

# BUS PROTOCOL SIMULATION TO OPERATE DAQ CARD USING DATA ACQUISITION TOOLBOX

*Doc.Ing. Miroslav Kubiček, CSc*

ICT, Department of Computing and Control Engineering, Prague

## Abstract

For education purpose has been created a simple system to show communication between PC and a data acquisition card. This system (Fig.1) employs the AD512 data acquisition card and to generate control signals on the ISA bus the device National Instruments USB-6501, having 24 digital I/O channels. Basic application of a data acquisition card is analog and digital input/output. Programs in MATLAB with Data Acquisition Toolbox are presented.

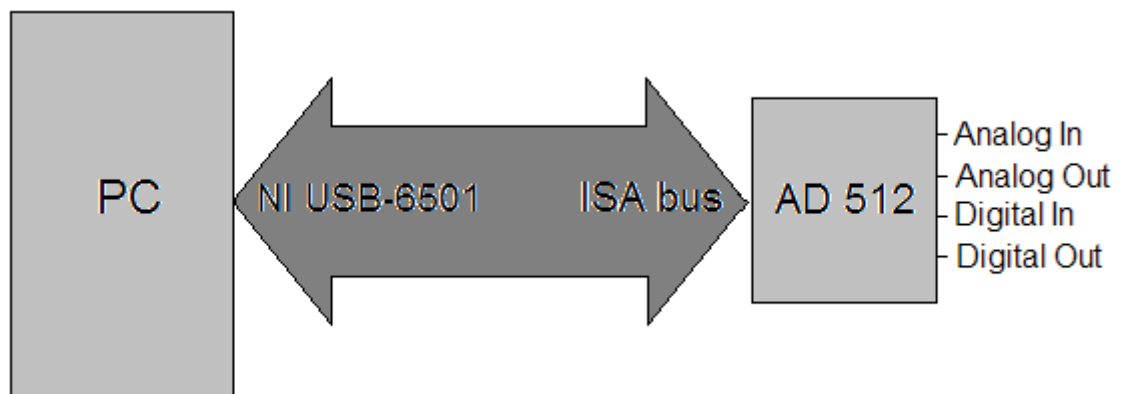


Figure 1. The system for the bus protocol simulation of the DAQ card

## 1 Introduction

Application of the AD512 data acquisition card is purely purpose-built to show how works relatively simple bus and use a low-cost card. There is no problem to apply similar system to any other bus for example to PCI, but system will be more complicated.

Main idea of this problem is to create necessary bus signal sequence using MATLAB instructions and put it on the bus connector of the operated device.

I/O space of AD 512 card consists of eight registers following the base address (Tab.1).

ADDRESS	READ	WRITE
Base+0	<b>ADLO</b> - A/D data low byte	<b>DA0LO</b> - D/A 0 data low byte
Base+1	<b>ADHI</b> - A/D data high byte	<b>DA0HI</b> - D/A 0 data high byte
Base+2		<b>DA1LO</b> - D/A 1 data low byte
Base+3		<b>DA1HI</b> - D/A 1 data high byte
Base+4		<b>DACTRL</b> - D/A control register
Base+5	<b>ADSTAT</b> - A/D status register	<b>ADCTRL</b> - A/D control register
Base+6		
Base+7	<b>DIN</b> - Digital input register	<b>DOUT</b> - Digital output register

Table 1. I/O Port Map of A/D 512 card

Time signal sequence of the ISA bus for data output and input are in Fig.2.

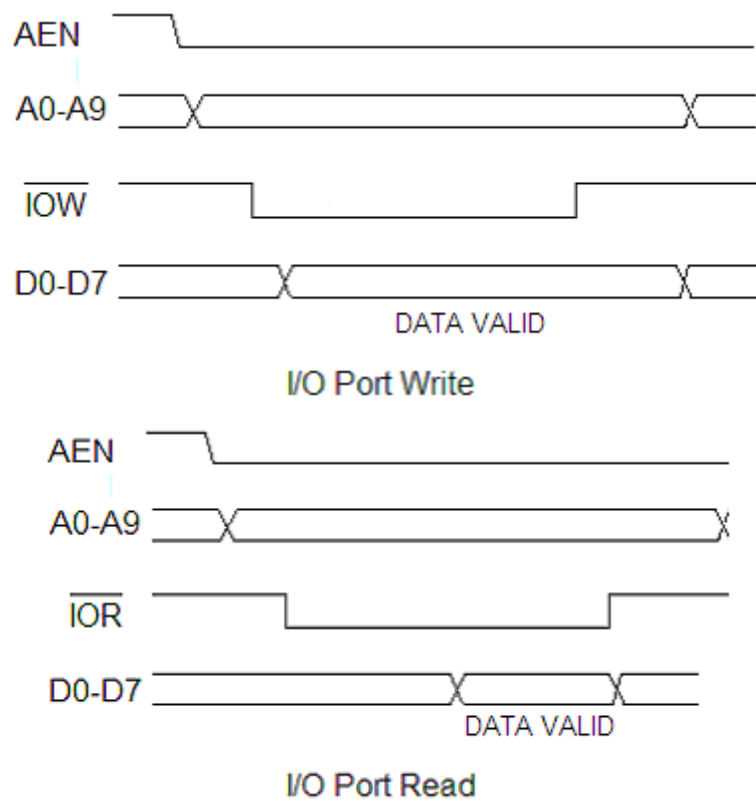


Figure 2. Signals on the ISA bus for data output and input

Port 0 (P0) of the NI USB-6501 is connected to following control signals of the ISA bus:

P0.0 > A0; P0.1>A1; P0.2 > A2; P0.3 > AEN; P0.4 > Reset; P0.5 > IOW; P0.6 > IOR.

This connection defines each bite in the binary control data.

Port 2 (P2) of the NI USB-6501 is used for output or input of a transferred data.

## 2 Programs

Programs realizing those time sequences in MATLAB using Data Acquisition Toolbox are following:

```
%program digitalOutput.m
mujDIO = digitalio('nidaq','Dev2');
addline(mujDIO, [0:6 16:23], 'out');
data = logical ([1 1 1 0 1 0 1 0]); % P2 data
rizeni = logical ([1 1 1 0 0 0 1]); % P0 - signals ISA bus(A0,A1,A2,AEN,RESET,IOW,IOR)
dataA =logical ([rizeni data]);
putvalue(mujDIO, dataA);
data =logical ([0 0 0 0 0 1 1 0 0 0 0 0 0 0 0]); % IOW
putvalue(mujDIO, data);
delete(mujDIO)
```

```

% program digitalInput.m
mujDIO = digitalio('nidaq','Dev2');
addline(mujDIO, 0:6, 'out');
addline(mujDIO, 16:23, 'in');
rizeni1 = logical ([1 1 1 0 0 0 0]); % P0 - signals ISA bus (A0,A1,A2,AEN,RESET,IOW,IOR)
putvalue(mujDIO.Line(1:7), rizeni1); % IOR must be on the LOW level
getvalue(mujDIO.Line(8:15)) % different line numbering (16:23 in addline)
delete(mujDIO)

```

```

% program ADconversion.m
addline(mujDIO, 0:6, 'out'); % control ISA bus
addline(mujDIO, 16:23, 'out'); % data register
ADrizeni = logical ([0 0 0 1 0 1 0]); % data for ADCTRL reg.: input range, address, see [2]
rizeni1 = logical ([1 0 1 0 0 0 1]); % P0 – signals ISA BUS (A0,A1,A2,AEN,RESET,IOW,IOR)
dataA =logical ([rizeni1 ADrizeni]);
putvalue(mujDIO, dataA); % write to ADCTRL register
pause(0.001);
mujDIO = digitalio('nidaq','Dev2');
addline(mujDIO, 0:6, 'out');
addline(mujDIO, 16:23, 'in');
rizeni2 = logical ([0 0 0 0 0 0 1]); P0 - signals ISA BUS(A0,A1,A2,AEN,RESET,IOW,IOR)
putvalue(mujDIO.Line(1:7), rizeni2); % address DA0LO
rizeni2 = logical ([0 0 0 0 0 0 0]); % P0 - signals ISA bus (A0,A1,A2,AEN,RESET,IOW,IOR)
putvalue(mujDIO.Line(1:7), rizeni2);
getvalue(mujDIO.Line(8:15)) % read from DA0LO
rizeni3 = logical ([1 0 0 0 0 0 1]); % P0 - signals ISA bus(A0,A1,A2,AEN,RESET,IOW,IOR)
putvalue(mujDIO.Line(1:7), rizeni3); % address DA0HI
rizeni3 = logical ([1 0 0 0 0 0 0]); % P0 - signals ISA bus (A0,A1,A2,AEN,RESET,IOW,IOR)
putvalue(mujDIO.Line(1:7), rizeni3);
getvalue(mujDIO.Line(8:15)) % read from DA0HI
delete(mujDIO)

```

```

% program DAconv.m
mujDIO = digitalio('nidaq','Dev2');
addline(mujDIO, [0:6 16:23], 'out');
%% write to DA0LO
data = logical ([0 0 0 0 0 0 0]); % port P2 data

```

```

rizeni = logical ([0 0 0 0 0 1]); % P0-signals ISA bus (A0,A1,A2,AEN,RESET,IOW,IOR)
dataA =logical ([rizeni data]);
%% write to DA0HI
data = logical ([0 0 0 1 0 0 0 0]); % port P2 data
rizeni = logical ([1 0 0 0 0 1]); % P0-signals ISA bus (A0,A1,A2,AEN,RESET,IOW,IOR)
dataB =logical ([rizeni data]);
putvalue(mujDIO, dataB);
putvalue(mujDIO, dataA);
%% write to DACTRL
rizeni = logical ([0 0 1 0 0 1]); % P0-signals ISA bus (A0,A1,A2,AEN,RESET,IOW,IOR)
putvalue(mujDIO.Line(1:7), rizeni);
delete(mujDIO)

```

### 3 Results

Presented programs have been experimentally verified. More experienced programmer can perhaps find easier way. Main contribution of the presented paper is application of the Data Acquisition Toolbox to control a digital system. Similar way can be used to control any digital system without direct connection to the PC.

### References

- [1] *Data Acquisition Toolbox – For Use with MATLAB, User's Guide*, The MATHWORKS Inc., 1999 or later
- [2] *AD 512 Data Acquisition Card, User's Manual*, Humusoft s.r.o, 1997