SICONOS

What is Siconos ?

SICONOS is the European Project IST2001-37172, funded by the Commission of the European Communities, from September 1, 2002, to August 31, 2006. It is a project of the Information Society Technologies programme, fifth framework programme (FP5). The main coordinator of the project is Prof. Bernard Brogliato (INRIA), and the UPC member responsible for the project is Dr. Gerard Olivar, member of the research groups ACES and SARTI.

This project's goal is the study of complementarity dynamical systems (a class of hybrid dynamical systems). It gathers scientists from various disciplines such as Mechanics, Applied Mathematics, Systems and Control, and Numerical Analysis.

Researchers of the following universities are participating in the SICONOS project:

ETH Zurich (CH), University of Bristol (UK), Tilburg university (NL), University of Saint-Etienne (F),

INRIA Rhone-Alpes (F), Technical University of Catalonia (SP), University of Naples (IT), Delft University (NL), University of Twente (NL), University of Bath (UK), University of Cambridge (UK), Technological University of Eindhoven (NL), University of Montpellier 2 (FR), University of Roma Tor Vergata (I), Imperial College London (UK), University College London (UK).

Project Summary

The name Siconos comes from the title of the project: Modelling, **Si**mulation and **Co**ntrol of **No**nsmooth Dynamical **S**ystems.

The purpose of this grant is to develop algorithms and software for the simulation and feedback control of dynamical systems which are nonsmooth, and more specifically so-called *complementarity dynamical systems*. Nonsmoothness is usually introduced into the system either by some nonsmooth control action or by the presence of nonsmooth events at a macroscopic level (such as impacts or switchings). Nonsmooth models abound in many engineering systems such as sliding mode or hybrid control and rigid body mechanics such as rattle of automotive components and other mechanical freeplay, and switching circuits in power electronics. We have chosen complementarity systems as the mathematical framework for studying nonsmooth nonlinear systems. This framework is large in terms of the range of potential applications, yet specific enough to allow for thorough investigation.

The research project will tackle head on two fundamental issues. First that smooth numerical methods fail on nonsmooth systems. Algorithms need to be developed that deal with hit crossings, impacts, complementarity problems, sliding and chatter in a robust and easily applicable way. Second, the qualitative understanding of the dynamics including the design of feedback and robust control algorithms requires specific methods and cannot be solved with simple adaptations of current techniques for smooth linear or nonlinear dynamical systems.

The research teams are made up of many of the world experts in the theory and applications of these disparate theories. Without doubt, this grant represents a unique opportunity to synthesise current knowledge and thus achieve the much needed goal of a general software for nonsmooth dynamics.

Project Objectives

The strategic aim of this project is the development of novel algorithms and numerical routines for the qualitative analysis, simulation and feedback control of nonsmooth complementarity dynamical systems. The end product of this project will be an integrated numerical software package for the virtual prototyping of systems with discontinuities and the development of novel control techniques for this class of dynamical systems. This will be achieved through an in-depth investigation of the mathematical and engineering open problems related to nonsmooth complementarity systems. This project is clearly focussed on the development of a **user-friendly, versatile and computationally effective numerical tool for nonsmooth systems**,validated through its application to 3 key engineering problems: **power electronic converters, walking robots and automotive systems** and will run over a period of four years.

The main requirements of the numerical software tool can be outlined as follows:

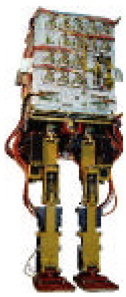
· **Efficient handling of nonsmooth models**.: Nonsmooth models need careful treatment as their phase space can contain an intricate web of discontinuity boundaries which need to be properly specified and dealt with. At the same time the formalism used to describe the system of interest should be synthetic, easy to implement and general enough to encompass a wide class of problems. Under these conditions, it was decided that the most effective analytical framework was that of **complementarity systems** . The decision for this framework was reached unanimously by all the participating teams.

· **Ability to characterise existence and stability of solutions**. In particular, methods are required to investigate the existence of different types of solutions (e.g. periodic solutions) and assess their stability. This is particularly relevant to solve parametric continuation problems for nonsmooth systems.

· **Fast computational engine for time-integration and parametric continuation**. Numerical algorithms need to be derived in order to perform an *accurate* time integration of complementarity systems incorporating their main features (LCP or NCP solvers, treatment of finite accumulation of events, collision detection, choice between event-driven and time-stepping schemes, etc.). Efficient methods and algortithms for the parametric continuations of solutions and identification of bifurcation points are also needed.

· **Toolbox for the design and validation of control strategies for nonsmooth systems.** The control of complementarity systems (and other formalisms for nonsmooth systems) is a rapidly expanding area of research We envisage the development of novel control techniques for nonsmooth complementarity systems (see below for further details) and the development of numerical routines to perform their validation.

· **Modular structure and user-friendly interface.** The numerical software we plan to develop should be modular and easy to expand with additional routines (or toolboxes). In contrast to some of the existing software, it should be easy to use through an interactive, user-friendly graphical interface.



USB Connectivity for Microcontrollers

The possibility of having a communication system between a personal computer and a microcontroller, microprocessor that initially lacks a data transmission interface, is undoubtedly a very useful tool in certain situations. The small-size of the components and the power supply of the device through a USB port (Universal Serial Bus) allows the design of a user-friendly interface.

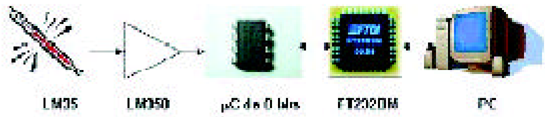


Fig. 5. Block diagram of the USB temperature measurement system

FT232BM. This provides the conversion of TTL levels coming from the microcontroller communication port to USB standard signals and also implements the connection, addressing, etc. protocol. The configuration «Bus Powered» is permitted because it can be powered between 4.35V and 5.25V. The integrated circuit supports the newest versions of USB as USB 1.1 and USB 2.0. The use of this integrated circuit provides USB connectivity to our system in a quick and easy way. This chip requires a 6MHz oscillator, which provides the needed frequency for USB communication.

MC68HC908QT4. This is an 8-bit microcontroller of the Motorola HC08 family, which integrates an analogdigital converter that will convert the signal, coming from the temperature sensor and send it by frames to FT232BM. We have used this microcontroller because of its small size, low power consumption and flexibility. Data is packed in a frame in series format with a start bit, 8 data bits, no parity and a stop bit. The frame is identical to one that we would send to a conventional RS232 port, but in this case the frame is sent to the FT232BM, which is in charge of sending the data to the USB port.

The software is composed of USB drivers which allow the operating system to receive the incoming data from this port, the compiler for the microcontroller programming and a software interface developed in LabVIEW which monitors the data coming from the device.

USB drivers. The USB devices can be connected in <<plug and play>>. The operating system recognizes the interface and requests from the associated controller. The controller used is the Virtual COM port and can be downloaded from the FTDI chip page.