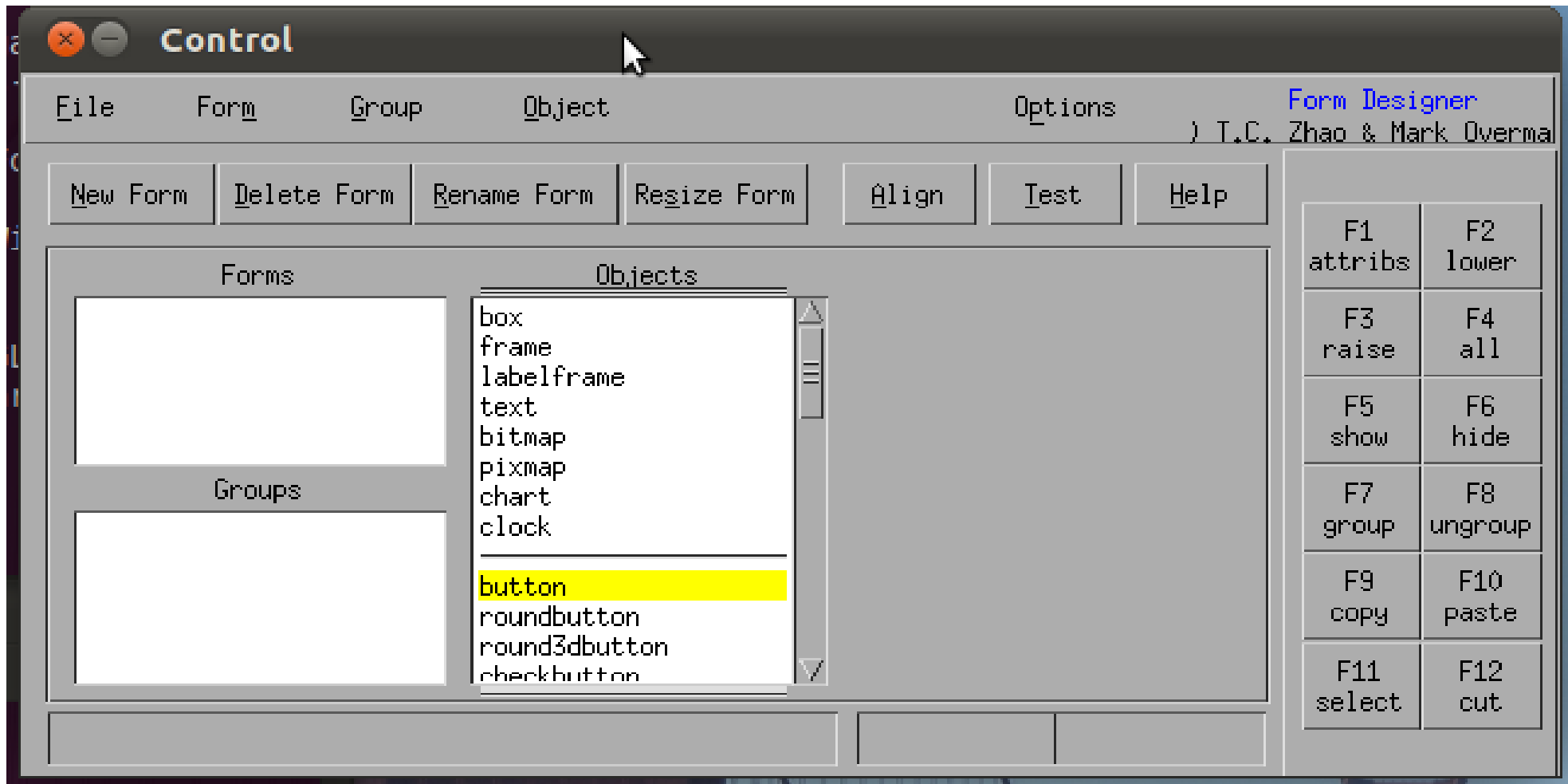


Graphische Benutzeroberfläche mit libforms

- Relativ einfache Programmierung
- Überschaubarer Vorrat an Controls
- Graphischer Designer verfügbar (fdesign)
 - Recht eigenwillige Bedienung
 - Generiert alle notwendigen Quellfiles
 - Eventhandling mit call-back-Funktionen
- Zu installierende Pakete (.deb)
 - libforms-doc, libforms-bin, libforms-dev
 - Doku unter
 /usr/share/doc/libforms-doc/html

Der Designer



Voreinstellungen

- Zu Beginn im Hauptfenster unter Options selektieren:
 - emit Main
 - emit emit Callback
 - emit C UI Code
- Nach save kann fdesign geschlossen werden.
- Soll an der Oberfläche nachträglich geändert werden, so sollten nachfolgende Optionen wieder deselektiert werden.
 - emit Main
 - emit emit Callback

Arbeitsschritte

1. new Form

TheApp (Name eintragen in Dialogfeld),
Das Hauptfenster wird sichtbar.

Doppelclick **rechts!!**

Dialog öffnet sich:

Name eintragen (theApp)

RESIZE_ALL auswählen

Fontgröße wählen

Name

Attributes

Generic Spec

Basic Attributes

Type

BoxType Up box

Label

Name mainWin

Callback

Argument

Shortcut

Font

Font helvetica-medium-r

Style Normal

Size 14 Medium

Misc.

LabelAlign CENTER

In/Out Outside

NW Gravity NoGravity

SE Gravity NoGravity

Resize RESIZE_ALL

Color

Color1

Color 2

LabelColor

Accept

Cancel

Restore

Form Design

Dialog öffnet sich
nach Doppelclick
rechte Maustaste
im rot umrandeten
Bereich unten

Hauptfenster
Im Designer

Einfügen weiterer Controls/ Widgets

2. input selektieren in Objects

Click in Form Designer

Aufziehen input Control

Doppelclick re (oder Button F1 Attribs)

Name eintragen

Callback eintragen

ev. Font eintragen

3. Browser selektieren in Objects

Click in Form Designer

Aufziehen input Control

Doppelclick re (oder Button F1 Attribs)

Name eintragen

Callback eintragen

ev. Font eintragen

4. Button selektieren in Objects

Click in Form Designer

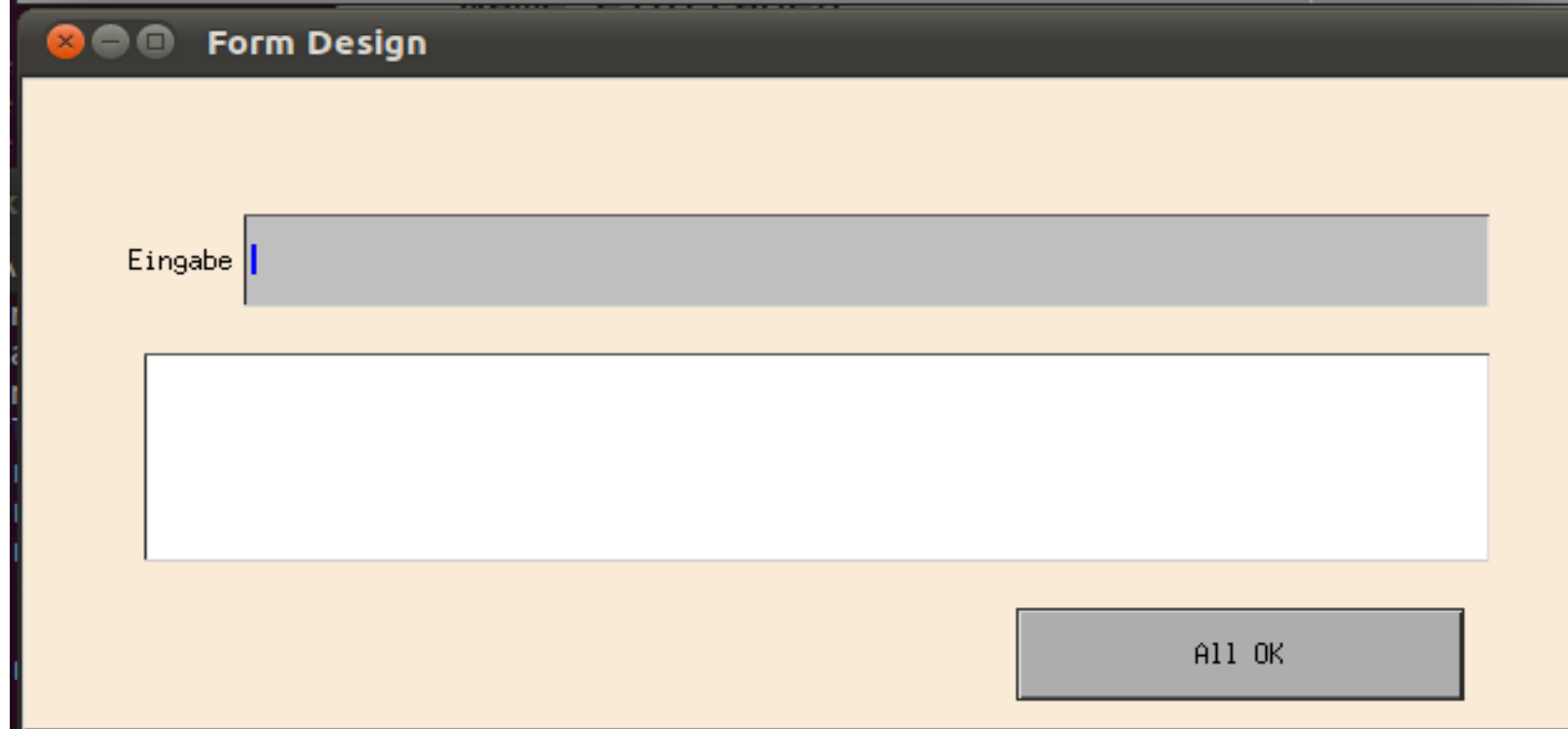
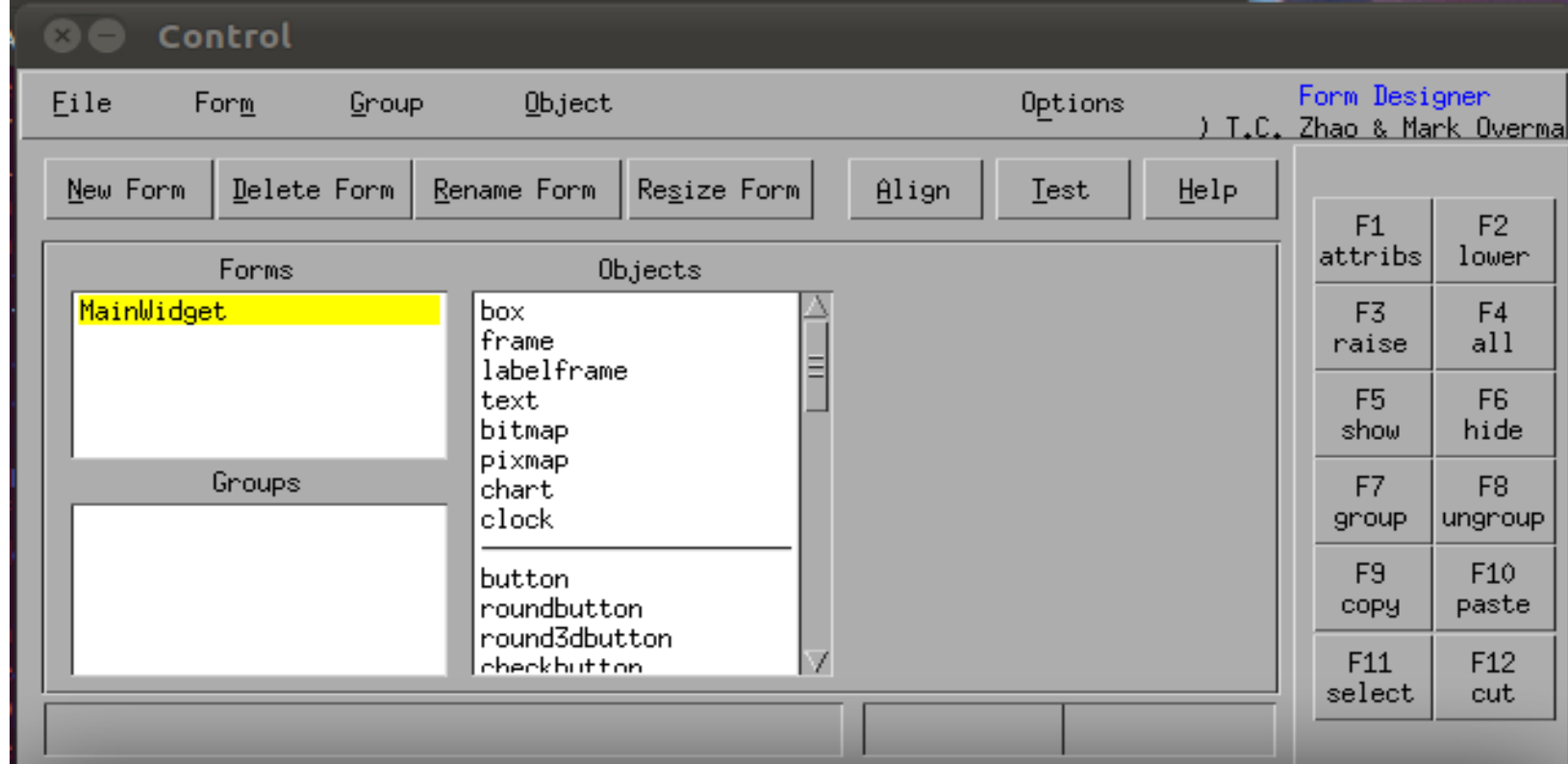
Aufziehen input Control

Doppelclick re (oder Button F1 Attribs)

Name eintragen

Callback eintragen

ev. Font eintragen



```

#include "v1.h"
FD_MainWidget *fd_MainWidget;
int
main( int   argc,
      char * argv[ ] )
{
    FD_MainWidget *fd_MainWidget;
    fl_initialize( &argc, argv, 0, 0, 0 );
    fd_MainWidget = create_form_MainWidget( );

    /* Fill-in form initialization code */

    /* Show the first form */

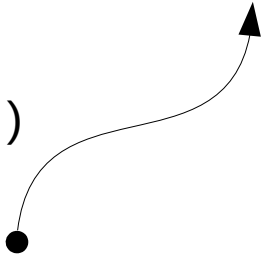
    fl_show_form( fd_MainWidget->MainWidget, . . . . "MainWidget" );

    fl_do_forms( );

    if ( fl_form_is_visible( fd_MainWidget->MainWidget ) )
        fl_hide_form( fd_MainWidget->MainWidget );
    fl_free( fd_MainWidget );
    fl_finish( );

    return 0;
}

```



/* Header file generated by fdesign on Mon Jan 30 17:07:12 2012 */

```
#ifndef FD_MainWidget_h_  
#define FD_MainWidget_h_
```

```
#include <forms.h>
```

```
/* Callbacks, globals and object handlers */
```

```
extern void callIn1( FL_OBJECT *, long );  
extern void callButt1( FL_OBJECT *, long );
```

```
/* Forms and Objects */
```

```
typedef struct {  
    FL_FORM  * MainWidget;  
    void      * vdata;  
    char      * cdata;  
    long      ldata;  
    FL_OBJECT * Mywidget;  
    FL_OBJECT * In1;  
    FL_OBJECT * Butt1;  
    FL_OBJECT * MyBrowser;  
} FD_MainWidget;
```

```
extern FD_MainWidget * create_form_MainWidget( void );  
extern FD_MainWidget *fd_MainWidget;  
#endif /* FD_MainWidget_h_ */
```



ergänzen!

/* Form definition file generated by fdesign */

#include <stdlib.h>

#include "v1.h"

/*

*/

FD_MainWidget *

create_form_MainWidget(void)

{
 FL_OBJECT *obj;
 FD_MainWidget *fdui = fl_malloc(sizeof *fdui);

 fdui->vdata = fdui->cdata = NULL;
 fdui->ldata = 0;

 fdui->MainWidget = fl_bgn_form(FL_NO_BOX, 627, 284);

 fdui->Mywidget = obj = fl_add_box(FL_UP_BOX, 0, 0, 627, 284, "");
 fl_set_object_color(obj, FL_ANTIQUWHITE, FL_BISQUE);

 fdui->In1 = obj = fl_add_input(FL_NORMAL_INPUT, 90, 60, 500, 40, "Eingabe");
 fl_set_object_callback(obj, callIn1, 0);
 fl_set_object_return(obj, FL_RETURN_END_CHANGED);

 fdui->Butt1 = obj = fl_add_button(FL_NORMAL_BUTTON, 400, 230, 180, 40, "All OK");
 fl_set_object_lalign(obj, FL_ALIGN_CENTER);
 fl_set_object_callback(obj, callButt1, 0);

 fdui->MyBrowser = obj = fl_add_browser(FL_NORMAL_BROWSER, 50, 120, 540, 90, "");
 fl_set_object_return(obj, FL_RETURN_CHANGED | FL_RETURN_SELECTION | FL_RETURN_DESELECTION);

 fl_end_form();

 fdui->MainWidget->fdui = fdui;

 return fdui;

}

```
#include <stdlib.h>
```

```
#include "v1.h"
```

```
/* Callbacks and freeobj handles for form MainWidget */
```

```
/*  
*****  
******/
```

```
void callIn1( FL_OBJECT * ob,  
             long      data )
```

```
{
```

```
/* Fill-in code for callback here */
```

```
const char * p=fl_get_input(ob);
```

```
printf("callIn1  called: %s\n",p);
```

```
fl_add_browser_line(fd_MainWidget->MyBrowser,p);
```

```
fl_set_input(ob,"");
```

```
}
```

```
/*  
*****  
******/
```


```
void callButt1( FL_OBJECT * ob,  
              long      data )
```

```
{
```

```
/* Fill-in code for callback here */
```

```
exit(0);
```

```
}
```



Doku unter:
</usr/share/doc/libforms-doc>