Appendix A – Software Manual

Short overview

Program MuxDmux enables computer assisted design for realizations of circuits with multiplexers and demultiplexers.

The operation of the program consists of entering a function definition and then selecting a structure in which this function will be implemented. Function may be entered in two forms: canonical numerical decimal or literal. The function definition may be also loaded from previously saved file. The entered function is then minimized using Quine-McCluskey algorithm. If user chooses literal form of function to enter then they have possibility to turn off minimization. Next a user can define a structure that is required. Afterwards the program generates a diagram of the chosen design. The result is displayed in the program window. When a structure is set up it can be stored in a file as a project definition, exported to a chosen graphical format or printed.

Two different file formats of data inputs are implemented in the program, mainly:

- Function definition file – it stores a definition of a function entered,
- Project definition file – it contains a definition of a function and a generated structure definition.

Therefore, everything in this manual referred to a function is connected with a function definition itself, and referred to a project is connected with a function and a design structure altogether.

Main window

Executing the program causes the window (Fig. 1) to be displayed. As it can be seen, some options from the toolbar and menu are disabled. They become active when appropriate data is entered, or structure generated.

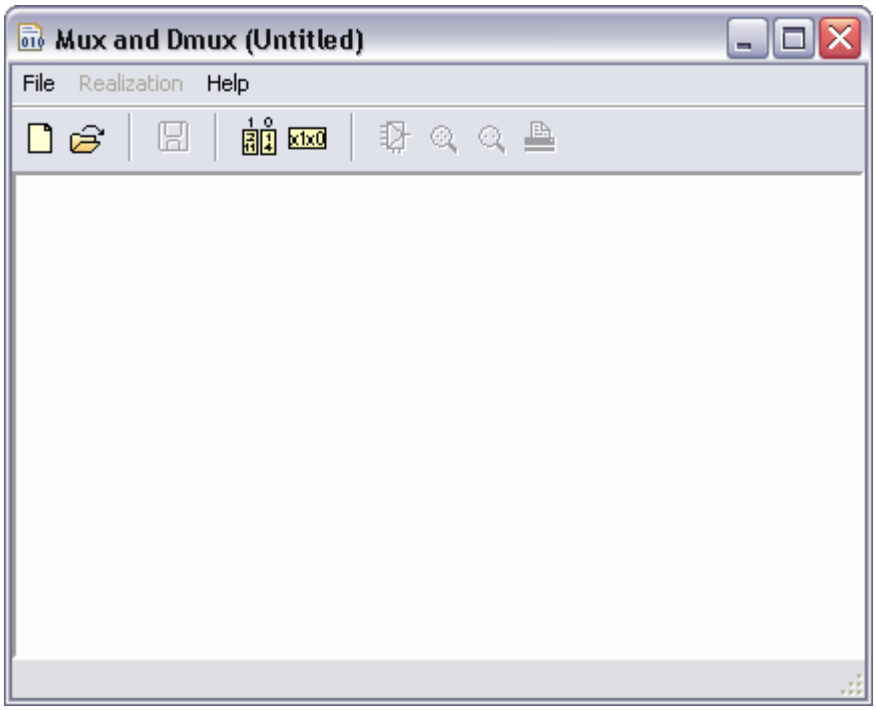


Fig. 1. Main program window

The toolbar with all available functions is shown in Fig. 2.

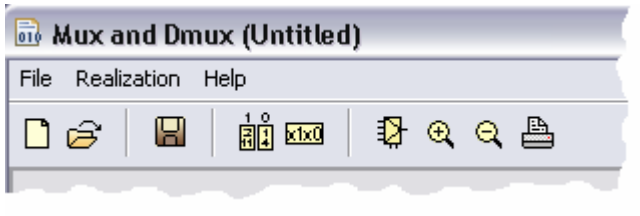


Fig. 2. Toolbar options

Toolbar options:

	– Start a new project
	– Open a project definition
	– Save the project in a project definition file
	– Canonical numerical representation of a function
	– Literal representation of a function
	– Realization of a function
	– Zoom into a diagram
	– Zoom out a diagram
	– Print generated project

The program menu hierarchy is presented in Fig. 3 and Fig. 4.

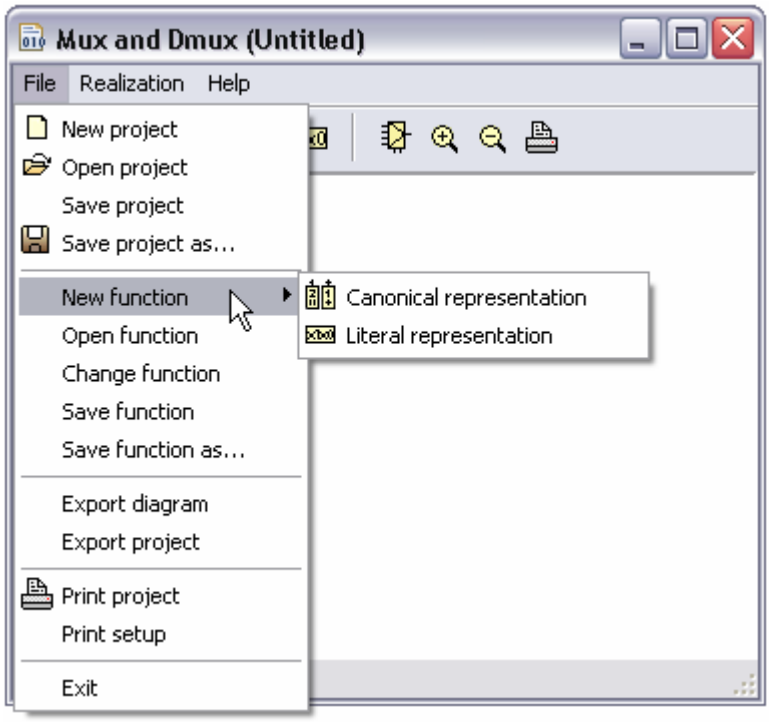


Fig. 3. File menu item hierarchy

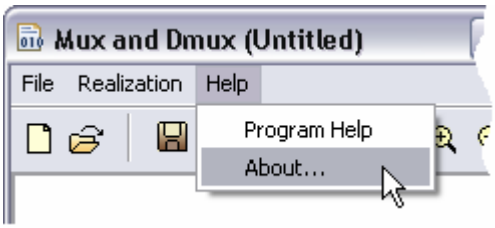


Fig. 4. Help menu item hierarchy

Description of menu items:

File – operations connected with entering function definition and files:

- New project – creates new project data, closing current one,
- Open project – opens existing project from a file,
- Save project – saves changes in the current project,
- Save project as... – saves the current project with a new name,
- New function – creates new function definition:
  - Canonical representation – allows to enter the function in canonical numerical representation,
  - Literal representation – allows to enter the function in literal representation,
- Open function – opens existing function definition from a file,
- Change function – changes currently entered function,

- Save function – saves changes in the current function definition file,
- Save function as... – saves the current function definition with a new name,
- Export diagram – exports diagram to graphical format,
- Export project – exports diagram and function definition to graphical format,
- Print project – prints the current project,
- Print setup – printer setup,
- Exit – closes the program,

Realization – realization of a function entered,
Help – help with the program:

- Program Help – displays program help,
- About... – short information about the program.

Entering function definition
Function may be entered in two forms:

- Canonical numerical decimal representation,
- Literal representation.

In *canonical numerical representation* function definition can be entered by proceeding menu steps **File ⇒ New function ⇒ Canonical representation** or by clicking "*New project*"[1] icon  and then "*Canonical representation*" icon  from the toolbar. Afterwards the window that enables entering data (Fig. 5) shows up.

The window consists of three lists of values:

- **Ones**
- **Zeros**
- **Don't cares**

[1] If currently open project or function definition is not saved user will be prompted to do it.

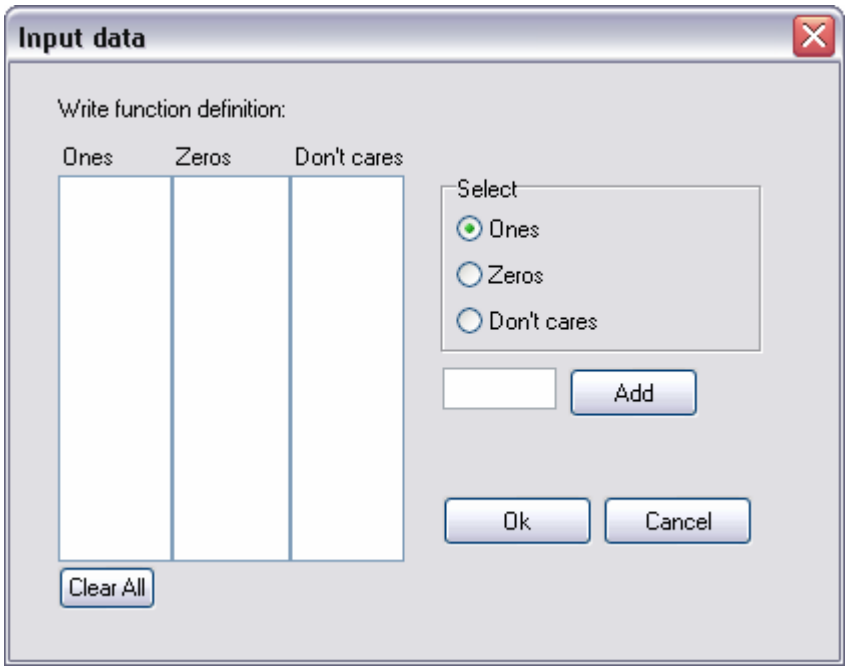
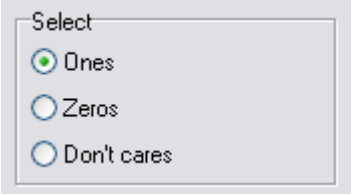


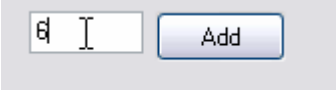
Fig. 5. Window that enables entering function definition in canonical form

Following steps should be performed to enter data to each of these lists:

1. Select to which list the value should be added, by clicking on one of radio buttons.



2. Write value in edit field.



3. Push "**Add**" button.

According to these steps all function components (ones, zeros and don't cares) have to be entered to these lists. The maximal implemented number of function variables is 10 so maximal possible value that can be entered in the edit field is **1023**, minimum is **0**.

To apply the changes "**OK**" button should be clicked. After these steps the middle column describing "**Zeros**" is filled up with values supplementary to the function entered. Thus, all function terms that are not present in "**Ones**" or "**Don't cares**" are put into "**Zeros**" list. In order to delete a term from one of the lists user should select this value, click right mouse button, then from popup

menu (Fig. 6) select "**Delete**". The same operation can be performed by selecting a term and pressing "**Del**" keyboard key.

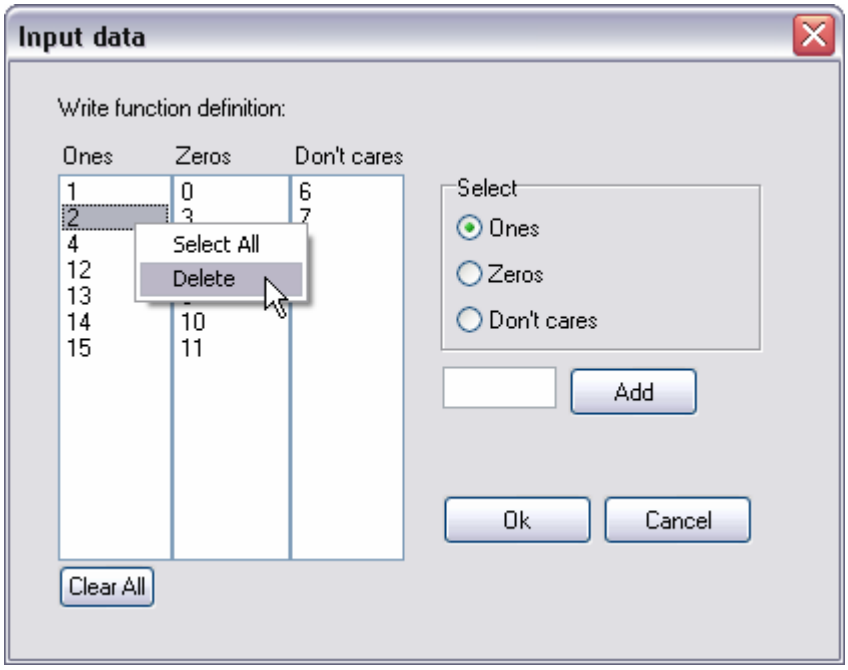


Fig. 6. Deleting term from the list

A list can be erased by selecting this list (by clicking in its area) and choosing in the popup menu "**Select All**" option and then clicking "**Del**" keyboard key or choosing "**Delete**" from the popup menu. All terms from all lists may also be cleared by selecting "**Clear all**" button. If there is a need to delete more than one value from one list the user should push and hold CTRL keyboard key and select as many values as they need. Then they should release the CTRL key, click right mouse button and select "**Delete**". All selected lists will be erased.

In order to change previously entered function definition it is advised to select the option from menu **File ⇒ Change function** or click on "*Canonical representation*" icon . To apply or to cancel changes made to the function definition button "**OK**" or "**Cancel**" should be pressed appropriately.

In *literal representation* function definition can be entered by proceeding menu steps **File ⇒ New function ⇒ Literal representation** or by clicking "*New project*" icon and then "*Literal representation*" icon . Afterwards the window that enables entering function definition (Fig. 7) shows up.

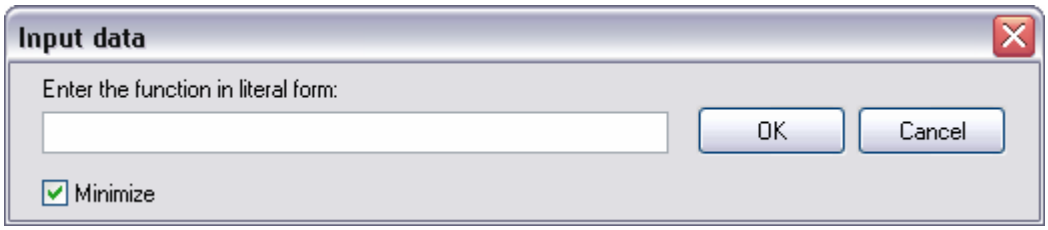


Fig. 7. Window that enables entering function definition in literal form

Function definition is written using characters: "~", "x", "+" and numbers 0, 1...9. The function should be rewritten in the form, where complemented value (i.e. $\overline{x_1}$) is represented by *tilde* sign and a variable (~x1). It is necessary to emphasize that function variables must be entered in descending order. Exemplary function that is required to be entered in literal form:

$F = (0,1,3)_{x_1 x_0} = \overline{x_1}\,\overline{x_0} + \overline{x_1} x_0 + x_1 x_0 \equiv$ **~x1~x0+~x1x0+x1x0**

Additionally, if user enters data in reversed order, the program will change the function according to decreasing order of variable indexes, i.e. $\overline{x_0}\,\overline{x_1} + \overline{x_0} x_2 + x_1 x_0 \equiv$ **~x1~x0+x2~x0+x1x0**.

In order to change previously entered function definition it is advised to select the option from menu **File ⇒ Change function** or click on "*Literal representation*" icon  . Editing a function definition is realized as in usual text fields, i.e. in word processor. A user can copy and paste text into text field in the window, as shown in Fig. 8. To apply or to cancel changes button "**OK**" or "**Cancel**" should be pressed appropriately.

As it was stated in overview, the function entered in canonical decimal form is minimized without reaction of the user, whereas function in literal form can be minimized on user request. As well as in entering a function in canonical decimal or in literal form[2], the function is minimized after clicking "**OK**". After minimization process function definition may change, thus when user needs to change a function it may look different than the one previously entered.

[2] In the window enabling entering function in literal form user has to check the checkbox in order to minimize the function.

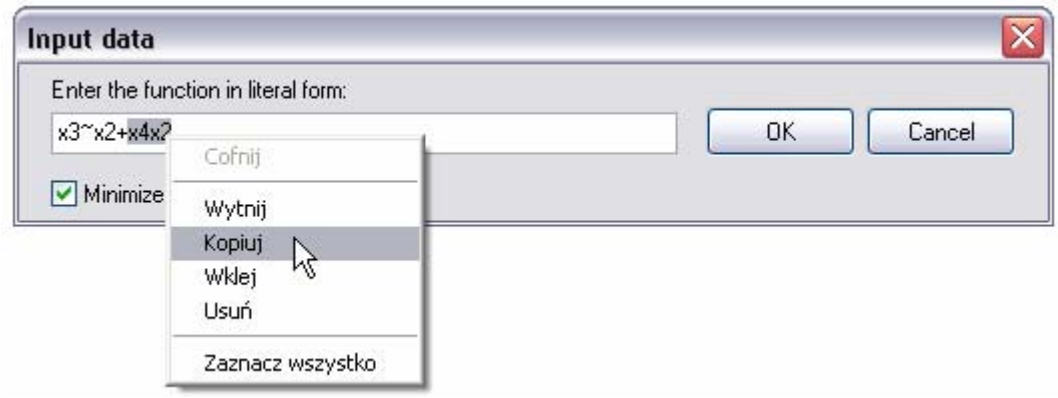


Fig. 8. Editing function definition entered in literal form

The progress bar window (Fig. 9) is displayed when there are eight or more independent function variables in window where function in literal form is entered or more than two hundred of function components in window where function in canonical decimal form is entered. For large number of data minimization process may last some few minutes. The progress bar window is displayed in order to visualize that the application is working properly.

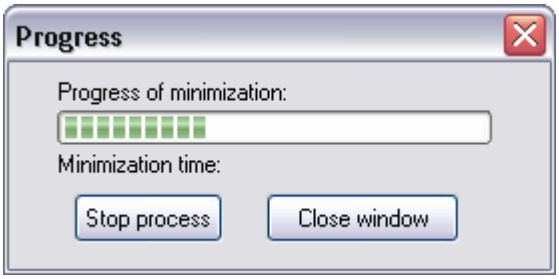


Fig. 9. Progress bar of minimization

When minimization is finished, window (Fig. 10) is displayed, where user can see minimization time. Then user can close the window.

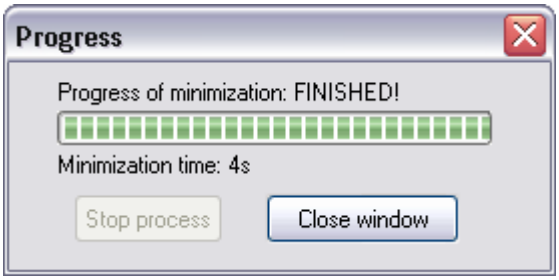


Fig. 10. Window displayed when minimization process is finished

If minimization time takes too much time, user can stop minimization process by clicking "**Stop process**" which causes the window (Fig. 11) to be displayed. If the process is stopped the function is left not minimized.

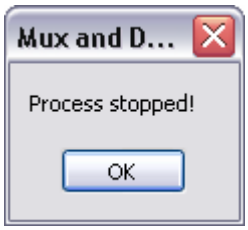


Fig. 11. Window displayed when minimization process is stopped

Structure realization

Program generates following structures:

- Multiplexer,
- Multiplexer with gates:
  - NAND gates only,
  - NOR gates only,
  - AND and OR and NOT gates,
- Demultiplexer,
- Tree of Multiplexers,
- Multiplexer – demultiplexer.

The structure is chosen by a user. The menu item which allows performing the structure realization is activated only when the function entered can be implemented[3]. Beginning of an implementation process is achieved by clicking **Realization** from the program menu or *realization* icon from the toolbar. It causes the window (Fig. 12) to be displayed.

[3] A function can be implemented in the program if it is not constant 0 or 1 after minimization.

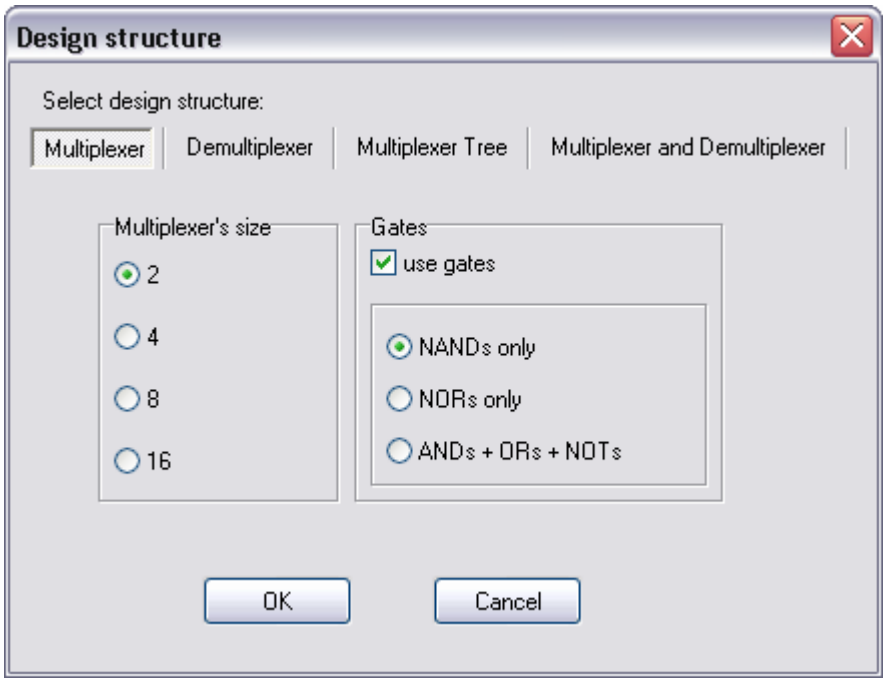


Fig. 12. Window that enables the structure selection

The window contains 4 buttons indicating the possible structures selections. These are: **Multiplexer**, **Demultiplexer**, **Tree of Multiplexers** and **Multiplexer – Demultiplexer**. Switching buttons causes that appropriate tab is displayed. Each tab represents parameters that should be set in order to generate a proper structure. For *Multiplexer* tab there exist a few of possible structures. Choosing different multiplexer's size user can obtain different results. Also marking "**use gates**" checkbox enables generating structures with different gates: NANDs only, NORs only or ANDs+ORs+NOTs. In order to obtain demultiplexer (or tree of multiplexers) structure from *Demultiplexer* (*Multiplexer Tree*) tab user should select demultiplexer's size. The size of multiplexer and demultiplexer ranges from 2 bits to 16 bits, as these are sizes of manufactured commutators nowadays. *Multiplexer and Demultiplexer* tab has two parameters – sizes of multiplexer and demultiplexer. As it is not always possible to generate a structure of a function of some number of variables (1 variable or more than 8) radio buttons that correspond to these impossible structures are disabled (Fig. 13). Also while clicking on the size whether of multiplexer or of demultiplexer, the second will change to enable proper generation of a Mux–Dmux structure.

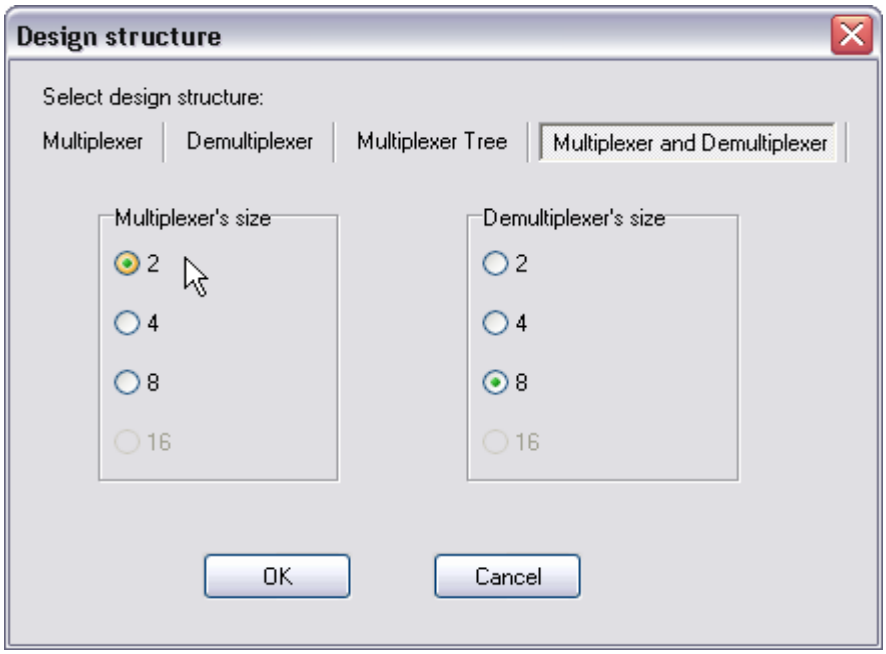


Fig. 13. Structure selection window with some options disabled

After applying the options by hitting "**OK**" button program generates a given structure and displays a diagram on the screen (exemplary result shown in Fig. 14).

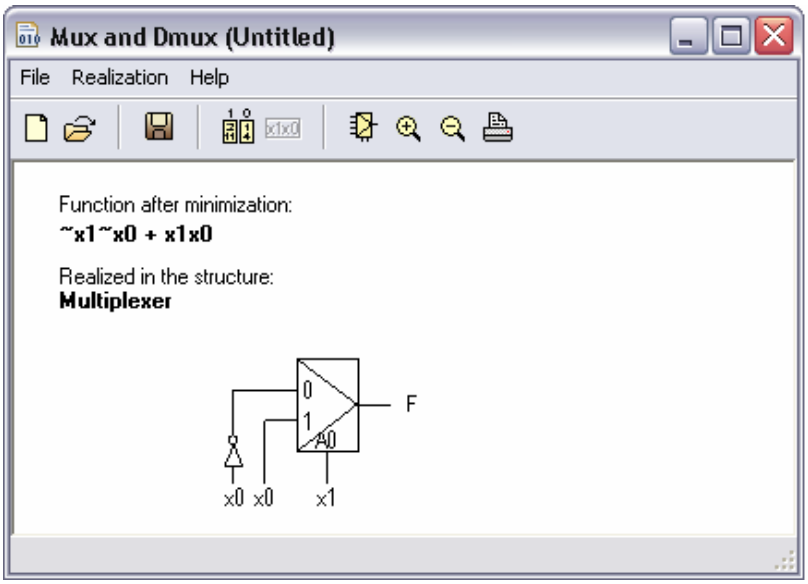


Fig. 14. Exemplary resulting structure

As it can be seen some toolbar options that were previously disabled are activated. The diagram can be zoomed in or out to facilitate viewing. The structure can be also changed by clicking **Realization** from the program menu or *realization* icon from the toolbar.

Saving and opening function definitions from a file

In order to save a function definition to a file a user should select **File ⇒ Save function as...** item. Choosing "**Save function**" option saves the current function to previously opened function definition file. If there was no previously opened file, while pressing "**Save function**", "**Save function as...**" dialog window appears. In this dialog user can enter a file name to be saved to (Fig. 15).

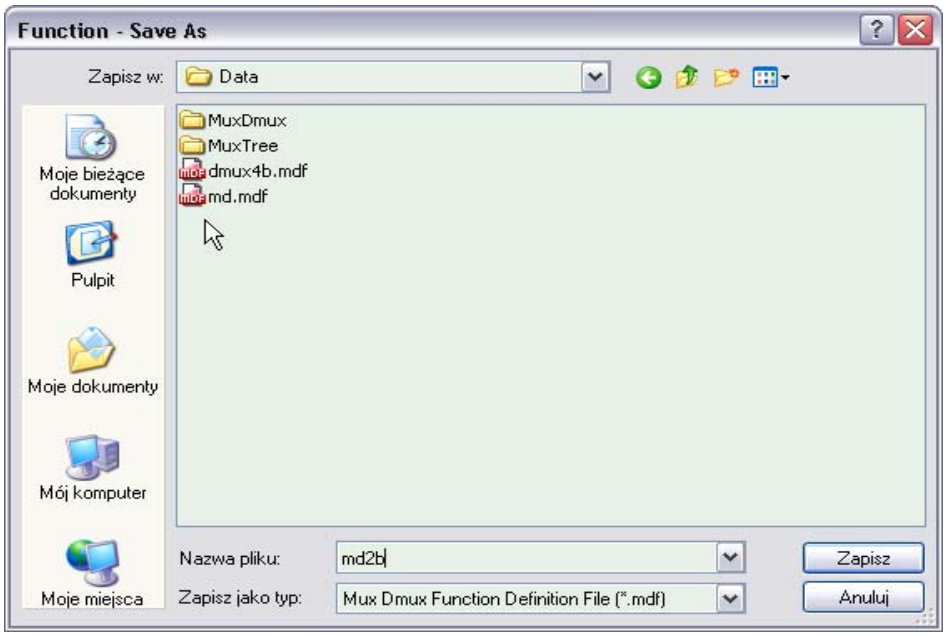


Fig. 15. Dialog window for saving a file

The file extension for function definition file is ".mdf". After the file is saved, at the bottom of the program window, on the status bar there appears the text "*File saved!*".

In order to open function definition from a file user should select **File ⇒ Open function**. Similarly to this in Fig. 15 the dialog window shows up where user can choose a file to be opened. If the current function definition in the program has been modified user is prompted to save it or discard these changes. Loading a function definition file enables the user to change the function or immediately implement this function.

In order to save project definition to a file user should select **File ⇒ Save project as...** item. Choosing "**Save project**" option saves the current function to previously opened project definition file. If there was no previously opened file the dialog window "**Save project as...**" appears where user can choose a file name to save to. The file extension for a project definition file is ".mdp". After the file is saved, at the bottom of the program window, on status bar "*File saved!*" text appears for three seconds.

In order to open project definition from a file user should select **File ⇒ Open project** item. Open dialog window (similar to Fig. 15) appears where user can choose a file. If the current function or project definition in the program has been modified user is prompted to save it or discard these changes. After loading a project definition file the program immediately generates the structure saved in the file.

Export and print of a project

A user has two options while exporting the design. The realized structure may be saved in a graphical format as a diagram file or as a project file. The difference is that the diagram file includes only diagram of a generated structure, while the project file contains a diagram as well as a minimized function definition. To export the generated structure user should select from program menu **File ⇒ Export diagram** or **File ⇒ Export project**.

The program enables exporting diagram/project to a vector (WMF) or raster (BMP, JPEG) graphic file format. For later use it is preferred to use vector format as it can be scaled without loosing quality.

The project can also be printed on a printer set up in **File ⇒ Print setup**. In order to print a project user should select **File ⇒ Print project** from the program menu or click toolbar icon  .

It has to be underlined that printing big structures (having more than five elements in one level) due to scaling may lead to illegible printouts. If user needs to obtain better results he is advised to use other graphical software and print the project on multiple pages.