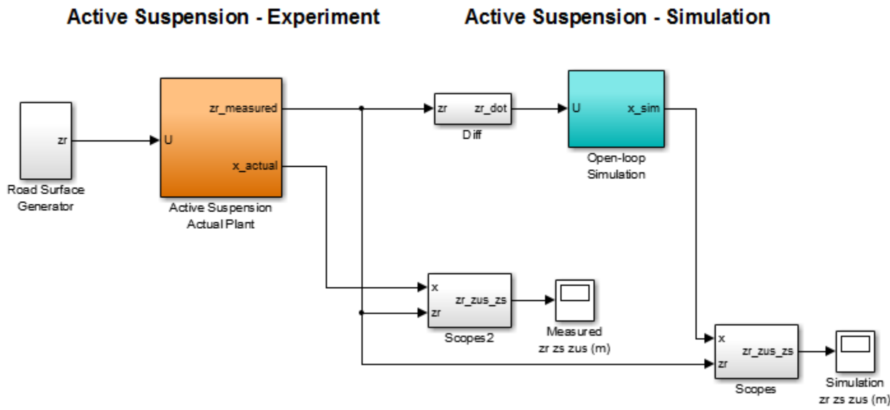



Figure 3.6: Simulink model used to run controller on the Active Suspension.

frequency of 8 Hz at 25 th second. This model can be used to experimentally determine the natual frequency and damping of the system.

4. In the Simulink diagram, go to QUARC | Build.
5. Click *Connect to Target* to connect to the real-time code, then Click on QUARC | Start to run the model.
6. Zr represents the bottom plate position which refers to the road. Zus represents the middle plate position which refers to vehicle tire. Zs represents the top plate position which refers to vehicle body. The open loop response of the system should look like Figure 3.7.

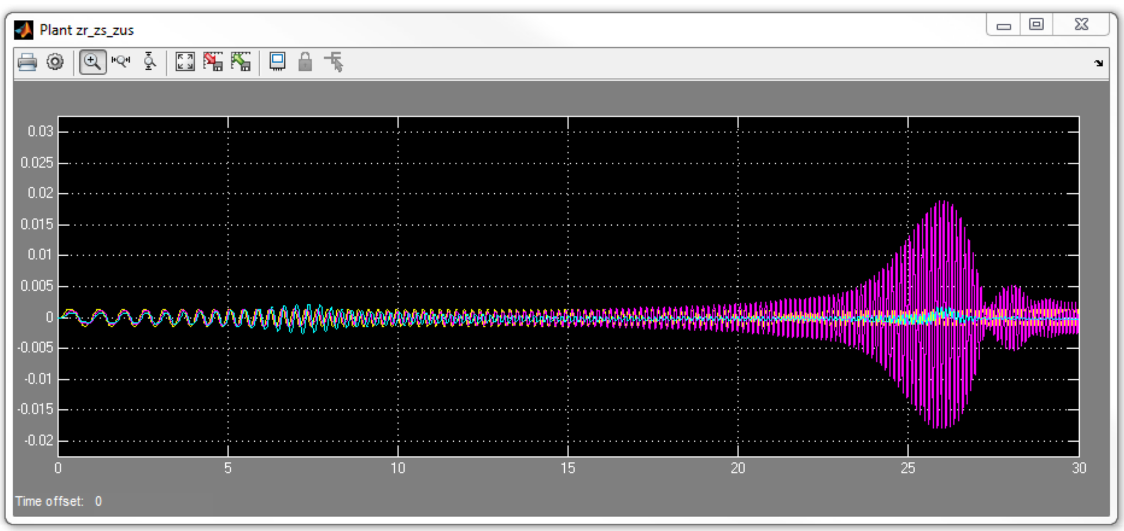



Figure 3.7: Active Suspension Control Simulation VI, Open Loop Response.

7. Stop the model by pressing the *Stop* button or it will stop in 30 seconds.

# 4 SYSTEM REQUIREMENTS

**Required Software**

- Microsoft Visual Studio (MS VS)
- Matlab® with Simulink®, Real-Time Workshop, and the Control System Toolbox
- QUARC®

See the QUARC® software compatibility chart in [4] to see what versions of MS VS and Matlab are compatible with your version of QUARC and for what OS.

**Required Hardware**

- Data acquisition (DAQ) device **with three encoder inputs** and that is compatible with QUARC®. This includes Quanser DAQ boards such as Q8-USB, QPID, and QPIDe and some National Instruments DAQ devices. For a full listing of compliant DAQ cards, see Reference [1].
- Quanser Active Suspension
- Quanser AMPAQ-L4 power amplifier, or equivalent.

**Before Starting Lab**

Before you begin this laboratory make sure:

- QUARC® is installed on your PC, as described in [3].
- DAQ device has been successfully tested (e.g., using the test software in the Quick Start Guide or the *Analog Loopback Demo*).
- Active Suspension and amplifier are connected to your DAQ board as described its User Manual [2].

## 4.1 Overview of Files

| File Name | Description |
|---|---|
| Active Suspension User Manual.pdf | This manual describes the hardware of the Active Suspension system and explains how to setup and wire the system for the experiments. |
| Active Suspension Laboratory Guide.pdf | This document demonstrates how to obtain the linear state-space model of the system, simulate the closed-loop system, and implement controllers on the Active Suspension plant using QUARC®. |
| setup_as.m | The main Matlab script that sets the Active Suspension motor and sensor parameters, the Active Suspension configuration-dependent model parameters, and the Active Suspension sensor parameters. Run this file only to setup the laboratory. |
| s_as_lqr | Simulink file that simulates the closed-loop control of a Active Suspension system using state-feedback control. |
| q_as_lqr | Simulink file that implements the state-feedback control on the Active Suspension system using QUARC®. |
| q_as_ol | Simulink file that runs an open-loop frequency analysis on the Active Suspension system using QUARC®. |

Table 4.1: Files supplied with the Active Suspension

## 4.2 Setup for Simulation

Before beginning the in-lab procedure outlined in Section 3.1, the s_as_lqr Simulink diagram and the setup_as.m script must be configured.

Follow these steps:

1. Load the Matlab software.
2. Browse through the *Current Directory* window in Matlab and find the folder that contains the file *setup_as.m*.
3. Open the *setup_as.m* script.
4. **Configure setup_as**: Make sure the script is setup to match this setup:
   - CONTROL_TYPE = 'AUTO';
5. Run setup_as.m to setup the Matlab workspace.
6. Open the *s_as_lqr.mdl* Simulink diagram, shown in Figure 3.1.

## 4.3 Setup for Running on Active Suspension

Before performing the in-lab exercises in Section 3.2.1, the Active Suspension system and the setup_as.m script must be configured.

Follow these steps to get the system ready for this lab:

1. Configure and run setup_as.m as explained in Section 4.2.

2. Open the *q_as_lqr.mdl* Simulink diagram, shown in Figure 3.3.
3. You are now ready to run and tune the LQR controller.
4. **Configure DAQ:** Ensure the HIL Initialize block in the Simulink model is configured for the DAQ device that is installed in your system. See [1] for more information on configuring the HIL Initialize block.

# REFERENCES

[1] Quanser Inc. *QUARC User Manual*.

[2] Quanser Inc. *Active Suspension Experiment User Manual*, 2009.

[3] Quanser Inc. *QUARC Installation Guide*, 2009.

[4] Quanser Inc. *QUARC Compatibility Table*, 2010.

[5] Norman S. Nise. *Control Systems Engineering*. John Wiley & Sons, Inc., 2008.

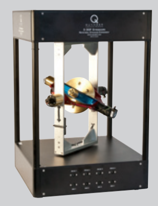
## Full range of robotic and mechatronic control plants for teaching and research

Active Suspension

Omni Bundle

2 DOF Planar Robot

3 DOF Gyroscope



Hexapod*

IMDU Base Unit

IMDU-Web Winding

IMDU-Multi DOF Torsion

* **Please note:** The Hexapod is not available for purchase in North America, Japan and Taiwan.

Choose from eight plants to create experiments for teaching or research related to robotics, haptics, mechatronics, aerospace, or process control. For more information please contact info@quanser.com



QUANSER

INNOVATE. EDUCATE.