

Software code

DCFlow.exe

.h header files

global.h

Unit1.h

Unit2.h

Unit3.h

```

1:
2: /**
3: /**
4: /** ----- /**
5: /** Filename: global.h /**
6: /** Part of: DCFLOW.exe /**
7: /** /**
8: /** Compiler: Borland C++ Builder 6 servicepack 4 /**
9: /** CPort VCL , JVCL /**
10: /** Made by: Eric Halmans /**
11: /** For: Fontys Highschool Eindhoven, Mechatronica /**
12: /** Date: April 2006 /**
13: /** Version: 1.0 beta test version /**
14: /** /**
15: /** Description: /**
16: /** This file is part a windows program, /**
17: /** which is used for universal AC & DC motor control, /**
18: /** with a HCS12 TBoard from Elektronik Laden /**
19: /** (Hbridge with PWM on PP0 & PP1, SCI1 input,IIC Output) /**
20: /** ----- /**
21: /** /**
22: /**
23:
24:
25: #ifndef globalh
26: #define globalh
27:
28: // global variables
29:
30:
31: bool Button1True= false;
32:
33: bool ConnectionTrue= false;
34:
35: void Delay(void)
36: {
37:     int i;
38:     for (i=0;i<10;i++)
39:         // MaxInt =2^32-1
40:         {
41:             ;
42:         }
43: }
44:
45: #endif
46:

```

```

1:
2: /**
3: /**
4: /** ----- /**
5: /** Filename: Unit1.h /**
6: /** Part of: DCFlow.exe /**
7: /** /**
8: /** Compiler: Borland C++ Builder 6 servicepack 4 /**
9: /** CPort VCL , JVCL /**
10: /** Made by: Eric Halmans /**
11: /** For: Fontys Highschool Eindhoven, Mechatronica /**
12: /** Date: April 2006 /**
13: /** Version: 1.0 beta test version /**
14: /** /**
15: /** Description: /**
16: /** This file is part a windows program, /**
17: /** which is used for universal AC & DC motor control, /**
18: /** with a HCS12 TBoard from Elektronik Laden /**
19: /** (Hbridge with PWM on PP0 & PP1, SCI1 input,IIC Output) /**
20: /** ----- /**
21: /** /**
22: /**
23:
24:
25: #ifndef Unit1H
26: #define Unit1H
27: /** -----
28: #include <Classes.hpp>
29: #include <Controls.hpp>
30: #include <StdCtrls.hpp>
31: #include <Forms.hpp>
32: #include "JvComponent.hpp"
33: #include "JvDialButton.hpp"
34: #include "JvExControls.hpp"
35: #include "JvSegmentedLEDDisplay.hpp"
36: #include <ExtCtrls.hpp>
37: #include "CPort.hpp"
38: #include "JvTimer.hpp"
39: #include <Menus.hpp>
40:
41: /** -----
42: class TForm1 : public TForm
43: {
44: published: // IDE-managed Components
45:     TLabel *Label1;
46:     TButton *Button1;
47:     TJvDialButton *JvDialButton1;
48:     TPanel *Panel1;
49:     TJvSegmentedLEDDisplay *JvSegmentedLEDDisplay1;
50:     TComPort *ComPort1;
51:     TMainMenu *MainMenu1;
52:     TMenuItem *N1;
53:     TMenuItem *Exit1;
54:     TMenuItem *Connections1;
55:     TMenuItem *ComPort2;
56:     TMenuItem *Costumbaudratel;
57:     TMenuItem *Help1;
58:     TMenuItem *Help2;
59:     TMenuItem *About1;
60:     TLabel *Label2;

```

```
61:     TMenuItem *Connect1;
62:     TTimer *Timer1;
63:     TLabel *Label3;
64:     TLabel *Label4;
65:     TLabel *Label5;
66:     TLabel *Label6;
67:     void __fastcall JvDialButton1Change(TObject *Sender);
68:     void __fastcall Button1Click(TObject *Sender);
69:     void __fastcall FormActivate(TObject *Sender);
70:     void __fastcall ComPort2Click(TObject *Sender);
71:     void __fastcall Exit1Click(TObject *Sender);
72:     void __fastcall About1Click(TObject *Sender);
73:     void __fastcall Costumbaudrate1Click(TObject *Sender);
74:     void __fastcall Connect1Click(TObject *Sender);
75:     void __fastcall Timer1Timer(TObject *Sender);
76:     void __fastcall FormDestroy(TObject *Sender);
77: private:    // User declarations
78: public:    // User declarations
79:     __fastcall TForm1(TComponent* Owner);
80: };
81: //-----
82: extern PACKAGE TForm1 *Form1;
83: //-----
84: #endif
85:
86: //-----
87: // end of file Unit1.h
88:
89:
```

```

1:
2: /**
3: /**
4: /** ----- /**
5: /** Filename: Unit2.h /**
6: /** Part of: DCFlow.exe /**
7: /** /**
8: /** Compiler: Borland C++ Builder 6 servicepack 4 /**
9: /** CPort VCL , JVCL /**
10: /** Made by: Eric Halmans /**
11: /** For: Fontys Highschool Eindhoven, Mechatronica /**
12: /** Date: April 2006 /**
13: /** Version: 1.0 beta test version /**
14: /** /**
15: /** Description: /**
16: /** This file is part a windows program, /**
17: /** which is used for universal AC & DC motor control, /**
18: /** with a HCS12 TBoard from Elektronik Laden /**
19: /** (Hbridge with PWM on PP0 & PP1, SCI1 input,IIC Output) /**
20: /** ----- /**
21: /** /**
22: /**
23:
24:
25: #ifndef Unit2H
26: #define Unit2H
27: /**-----
28: #include <Classes.hpp>
29: #include <Controls.hpp>
30: #include <StdCtrls.hpp>
31: #include <Forms.hpp>
32: /**-----
33: class TForm2 : public TForm
34: {
35: __published: // IDE-managed Components
36:     TLabel *Label1;
37:     TLabel *Label2;
38:     TLabel *Label3;
39:     TLabel *Label4;
40:     TLabel *Label5;
41:     TLabel *Label6;
42:     TLabel *Label7;
43:     TButton *Button1;
44:     void __fastcall Button1Click(TObject *Sender);
45: private: // User declarations
46: public: // User declarations
47:     __fastcall TForm2(TComponent* Owner);
48: };
49: /**-----
50: extern PACKAGE TForm2 *Form2;
51: /**-----
52: #endif
53:
54: /**-----
55: /** end of file Unit2.h
56:

```

```

1:
2: /**
3: /**
4: /** ----- /**
5: /** Filename: Unit3.h /**
6: /** Part of: DCFlow.exe /**
7: /** /**
8: /** Compiler: Borland C++ Builder 6 servicepack 4 /**
9: /** CPort VCL , JVCL /**
10: /** Made by: Eric Halmans /**
11: /** For: Fontys Highschool Eindhoven, Mechatronica /**
12: /** Date: April 2006 /**
13: /** Version: 1.0 beta test version /**
14: /** /**
15: /** Description: /**
16: /** This file is part a windows program, /**
17: /** which is used for universal AC & DC motor control, /**
18: /** with a HCS12 TBoard from Elektronik Laden /**
19: /** (Hbridge with PWM on PP0 & PP1, SCI1 input,IIC Output) /**
20: /** ----- /**
21: /** /**
22: /**
23:
24: /**-----
25: #include "Unit1.h"
26: #include <Classes.hpp>
27: #include <Controls.hpp>
28: #include <StdCtrls.hpp>
29: #include "JvComponent.hpp"
30: #include "JvEDIDBBuffering.hpp"
31:
32: #ifndef Unit3H
33: #define Unit3H
34: /**-----
35: #include <Classes.hpp>
36: #include <Controls.hpp>
37: #include <StdCtrls.hpp>
38: #include <Forms.hpp>
39:
40: /**-----
41: class TForm3 : public TForm
42: {
43: published: // IDE-managed Components
44:     TLabel *Label1;
45:     TEdit *Edit1;
46:     TButton *Button1;
47:     TLabel *Label2;
48:     TLabel *Label3;
49:     void __fastcall Button1Click(TObject *Sender);
50: private: // User declarations
51: public: // User declarations
52:     __fastcall TForm3(TComponent* Owner);
53: };
54: /**-----
55: extern PACKAGE TForm3 *Form3;
56: /**-----
57: #endif
58:
59: /**-----
60: /** end of file Unit3.h

```

Software code

DCFlow.exe

.cpp code files

DCFlow.cpp

Unit1.cpp

Unit2.cpp

Unit3.cpp

```

1: //-----
2:
3: #include <vcl.h>
4: #pragma hdrstop
5: //-----
6: USEFORM( "Unit1.cpp", Form1);
7: USEFORM( "Unit2.cpp", Form2);
8: USEFORM( "Unit3.cpp", Form3);
9: //-----
10: WINAPI WinMain(HINSTANCE, HINSTANCE, LPSTR, int)
11: {
12:     try
13:     {
14:         Application->Initialize();
15:         Application->Title = "DCMOTOR";
16:         Application->CreateForm(__classid(TForm1), &Form1);
17:         Application->CreateForm(__classid(TForm2), &Form2);
18:         Application->CreateForm(__classid(TForm3), &Form3);
19:         Application->Run();
20:     }
21:     catch (Exception &exception)
22:     {
23:         Application->ShowException(&exception);
24:     }
25:     catch (...)
26:     {
27:         try
28:         {
29:             throw Exception("");
30:         }
31:         catch (Exception &exception)
32:         {
33:             Application->ShowException(&exception);
34:         }
35:     }
36:     return 0;
37: }
38: //-----
39:

```



```

1:
2: /**
3: /**
4: /** ----- *
5: /** Filename: Unit1.cpp *
6: /** Part of: DCFlow.exe *
7: /** *
8: /** Compiler: Borland C++ Builder 6 servicepack 4 *
9: /** CPort VCL , JVCL *
10: /** Made by: Eric Halmans *
11: /** For: Fontys Highschool Eindhoven, Mechatronica *
12: /** Date: April 2006 *
13: /** Version: 1.0 beta test version *
14: /** *
15: /** Description: *
16: /** This file is part a windows program, *
17: /** which is used for universal AC & DC motor control, *
18: /** with a HCS12 TBoard from Elektronik Laden *
19: /** (Hbridge with PWM on PP0 & PP1, SCI1 input,IIC Output) *
20: /** ----- *
21: /** *
22: /**
23:
24: #include <vcl.h>
25: #pragma hdrstop
26:
27: #include "global.h"
28: #include "Unit1.h"
29: #include "Unit2.h"
30: #include "Unit3.h"
31:
32: //-----
33: #pragma package(smart_init)
34: #pragma link "JvComponent"
35: #pragma link "JvDialButton"
36: #pragma link "JvExControls"
37: #pragma link "JvSegmentedLEDDisplay"
38: #pragma link "CPort"
39: #pragma link "JvTimer"
40: #pragma resource "*.dfm"
41: TForm1 *Form1;
42:
43:
44: //-----
45: fastcall TForm1::TForm1(TComponent* Owner)
46:     : TForm(Owner)
47: {
48: }
49:
50: //-----
51:
52: // Short description:
53:
54: // If JvDialButton1Change changes, JvSegmentedLEDDisplay1
55: // changes according to the dialbutton
56:
57: // The dialbutton is divided in 3 zones,
58: // Zone 1: 256 steps PWM duty turn left
59: // Zone 2: Zero value zone
60: // Zone 3: 256 steps PWM duty turn right

```

```

61:
62: // JvSegmentedLEDDisplay1 changes color according to change of a zone,
63: // and displays the step value in %.
64:
65: // Pre: Old position of JvDialButton1
66:
67: // Post: new position of JvDialButton1
68: //      Changed value (and color) of JvSegmentedLEDDisplay1
69:
70: //-----
71: void __fastcall TForm1::JvDialButton1Change(TObject *Sender)
72: {
73:     if (JvDialButton1->Position<=256)
74:     {
75:         JvSegmentedLEDDisplay1->SegmentLitColor=clLime;
76:         JvSegmentedLEDDisplay1->Text=((256-JvDialButton1->Position)/2.56);
77:     }
78:     else if (JvDialButton1->Position>256 && JvDialButton1->Position<=344)
79:     {
80:         JvSegmentedLEDDisplay1->SegmentLitColor=clYellow;
81:         JvSegmentedLEDDisplay1->Text=0,0;
82:     }
83:     else if (JvDialButton1->Position>344)
84:     {
85:         JvSegmentedLEDDisplay1->SegmentLitColor=clRed;
86:         JvSegmentedLEDDisplay1->Text=((JvDialButton1->Position-344)/2.56);
87:     }
88: }
89:
90:
91: //-----
92:
93: // Short description:
94:
95: // When ever this button is clicked it sets the boolean Button1True either
96: // to be true or false. It also changes the caption according
97:
98: // Pre: Button1True is true or false
99: //
100: // Post: If Button1True = false : Button1True -> true
101: //       If Button1True = true  : Button1True -> false
102: //       Caption changes accordingly (Motor ON or Motor OFF)
103:
104: //-----
105: void __fastcall TForm1::Button1Click(TObject *Sender)
106: {
107:     if (Button1True)
108:     {
109:         Button1->Caption="MOTOR OFF";
110:         Button1True=false;
111:     }
112:     else if (!Button1True)
113:     {
114:         Button1->Caption="MOTOR ON";
115:         Button1True=true;
116:         JvDialButton1->Position=300;
117:     }
118: }
119:
120:

```

```
121: //-----
122:
123: // Short description:
124:
125: // When ever this program is started, this function is run first.
126: // At the moment it only sets the custom baudrate to 500.000
127:
128: // Pre: Program is not running
129: //
130: // Post: If program is started, CustomBaudRate is set to 500.000
131:
132: //-----
133: void __fastcall TForm1::FormActivate(TObject *Sender)
134: {
135:     ComPort1->CustomBaudRate=500000;
136: }
137:
138:
139: //-----
140:
141: // Short description:
142:
143: // Choosing this menu option, a setup dialog for the Comport VCL is shown.
144: // In this dialog, you can change:
145: // - Port number
146: // - Baudrate
147: // - Data bits
148: // - Stop bits
149: // - Parity
150: // - Flow control
151:
152: // Pre: ComPort1
153:
154: // Post: setup dialog shown and changes effect ComPort1
155:
156: //-----
157: void __fastcall TForm1::ComPort2Click(TObject *Sender)
158: {
159:     ComPort1->ShowSetupDialog();
160: }
161:
162:
163: //-----
164:
165: // Short description:
166:
167: // Program end
168:
169: // Pre: DCFlow.exe is running
170:
171: // Post: DCFlow.exe is stopped
172:
173: //-----
174: void __fastcall TForm1::Exit1Click(TObject *Sender)
175: {
176:     exit(0);
177: }
178:
179:
180: //-----
```

```

181: // Short description:
182:
183: // Show about window
184:
185: // Pre: DCFflow.exe is running
186:
187: // Post: On top of DCFflow.exe is the about window shown
188:
189: //-----
190: void __fastcall TForm1::About1Click(TObject *Sender)
191: {
192:     Form2->Show();
193: }
194:
195:
196:
197: //-----
198:
199: // Short description:
200:
201: // Show set custom baud
202:
203: // Pre: DCFflow.exe is running
204:
205: // Post: On top of DCFflow.exe is the custom baud window shown
206:
207: //-----
208: void __fastcall TForm1::Costumbaudrate1Click(TObject *Sender)
209: {
210:     Form3->Show();
211: }
212:
213:
214: //-----
215:
216: // Short description:
217:
218: // This menu option if checked opens a connection with the Comport VCL
219:
220: // Pre: If Connect1 = Checked Comport is open
221: //      If Connect1 = not Checked Comport is closed
222:
223: // Post: If Connect1 was Checked Comport is now closed
224: //       If Connect1 was not Checked Comport is now open
225:
226: //-----
227: void __fastcall TForm1::Connect1Click(TObject *Sender)
228: {
229:     if (ComPort1->Connected==0)
230:     {
231:         ConnectionTrue = true;
232:         Connect1->Checked = true;
233:         ComPort1->Open();
234:         Timer1->Enabled=true;
235:     }
236:     else if (ComPort1->Connected)
237:     {
238:         ConnectionTrue = false;
239:         Connect1->Checked = false;
240:         ComPort1->Close();

```

```

241:     Timer1->Enabled=false;
242: }
243: }
244:
245:
246: //-----
247:
248: // Short description:
249:
250: // If ComPort is open, and TimerInterval is reached,
251: // a maximum of 2 characters, are send to the HCS12.
252: // This is done by reading out the JvDialButton1->Position.
253:
254: // The dialbutton is divided in 3 zones,
255: // Zone 1: 256 steps PWM duty turn left
256: // Zone 2: Zero value zone
257: // Zone 3: 256 steps PWM duty turn right
258:
259: // If JvDialButton1->Position = Zone1:
260: //     first char  = 0x52;
261: //     Second char = 0x00..0xff (duty left)
262: // If JvDialButton1->Position = Zone2:
263: //     first char  = 0x00;  Stops PWM
264: //     NO Second char
265: // If JvDialButton1->Position = Zone3:
266: //     first char  = 0x4c;
267: //     Second char = 0x00..0xff (duty Right)
268:
269: // Pre: postion of JvDialButton1
270:
271: // Post: At timerinterval reached the fnction translates postion
272: //       of JvDialButton1 in a 2 character coded value
273: //       used as change of the PWM channels of the HCS12
274:
275: //-----
276: void __fastcall TForm1::Timer1Timer(TObject *Sender)
277: {
278:     unsigned char Temp;
279:     unsigned char* PTemp=&Temp;
280:
281:     if (ConnectionTrue==True)
282:     {
283:         if (Button1True==true)
284:         {
285:             if (JvDialButton1->Position<=255)
286:             {
287:                 Temp=0x52;
288:                 Delay();
289:                 ComPort1->Write(PTemp,1);
290:                 Label3->Caption="0x52";
291:                 Temp=(255-JvDialButton1->Position);
292:                 Delay();
293:                 ComPort1->Write(PTemp,1);
294:                 Label4->Caption="0x" + IntToHex(Temp,2);
295:             }
296:             else if (JvDialButton1->Position>255 && JvDialButton1->Position<=345)
297:             {
298:                 Temp=0x00;
299:                 Delay();
300:                 ComPort1->Write(PTemp,1);

```

```

301:         Label3->Caption="0x00";
302:         Label4->Caption="0x";
303:     }
304:     else if (JvDialButton1->Position>345)
305:     {
306:         Temp=0x4c;
307:         Delay();
308:         ComPort1->Write(PTemp,1);
309:         Label3->Caption="0x4c";
310:         Temp = (JvDialButton1->Position-345);
311:         Delay();
312:         ComPort1->Write(PTemp,1);
313:         Label4->Caption="0x" + IntToHex(Temp,2);
314:     }
315: }
316: else if (Button1True==false)
317: {
318:     Temp =0x00;
319:     Delay();
320:     ComPort1->Write(PTemp,1);
321:     Label3->Caption="0x00";
322:     Label4->Caption="0x";
323: }
324: }
325: }
326:
327:
328: //-----
329:
330: // Short description:
331:
332: // This function is entered when the program is terminated.
333: // Before termination it sends a stop character to the HCS12,
334: // if connected.
335:
336: // Pre: Comport is connected or not
337:
338: // Post: If connected: 0x00 stop char is send and Comport1 is closed
339: //        If not connected: Comport1 is opened,
340: //        0x00 stop char is send and Comport1 is closed.
341:
342: //-----
343: void __fastcall TForm1::FormDestroy(TObject *Sender)
344: {
345:     unsigned char Temp;
346:     unsigned char* PTemp=&Temp;
347:
348:     if (ConnectionTrue==True)
349:     {
350:         Temp =0x00;
351:         ComPort1->Write(PTemp,1);
352:         ComPort1->Close();
353:     }
354: }
355:
356: //-----
357: // end of file Unit1.cpp
358:
359:

```

```

1:
2: //*****
3: //*
4: //* -----*
5: //*  Filename: Unit2.cpp
6: //*  Part of:  DCFLOW.exe
7: //*
8: //*  Compiler: Borland C++ Builder 6 servicepack 4
9: //*             CPort VCL , JVCL
10: //*  Made by:  Eric Halmans
11: //*  For:     Fontys Highschool Eindhoven, Mechatronica
12: //*  Date:    April 2006
13: //*  Version: 1.0 beta test version
14: //*
15: //*  Description:
16: //*    This file is part a windows program,
17: //*    which is used for universal AC & DC motor control,
18: //*    with a HCS12 TBoard from Elektronik Laden
19: //*    (Hbridge with PWM on PP0 & PP1, SCI1 input,IIC Output)
20: //* -----*
21: //*
22: //*****
23:
24:
25: #include <vcl.h>
26: #pragma hdrstop
27:
28: #include "Unit2.h"
29:
30: //-----
31: #pragma package(smart_init)
32: #pragma resource "*.dfm"
33: TForm2 *Form2;
34: //-----
35: __fastcall TForm2::TForm2(TComponent* Owner)
36:     : TForm(Owner)
37: {
38: }
39:
40: //-----
41:
42: // Short description:
43:
44: // Button to close about window
45:
46: // Pre:  about window is shown
47:
48: // Post: about window is closed
49:
50: //-----
51: void __fastcall TForm2::Button1Click(TObject *Sender)
52: {
53:     Form2->Close();
54: }
55:
56: //-----
57: // end of file Unit2.cpp
58:
59:

```

```

1:
2: /**
3: /**
4: /** ----- /**
5: /** Filename: Unit3.cpp /**
6: /** Part of: DCFlow.exe /**
7: /** /**
8: /** Compiler: Borland C++ Builder 6 servicepack 4 /**
9: /** CPort VCL , JVCL /**
10: /** Made by: Eric Halmans /**
11: /** For: Fontys Highschool Eindhoven, Mechatronica /**
12: /** Date: April 2006 /**
13: /** Version: 1.0 beta test version /**
14: /** /**
15: /** Description: /**
16: /** This file is part a windows program, /**
17: /** which is used for universal AC & DC motor control, /**
18: /** with a HCS12 TBoard from Elektronik Laden /**
19: /** (Hbridge with PWM on PP0 & PP1, SCI1 input,IIC Output) /**
20: /** ----- /**
21: /** /**
22: /**
23:
24:
25: #include <vcl.h>
26: #pragma hdrstop
27:
28: #include "Unit3.h"
29:
30: //-----
31: #pragma package(smart_init)
32: #pragma link "JvComponent"
33: #pragma link "JvEDIDBBuffering"
34: #pragma resource "*.dfm"
35: TForm3 *Form3;
36: //-----
37: __fastcall TForm3::TForm3(TComponent* Owner)
38:     : TForm(Owner)
39: {
40: }
41:
42: //-----
43:
44: // Short description:
45:
46: // Button to set CPort VCL Custom baudrate
47:
48: // Pre: - CBaudRate->Text contains a value between 0..500.000
49: // - CustomBaudRate
50:
51: // Post: The string value is translated in a integervalue and
52: // is used to set a new CustomBaudRate
53: // if buadrate is not within specs a non valid message is shown
54:
55: //-----
56: void __fastcall TForm3::Button1Click(TObject *Sender)
57: {
58:     if ((StrToInt(Edit1->Text)>=1)&&(StrToInt(Edit1->Text)<=50000))
59:     {
60:         Form1->ComPort1->CustomBaudRate=StrToInt(Edit1->Text);

```



```
61:     Label2->Caption = Form1->ComPort1->CustomBaudRate;
62: }
63: else ShowMessage("    Not valid input for BaudRate    ");
64: }
65:
66: //-----
67: // end of file Unit3.cpp
68:
69:
```