

Possible way of data communication between Matlab and MS Excel

Karel Zaplatílek, Bohuslav Doňar

Brno Military Academy
Department of Electrical engineering and electronics
Kounicova 65, 612 00 Brno, Czech Republic
Email: karel.zaplatilek@vabo.cz bohuslav.donar@vabo.cz

Summary

An original application was developed by means of the Matlab in our department. This program serves for reciprocal dynamic data exchange between program systems Matlab and Microsoft Excel. It is possible to exchange the data by a single or with utilization of a hot-line (see bellow). The presented application is not restricted on two mentioned programs. The Matlab is set as a client and the other systems (e.g. Excel) as a server.

1. Introduction

An original measuring system named AMEX was recently developed in our department [1]. The system is developed as add-in module of Microsoft Excel program. The measuring is very effective with mentioned AMEX system because controlling of one is intuitive. The system Matlab 5 is often used in our department for teaching and scientific works [2]. We wanted to connect through these two applications. The developed Matlab application was named DATA TRANSLATOR, see fig. 1.

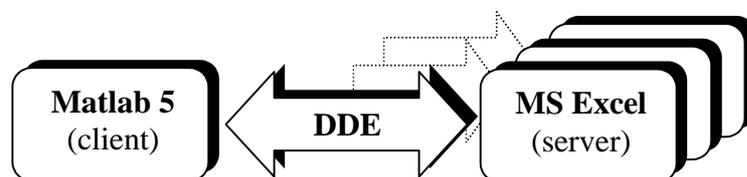


Fig. 1. Block diagram of the DATA TRANSLATOR system.

2. Possibilities of developed system

The described system DATA TRANSLATOR is developed as Matlab user's function (one *m-file*). Its fundamental properties are as follows:

- two-way communication (Matlab←Excel, Matlab→Excel),
- one-shot data transmission,
- hot-line data transmission (reaction to data change),
- graphical interface and simple operating.

The system has Czech or English menu. The following conditions must be realized to correct working:

- ◆ the Matlab 5 and its user's function DATA TRANSLATOR must be run,
- ◆ the MS Excel with required book must be open.

Example of one-shot communications in direction Matlab ← Excel

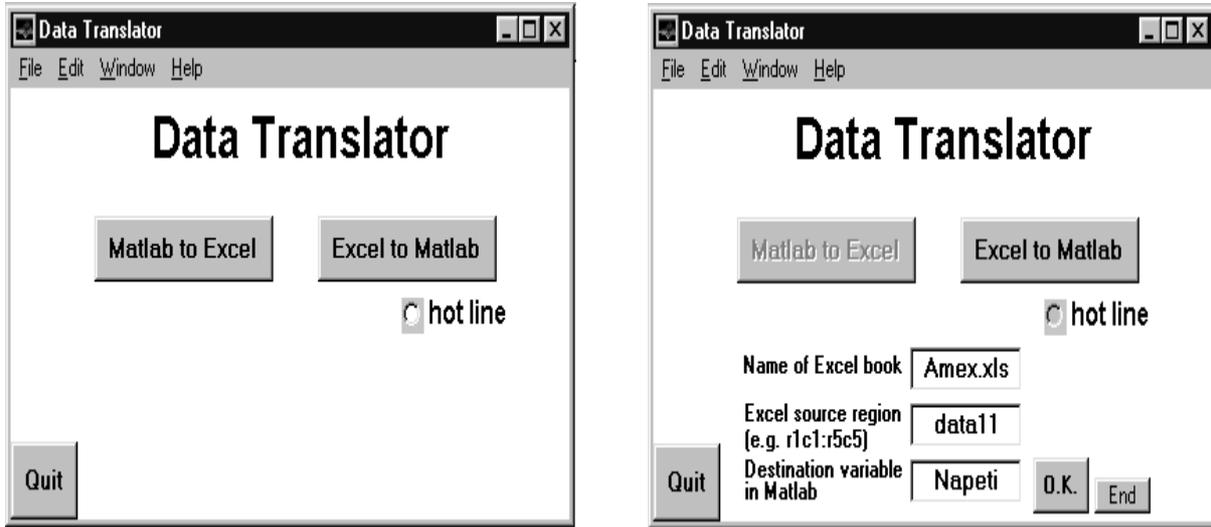


Fig. 2. A basic environment of the DATA TRANSLATOR system.

A basic program environment is shown in fig. 2. A direction *Matlab ← Excel* is chosen there, the *hot-line* is not selected. The system by a single translates the data from the Excel region *Data1* to the Matlab variable *Napeti*. If the *hot-line* is chosen then the data transmission is realized only if the data change.

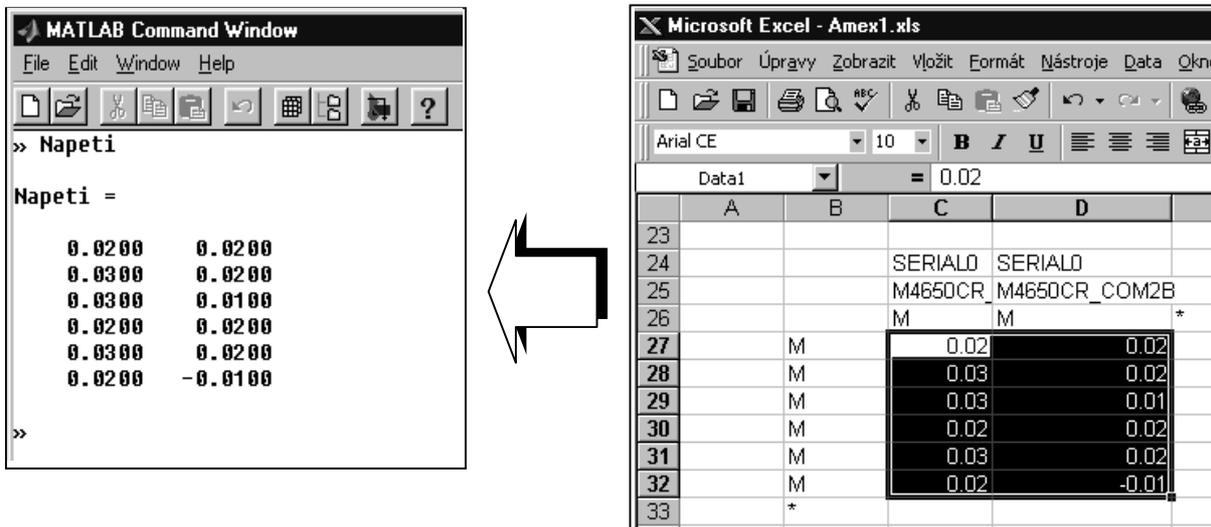


Fig. 3. A result of the data transmission.

A basic controlling of developed system is shown in fig. 2 and fig. 3. If the button *O.K.* is pressed (fig. 2) then the data are transferred. Nevertheless the connection between programs are not interrupted that is why system action can continue. The source data were obtained by direct measurement by means of two multimeters M4650CR.

3. Structure of developed function

The described function DATA TRANSLATOR is programmed directly in Matlab programming language. A definition and operating of a data channel is an interesting illustration of this *m-file*.

Example of data channel definition a operating

```
case('OKExMat')
    Hot=get(findobj('Tag','radiol'),'Value'); % Hot line detection
    Sesit=get(findobj('Tag','ExMat1'),'String'); % Excel book
    region=get(findobj('Tag','ExMat2'),'String'); % Excel region
    channel=ddeinit('excel',sesit); % A data channel definition
    set(findobj('Tag','fig1'),'UserData',channel);
    if Hot==1 % Selection of the Hot line
        rc=ddeadv(channel,region,'assignin(''base'',get(findobj
            (''Tag'', 'ExMat3'),'String'),'Pp);disp(Pp),'Pp');
    else
        Mat=ddereq(channel,region); % data transmission from Excel
        assignin('base',get(findobj('Tag','ExMat3'),'String')
            ,Mat) % activation of required variable in Workspace
    end
    .
    .
    .
rc=ddeterm(channel) % termination of open data channel
```

The Matlab 5 has seven DDE client functions. Some of them were used in my function, see underline commands above. There are two user's environments. The first of them is so-called *Workspace*, which represents command interpreter. The second of them is programming environment where it is possible to write source texts of *scripts* or *functions*. It is not trivial to solve the reciprocal communication between two mentioned environments, see command *assignin* above.

4. Conclusion

An unusual Matlab application for DDE communication was developed. It was tested in practice during the measurement by Excel measuring add-in module AMEX. In future the developed application DATA TRANSLATOR will be extended because of communication with the other software including the measuring cards.

References

- [1] TALPA, M.: Automated measurement with Microsoft Excel.. In: Proceedings. of the conference „Moderní směry výuky elektrotechniky a elektroniky“. Brno, Vojenská akademie, 1999, pp. 235-238.
- [2] KRTIČKA, A.-ZAPLATÍLEK, K.-BIOLEK, D.: Matlab and signals teaching in the circuit theory. In: Proceedings of the conference „Moderní směry výuky elektrotechniky a elektroniky“. Brno, Vojenská akademie, 1999, pp. 82-85.