EVALUATION AND IMPLEMENTATION OF ADVANCED PROCESS CONTROL WITH THE COMPACTRIO MATERIAL FROM NATIONAL INSTRUMENTS

G. Maire, A. Kehrli, M. Pezzetti, S. Ravat, CERN Geneva, Switzerland
B. Charnier, H. Coppier, ESIEE Amiens, France

Abstract

Programmable Logic Controller (PLC) is very commonly used in many industries and research applications for process control. However a very complex process control may require algorithms and performances beyond the capability of PLC's, very high-speed or precision controls may also require other solutions. This paper describes recent research conducted to implement advanced process controls with the compactRIO (cRIO) material from National Instruments (decoupling of MIMO process control, steady state feedback, observer, Kalman filter, etc…). The cRIO systems consist of an embedded real-time controller for communication and processing, a Reconfigurable Field Programmable Array (FPGA) and hot-swappable I/O modules. The paper presents experimental results and the ability of the cRIO to treat complex process control.

INTRODUCTION

Programmable Logic Controller (PLC) is very commonly used in many control systems at CERN. However, PLCs show limitations to perform very high-speed treatment or complex calculations for advanced controls. After a brief description of the cRIO material from National Instruments, this paper presents the capability and the performances of the cRIO on the basis of two case studies. The first study is a decoupling controller with a full state feedback for Multiple Input Multi Output (MIMO) process control. The second study is a linear Kalman filter at very high-speed with calculations of complex algorithms.

THE COMPACTRIO

The cRIO is a reconfigurable embedded control and acquisition system (Figure 1). The cRIO is made of:

- A Real-Time Controller containing a processor running a real-time operating system (RTOS), that reliably and deterministically executes LabVIEW Real-Time applications and offers multirate control, execution tracing, onboard data logging, and communication with peripherals.
- A FPGA containing a matrix of reconfigurable gate array logic circuitry that, when configured, is connected in a way that creates a hardware implementation of a software application.
- A large number and hot-swappable I/O module are available.

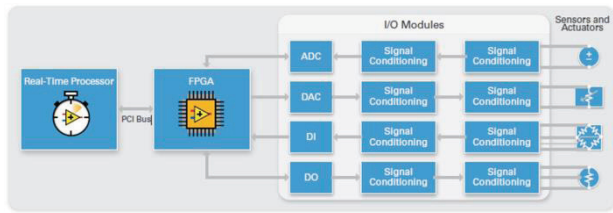

Figure 1: Architecture of compactRIO.

Unlike the real-time controller processor, the FPGA uses dedicated hardware for processing logic and does not have an operating system. Because the processing paths are parallel, different operations do not have to compete for the same processing resources. That means speed can be very fast, and multiple control loops can run on a single FPGA device at different rates.

MULTI INPUT MULTI OUTPUT (MIMO) PROCESS CONTROL

Introduction

To investigate the feasibility and assess the performance of the cRIO for advanced control systems, this subchapter presents the design of a decoupling MIMO corrector (Figure 2) and full state-feedback pole placement required to perform the matrix-vector multiplication.

In order to validate this type of advanced control, we applied it to the SPS beam transfer lines at CERN.

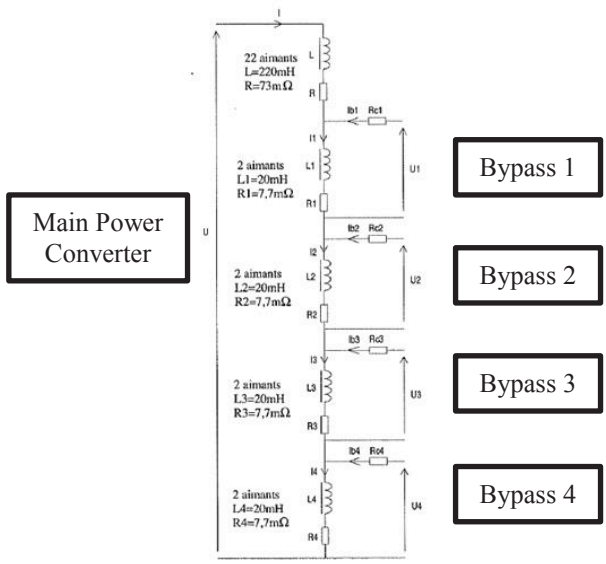

Figure 2: MIMO process.

ISBN 978-3-95450-139-7

Copyright © 2014 CC-BY-3.0 and by the respective authors

The magnets connected in series are supplied by a main power converter; in order to do a correction some magnets are connected to a bypass power converter.

Modelling of the process

From the linear differential equations, we can describe the process by a state-space model (with p inputs, q outputs, n state variables) by the following equations:

$$\dot{x}(t) = Ax(t) + Bu(t) \quad (1)$$

$$y(t) = Cx(t) + Du(t) \quad (2)$$

Where A is the state matrix (nxn), B is the input matrix (nxq), C is the output matrix (qxn), D is the feedforward matrix (qxp).

The linear differential equations are the following:

$$U = L\frac{dI}{dt} + RI + L_1\frac{dI_1}{dt} + R_1I_1 + L_2\frac{dI_2}{dt} + R_2I_2 + L_3\frac{dI_3}{dt} + R_3I_3 + L_4\frac{dI_4}{dt} + R_4I_4 \quad (3)$$

$$U_1 = L_1\frac{dI_1}{dt} + R_1I_1 + R_{C1}(I_1 - I) \quad (4)$$

$$U_2 = L_2\frac{dI_2}{dt} + R_2I_2 + R_{C2}(I_2 - I) \quad (5)$$

$$U_3 = L_3\frac{dI_3}{dt} + R_3I_3 + R_{C3}(I_3 - I) \quad (6)$$

$$U_4 = L_4\frac{dI_4}{dt} + R_4I_4 + R_{C4}(I_4 - I) \quad (7)$$

The output equations are the following:

$$I_{b1} = I_1 - I \quad (8)$$

$$I_{b2} = I_2 - I \quad (9)$$

$$I_{b3} = I_3 - I \quad (10)$$

$$I_{b4} = I_4 - I \quad (11)$$

In accordance with (1) and (2) the MIMO process can be represented by the following matrix system:

$$\begin{bmatrix} \frac{dI}{dt} \\ \frac{dI_1}{dt} \\ \frac{dI_2}{dt} \\ \frac{dI_3}{dt} \\ \frac{dI_4}{dt} \end{bmatrix} = \begin{bmatrix} \frac{(-R-R_{C1}-R_{C2}-R_{C3}-R_{C4}}{L} & \frac{R_{C1}}{L} & \frac{R_{C2}}{L} & \frac{R_{C3}}{L} & \frac{R_{C4}}{L} \\ \frac{R_{C1}}{L_1} & \frac{(-R_{C1}-R_1)}{L_1} & 0 & 0 & 0 \\ \frac{R_{C2}}{L_2} & 0 & \frac{(-R_{C2}-R_2)}{L_2} & 0 & 0 \\ \frac{R_{C3}}{L_3} & 0 & 0 & \frac{(-R_{C3}-R_3)}{L_3} & 0 \\ \frac{R_{C4}}{L_4} & 0 & 0 & 0 & \frac{(-R_{C4}-R_4)}{L_4} \end{bmatrix} \begin{bmatrix} I \\ I_1 \\ I_2 \\ I_3 \\ I_4 \end{bmatrix} + \begin{bmatrix} \frac{1}{L} & -\frac{1}{L} & -\frac{1}{L} & -\frac{1}{L} & -\frac{1}{L} \\ 0 & \frac{1}{L_1} & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{L_2} & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{L_3} & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{L_4} \end{bmatrix} \begin{bmatrix} U \\ U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} \quad (12)$$

$$\begin{bmatrix} I \\ I_{b1} \\ I_{b2} \\ I_{b3} \\ I_{b4} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} I \\ I_1 \\ I_2 \\ I_3 \\ I_4 \end{bmatrix} \quad (13)$$

For this process we can see that the matrix D is equal to zero.

Process Simulation

As we can see in Figures 3 and 4, the step response to an input affects all outputs.

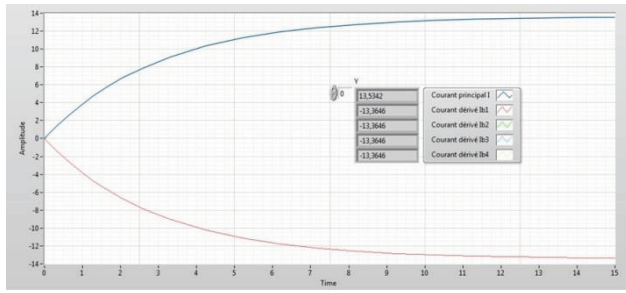

Figure 3: Step response of the main power converter.

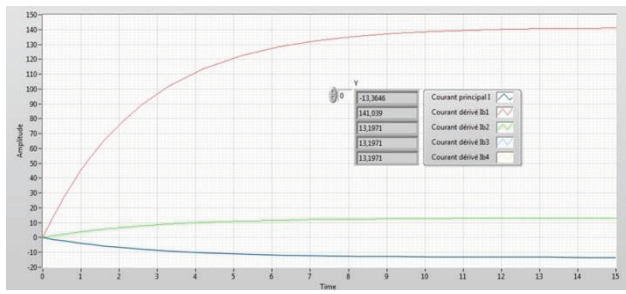

Figure 4: Step response of the bypass 1.

MIMO decoupling and pole placement

The goal of decoupling corrector is to limit the response of an input to an output with the following state feedback:

$$U = \beta V + \alpha X \quad (14)$$

The necessary and sufficient condition to decouple a MIMO process is that matrix β is invertible which is the case for this process.

In order to adjust the dynamic in close loop by pole placement, we add an additional corrector with the following state feedback:

$$V = E - KX \quad (15)$$

Figure 5 shows the structure of the process, the decoupling corrector and pole placement state feedback.

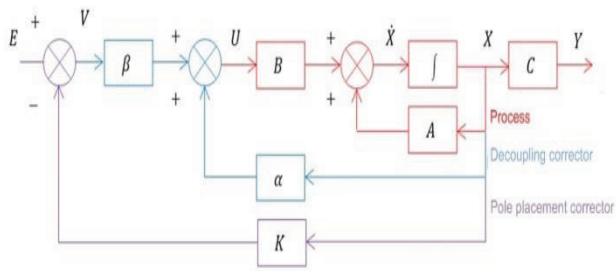

Figure 5: Decoupling and pole placement corrector.

Copyright © 2014 CC-BY-3.0 and by the respective authors

ISBN 978-3-95450-139-7

Software Implementation

Figure 6 shows the implementation in the FPGA of the digital corrector for decoupling and the pole placement, the process is also implemented in the FPGA.

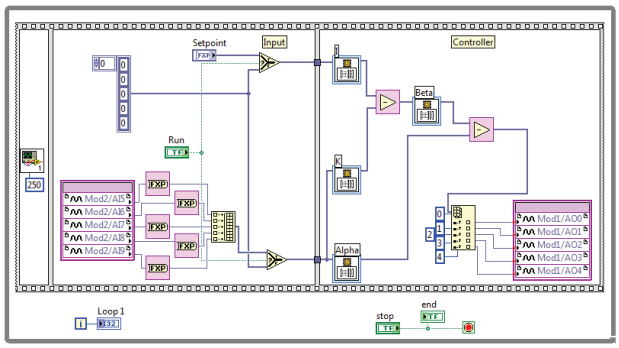

Figure 6: Labview VI of the controller.

Results

Figure 7 shows the step response for the main power converter command, the step response for a bypass is similar. The inputs are decoupled of the outputs and the dynamic of each subsystem can be adjusted (here with pole at -70, time constant is 14ms).

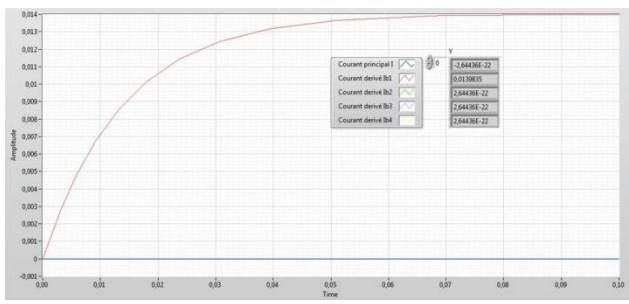

Figure 7: Step response.

KALMAN FILTER

Introduction

To study and evaluate the performance of the cRIO FPGA for signal processing, it is necessary to implement more complex algorithms. The linear Kalman filter is an interesting candidate, indeed the related filter algorithm consists of several matrix operations (addition, multiplication and inversion). These operations with non-constant variables are not offered by the LabVIEW toolbox, the first task was to create them.

Kalman Filter Theory

The Kalman filter is a powerful and useful mathematical tool in the embedded system. It allows to estimate the state of a system, depending on its previous state, controls applied and noisy measurement.

Equations of a discrete Kalman filter are divided into two stages, a step of prediction given by (16) and (17), and a step of correction given by (18) and (19).

$$\hat{x}_{k+1/k} = A_k \hat{x}_{k/k} + B_k u_k \quad (16)$$

$$P_{k+1/k} = A_k P_{k/k} {A_k}^T + G_k Q_k {G_k}^T \quad (17)$$

$$\hat{x}_{k/k} = \hat{x}_{k/k-1} + K_k (y_k - C_k \hat{x}_{k/k-1}) \quad (18)$$

$$P_{k/k} = (I - K_k C_k) P_{k/k-1} \quad (19)$$

And K_k is the optimal gain of the filter given by:

$$K_k = P_{k/k-1} {C_k}^T \left(C_k P_{k/k-1} {C_k}^T + R_k \right)^{-1} \quad (20)$$

Where $\hat{x}$ is the estimation of the state correction, P is the estimation error, G_k is the matrix specifying the noise vector involved in the model of evolution and H_k is the matrix specifying the noise vector involved in the model of observation.

Kalman Filter Implementation

The toolbox of FPGA LabVIEW provides a single tool to multiply a matrix with constant coefficients with a vector. As part of the Kalman filter, it is necessary to perform matrix multiplication with variable coefficients with vector and also matrix multiplication by matrix (matrix being variable coefficients).

Algorithms for matrix multiplication are easily designed but the calculation of the inverse matrix is a problem with the Gauss algorithm; however for a small order matrix the Laplace formula simplifies the calculation:

$$A^{-1} = \frac{1}{det\ A} {}^t com\ A \quad (21)$$

The implementation of the Kalman filter was realized in the FPGA according the equations described in the previous subchapter.

The strategy is to implement each algorithm one time in a main loop in order to minimize the use of the DSP, which however increases the total time calculation of the filter. The calculation of the estimation and correction of the state vector $\hat{x}$ are implemented in a secondary loop parallel to the calculation of the covariance matrix of the estimation error P and the gain matrix K.

Results

In order to test Kalman filter implemented in the FPGA, the test was realized on a process with the state-space representation:

$$\dot{X} = \begin{bmatrix} -1 & 1 & -1 \\ 0 & -1 & 0 \\ 0 & 1 & 0 \end{bmatrix} X + \begin{bmatrix} 0 \\ 1 \\ -1 \end{bmatrix} u \quad (17)$$

$$Y = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} X \quad (18)$$

Figure 8 shows the simulation of this process. A noise is added according the Kalman filter theory (figure 9). Figure 10 shows the results with Kalman filter implemented in the cRIO. Figure 11 shows the estimation error.

ISBN 978-3-95450-139-7

Copyright © 2014 CC-BY-3.0 and by the respective authors

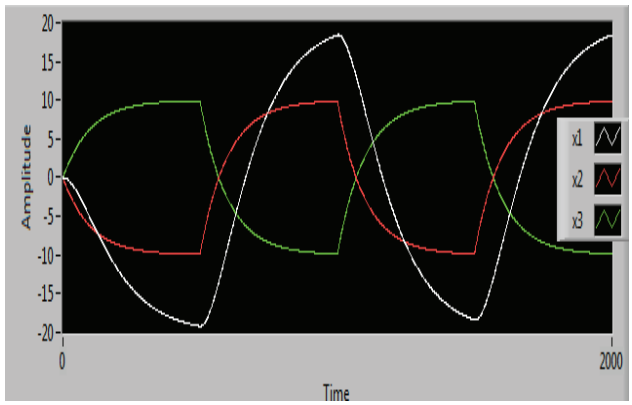

Figure 8: State of the process.

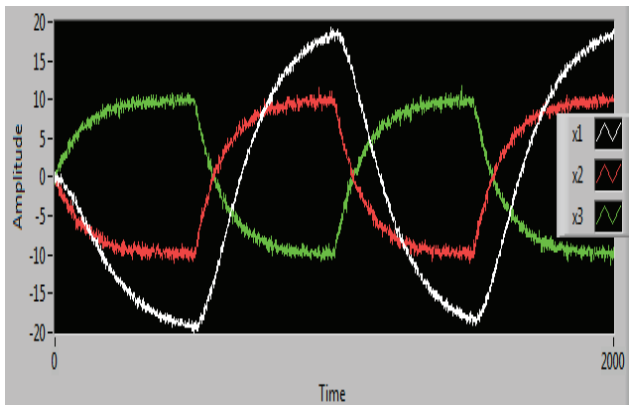

Figure 9: The noisy state.

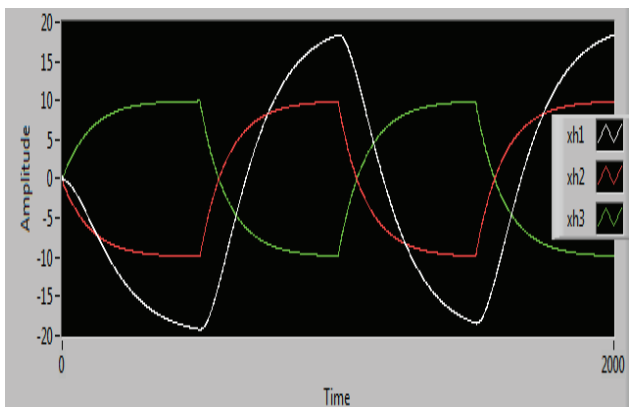

Figure 10: Estimated state with Kalman filter.

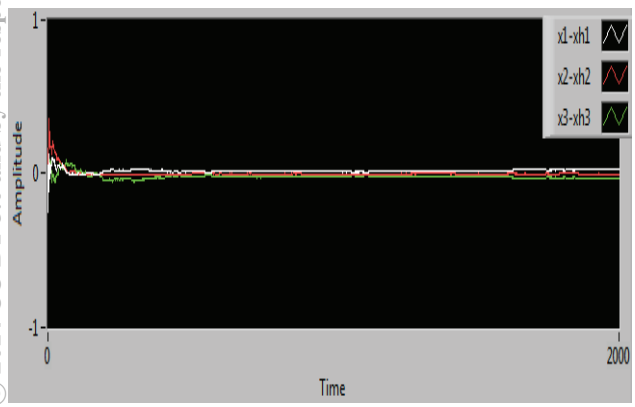

Figure 11: Estimation error.

SUMMARY

The speed of the FPGA in the cRIO and the toolbox from Labview allows the implementation of complex calculations in order to achieve the advanced control process command and the signal processing.

The acquisition time of the I/O modules and the treatment for the MIMO advanced control presented previously is 10 µs. Thus for a fast and advanced control process the cRIO technology is a very good solution in comparison to the PLC's, in which the time cycle is between 20 ms to 50 ms depending on the complexity of the process and the number of I/O modules.

The implementation of a linear Kalman filter demonstrates the speed capacity of the FPGA (cycle of 50 µs), and also its ability to process complex algorithms.

The algorithm calculation of the Kalman filter, especially for the inverse matrix of third order, represents the major expertise in this study. Indeed the toolbox of LabVIEW does not provide routines for multiplying or inverse matrix with non-constant variables in the FPGA, against this tool exists in the LabVIEW library for the real-time controller.

REFERENCES

[1] G. Maire, "Advanced control to supply magnets connected in series", Thesis engineer of CNAM school, June 1998..

[2] B. Charnier, "Performance study of the CompactRIO for the advanced control". Thesis engineer of ESIEE school, June 2013.

[3] C. Keramsi, T. Porteboeuf and F. Thorey, "Les filtres de Kalman," Robotics and Embedded Systems classes of Télécom ParisTech; http://rose.eu.org/2011/wp-content/uploads/2011/03/Kalman-article.pdf

[4] http://www.ni.com/compactrio/

Copyright © 2014 CC-BY-3.0 and by the respective authors

ISBN 978-3-95450-139-7