

**DR. BOB DAVIDOV**

## **Построение графического интерфейса пользователя (GUI) в среде МатЛАБ для целей управления и наблюдением за состоянием объекта**

**Цель работы:** освоение правил построения графического интерфейса пользователя (Graphical User Interface “GUI”) в среде МатЛАБ..

**Задача работы:** построение интерфейса в среде МатЛАБ для обеспечения связи пользователя с устройством дискретного ввода-вывода МК110-224.8Д.4Р через OPC сервер.

**Приборы и принадлежности:** Персональный компьютер, МК110, преобразователь UPort1150 (USB/RS-485), МатЛАБ, OPC Toolbox, OPC DA сервер.

### **ОБЩИЕ СВЕДЕНИЯ**

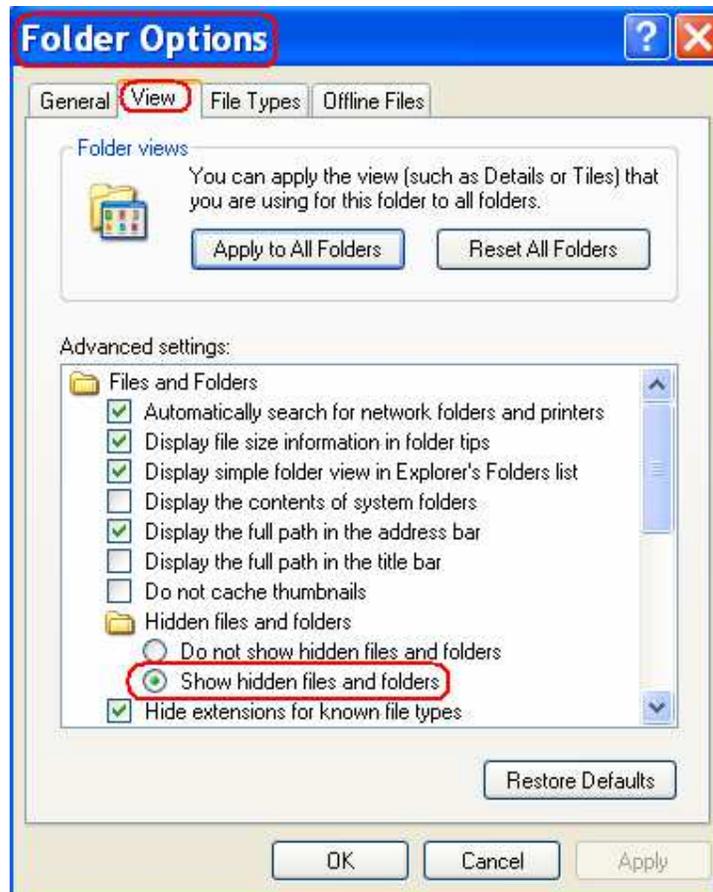
Интерфейс пользователя обеспечивает связь пользователя с программными и аппаратными ресурсами системы. Через интерфейс пользователь вводит заданные значения и принимает параметры системы. Интерфейс должен отображать состояние системы в наиболее удобном для восприятия виде.

Для выполнения работы необходимо убедиться в наличии OPC сервера на рабочем компьютере

  
со стартовой конфигурацией МК110\_ENG.mbc. Которая должна находиться в следующем разделе:



Если раздел невиден (спрятан), то его необходимо высветить командой “Show hidden files and folders”:



Убедитесь что драйвер UPort1150 (устройство связи USB порта компьютера с МК110) настроен на работу в режиме RS485.

Примеры интерфейсов, разработанные в среде MatLAB представлены на **Рис. 1** **Рис.2** и **Рис.4**.



Рис. 1. Интерфейс системы для тестирования магнитного поля объекта.



**Рис.2.** Магнитоэлектрический датчик с эталонной катушкой интерфейс которого представлен на **Рис. 3.**

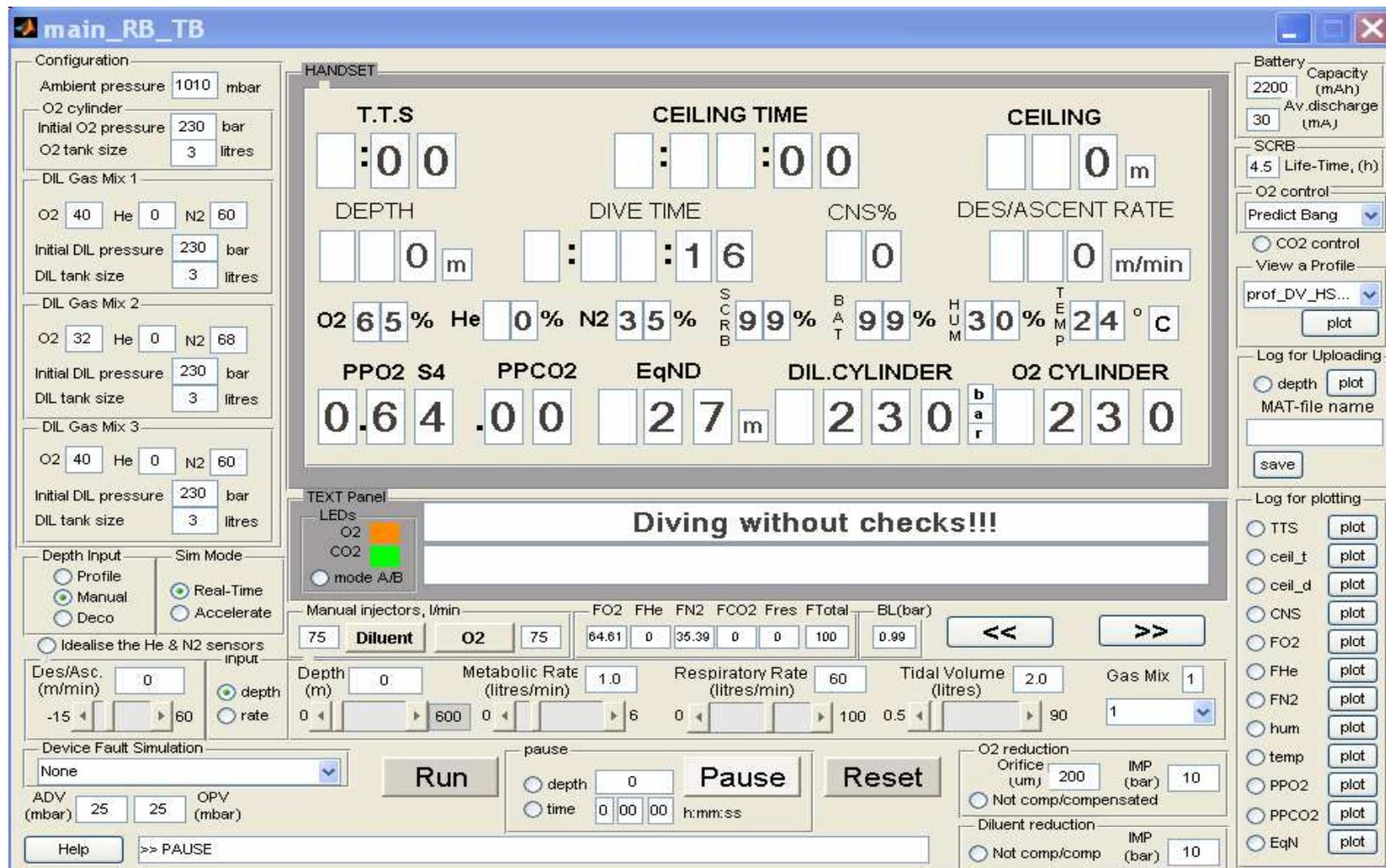


Рис.4. Интерфейс модели глубоководного аппарата.

## ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

**Задание 1.** Разработка графического интерфейса пользователя для связи с устройством дискретного ввода-вывода МК110.

В рамках этой работы необходимо построить интерфейс связи с МК110 который имеет

- восемь окон для отображения состояния счетных входов устройства. В первом окне отображается количество циклов связи с OPC сервером. В последнем 8-м окне отображается значение счетчика восьмого входа устройства МК110..
- четыре окна для ввода 4-х значений ШИМ в диапазоне от 1 до 1000.
- клавиша Reset для сброса выбранного счетчика.
- клавиша Run/Stop для записи установленного ШИМ или его нулевого значения в 4-й выход устройства.
- клавиши Connect и Disconnect для установки и разрыва связи с OPC сервером.

После установки связи с OPC устройством, программа должна каждую секунду отображать состояние восьмого входа устройства МК110. И записывать значение ШИМ в четвертый выход устройства после нажатия клавиши Run.

Пример интерфейса показан на рисунке ниже.

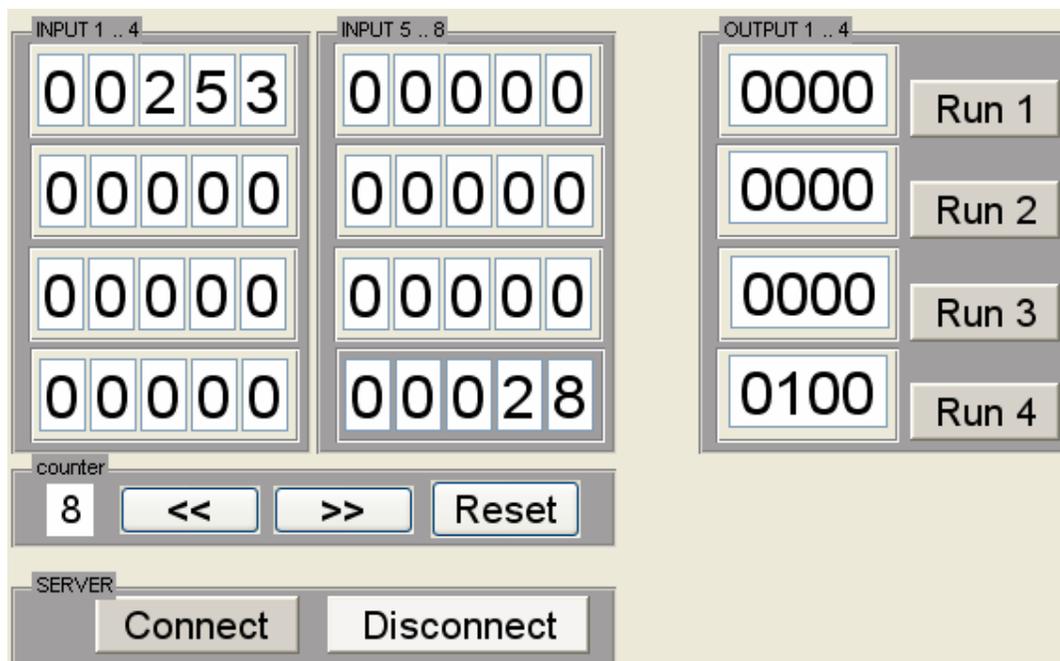
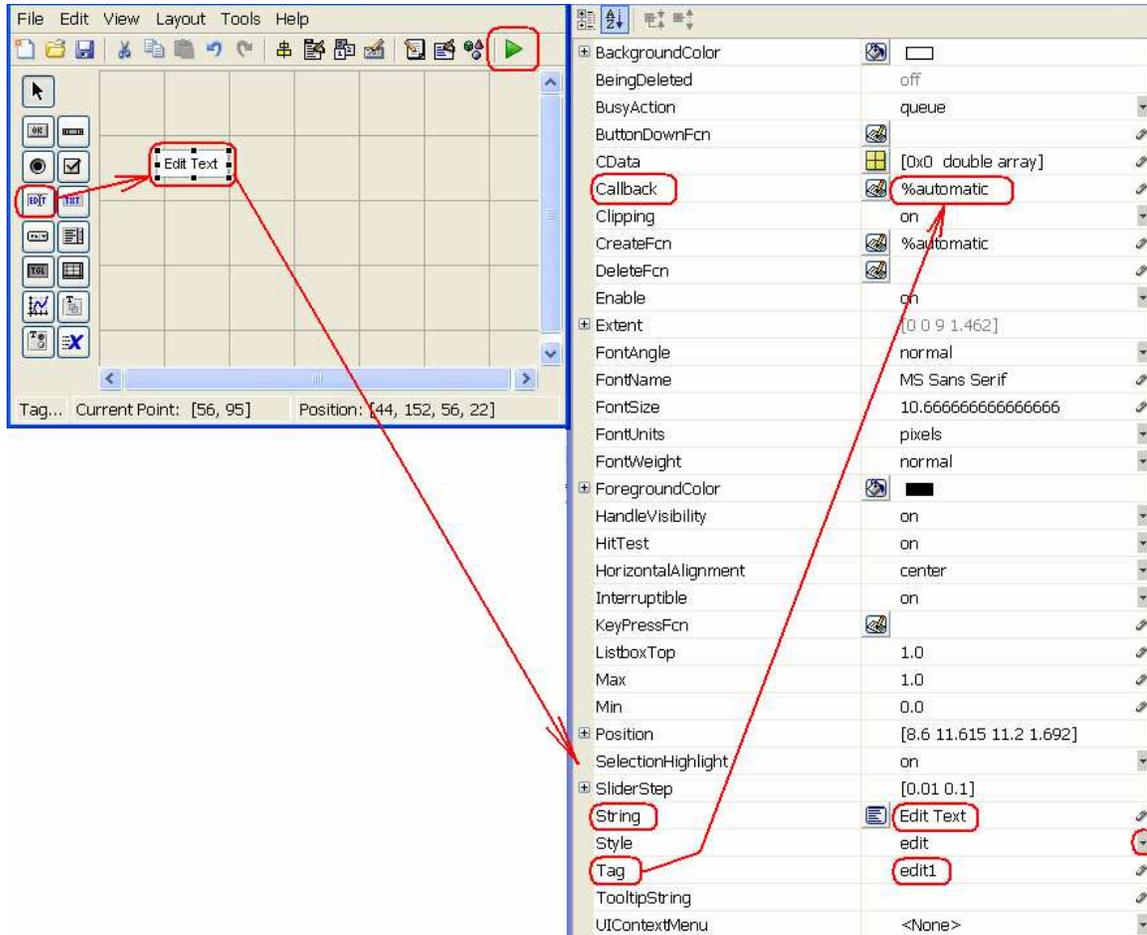


Рис. 5.Интерфейс связи с МК110.

Последовательность выполнения работы:

1. Загрузите МатЛАБ. Настройте ее на рабочую папку.
2. В командном окне наберите **guide** и затем нажмите клавишу <Enter>
3. В появившемся окне GUI Quick Start выберите закладку Create New GUI > Blank GUI (Default).

4. Дайте название проекту (файл fig), например, main\_MK110. В рабочей папке дополнительно к этому файлу появится файл main\_MK110.m. В этот файл система MatLAB автоматически добавляет функции для связи с каждым объектом GUI (кнопки, окна, меню, списки, графики и др.) которые будут перенесены из панели типовых объектов в рабочее пространство GUI.
5. Перенесите нужные объекты в поле GUI. С каждым объектом связан набор атрибутов. Наиболее важные из них следующие:
  - Tag – поле для ввода имени объекта, это имя автоматически присваивается соответствующей функции m-файла и вводится в поле Callback
  - String – надпись в поле объекта, которая отображается при загрузке GUI..



**Рис. 6.**Порядок составления GUI

6. Для построения рабочего варианта GUI и m-файла необходимо нажать на зеленый треугольник меню (см. **Рис. 6**).
7. Для установки и запуска таймера можно использовать следующие функции.
 

```
rt_timer = timer('TimerFcn', 'do_mode_MK110', 'ExecutionMode',
'FixedRate');
rt_timer.Period = 1;
rt_timer = timerfind;
rt_timer.TasksToExecute = 36000;
get(rt_timer, 'Running')
start(rt_timer)
stop(rt_timer)
```

8. Для установки связи с OPC, чтения и записи в устройство МК110 используются следующие команды.

```
da = opcda('localhost', 'InSAT.ModbusOPCServer.DA');
connect(da)

grp_in = addgroup(da);
additem(grp_in, 'COM_port.MK110.Counters.Input8');
r_8 = read (grp_in);

grp_out = addgroup(da);
additem(grp_out, 'COM_port.MK110.Outputs.Output4');
write(grp_out, 500)
write(grp_out, 0)

disconnect(da)
```

Пример фрагментов кода основной программы:

```
function main_MK110_OpeningFcn(hObject, eventdata, handles, varargin)
model_open(handles)
handles.output = hObject;
guidata(hObject, handles);

% -----
function model_open(handles)
global run_1 run_2 run_3 run_4 sim_state GUIparams counter da
run_1 = false;
in_panel_num = 8;
sim_state = 'stop';
counter = 0;

set(handles.uipanel18, 'BackgroundColor', [160 158 158] ./256)
set(handles.input_number, 'String', num2str(in_panel_num))

% Create new timer object
rt_timer = timer('TimerFcn', 'do_mode_MK110', 'ExecutionMode', 'FixedRate');
rt_timer.Period = 1; % Specifies the delay, in seconds, between executions

da = opcda('localhost', 'InSAT.ModbusOPCServer.DA');

function counter1_1_Callback(hObject, eventdata, handles)
function counter1_2_Callback(hObject, eventdata, handles)
function counter1_3_Callback(hObject, eventdata, handles)
function counter1_4_Callback(hObject, eventdata, handles)
function counter1_5_Callback(hObject, eventdata, handles)

function counter8_1_Callback(hObject, eventdata, handles)
function counter8_2_Callback(hObject, eventdata, handles)
function counter8_3_Callback(hObject, eventdata, handles)
function counter8_4_Callback(hObject, eventdata, handles)
function counter8_5_Callback(hObject, eventdata, handles)

% -----
```

```

function left_button_Callback(hObject, eventdata, handles)

NewStrVal = get(handles.input_number, 'String');
in_panel_num = str2num(NewStrVal);

in_panel_num = in_panel_num - 1;
if in_panel_num < 1
    in_panel_num = 8;
end
switch in_panel_num
    case 1
        set(handles.uipanel11, 'BackgroundColor', [160 158 158] ./256)
        set(handles.uipanel12, 'BackgroundColor', [236 233 216] ./255)
        ...
    case 8
        set(handles.uipanel18, 'BackgroundColor', [160 158 158] ./256)
        set(handles.uipanel11, 'BackgroundColor', [236 233 216] ./255)
end
set(handles.input_number, 'String', num2str(in_panel_num))

% -----
function reset_button_Callback(hObject, eventdata, handles)
global counter da

NewStrVal = get(handles.input_number, 'String');
in_panel_num = str2num(NewStrVal);

switch in_panel_num
    case 1
        counter = 0;

        set(handles.counter1_1, 'string', '0');
        set(handles.counter1_2, 'string', '0');
        set(handles.counter1_3, 'string', '0');
        set(handles.counter1_4, 'string', '0');
        set(handles.counter1_5, 'string', '0');

    case 8
        grp_in = addgroup(da);
        additem(grp_in, 'COM_port.MK110.Counters.Input8');

        if strcmp(da.status, 'connected')
            write(grp_in, 0) % reset counter 8
        end
end

% -----
function output_4_Callback(hObject, eventdata, handles)
NewStrVal = get(handles.output_4, 'String');
NewVal = str2num(NewStrVal);
if NewVal > 1000
    NewVal = 1000;

```

```

end
if NewVal < 0
    NewVal = 0;
end
NewStrVal = num2str(NewVal);
OutStrVal = '0000';
k = length(NewStrVal);
for i = 1:k
    OutStrVal (5-i) = NewStrVal (k+1-i);
end
set(handles.output_4,'String',OutStrVal)

% -----
function run_button_4_Callback(hObject, eventdata, handles)
global run_4 da grp_out

grp_out = addgroup(da);
additem(grp_out, 'COM_port.MK110.Outputs.Output4');

if ~run_4
    set(handles.run_button_4, 'BackgroundColor', [0.835294 0.815686 0.784314])
    %wavplay(param.DING_sound.y,param.DING_sound.Fs)
    NewStrVal = get(handles.output_4,'String');
    NewVal = str2num(NewStrVal);
    set(handles.run_button_4,'String','Run 4')

    write(grp_out, 0) % stop PWM 4

    run_4 = true;
else
    set(handles.run_button_4, 'BackgroundColor', [0.9608 0.9569 0.9529])
    set(handles.run_button_4,'String','Stop 4')

    NewStrVal = get(handles.output_4,'String');
    NewVal = str2num(NewStrVal);
    write(grp_out, NewVal) % start PWM 4

    run_4 = false;
end

% -----
function OPC_connect_button_Callback(hObject, eventdata, handles)
global sim_state GUIparams da grp_out

if strcmp(sim_state,'stop')
    % RT sim mode
    rt_timer = timerfind;
    rt_timer.TasksToExecute = 36000; % Specifies max possible number of clocks

    GUIparams = handles;
    if strcmp(get(rt_timer, 'Running'),'off')

```

```

        start(rt_timer) % start timer that run do_mode_MK110
    end
    set(handles.OPC_disconnect_button, 'BackgroundColor', [0.835294 0.815686 0.784314])
    set(handles.OPC_connect_button, 'BackgroundColor', [0.9608 0.9569 0.9529])
    sim_state = 'run';

    connect(da);
end

% -----
function OPC_disconnect_button_Callback(hObject, eventdata, handles)
global sim_state da

if strcmp(sim_state,'run')
    % stop RT mode
    rt_timer = timerfind;
    % if strcmp(get(rt_timer, 'Running'),'on')
    stop(rt_timer);
% end
    set(handles.OPC_connect_button, 'BackgroundColor', [0.835294 0.815686 0.784314])
    set(handles.OPC_disconnect_button, 'BackgroundColor', [0.9608 0.9569 0.9529])
    % wavplay(param.DING_sound.y,param.DING_sound.Fs)
    sim_state = 'stop';

    disconnect(da)
end

```

Пример кода подпрограммы вызываемой таймером:

```

function do_mode_MK110
global GUIparams counter da

handles = GUIparams;

grp_in = addgroup(da);
additem(grp_in, 'COM_port.MK110.Counters.Input8');
connect(da)

r_8 = read (grp_in);
str_counter = num2str (r_8.Value);

k = length(str_counter);
for i = 1:k
    switch i
        case 1
            set(handles.counter8_1, 'string', str_counter (k));
        case 2
            set(handles.counter8_2, 'string', str_counter (k-1));
        case 3
            set(handles.counter8_3, 'string', str_counter (k-2));
        case 4
            set(handles.counter8_4, 'string', str_counter (k-3));
    end
end

```

```
        case 5
            set(handles.counter8_5, 'string', str_counter (k-4));
        end
    end
end

GUIparams = handles;
```

### **КОНТРОЛЬНЫЕ ВОПРОСЫ**

1. Назовите возможные варианты применения графического интерфейса пользователя GUI?
2. Каким образом задается требуемая частота связи GUI с устройством МК 100?
3. Перечислите основные компоненты канала связи GUI - устройство МК.

### **БИБЛИОГРАФИЧЕСКИЙ СПИСОК**

1. Help система МатЛАБ
2. Инструкция по эксплуатации МК110-224.8Д.4Р
3. Инструкция по установке драйвера преобразователя. UPort1150.
4. Инструкция по установке ModBUS OPC сервера компании ИнСАТ.