void pro\_J1939\_readdtc()

{

ezxml\_t msgxml;

ezxml\_t ndtcxml;

ezxml\_t fmixml;

U8 buf[512];

int number;

int i;

BOOL DTCSTOP=TRUE;

int dtcnumber=0;

ezxml\_t faultxml;

ezxml\_t faultnode;

ezxml\_t faultp;

ezxml\_t faultFmi;

char faultpath[256];

char strexpress[512];

char fmibuf[128];

char strbuf[256];

char stabuf[128];

char \*seg;

U32 SPN;

U8 dtccount=0;

MsgBox(BUTTON\_CANCLE, trr(String\_Read\_dtc), trr(String\_Processing\_please\_wait), 3);

GetDBPath(faultpath, 256);

strcat(faultpath, ezxml\_attr(m\_threaddata.xml, "FILE"));

seg = (char\*)ezxml\_attr(m\_threaddata.xml, "SEG");

faultxml = ezxml\_parse\_file(faultpath);

if(faultxml == NULL)

{

ezxml\_free(faultxml);

goto\_err();

return;

}

if(ezxml\_child(faultxml, seg))

{

faultnode = ezxml\_child(faultxml, seg);

}

else

{

Debug\_vc("没有故障码文件 %s\r\n",seg);

InsertMenuToDialog(trr(String\_System\_OK),NULL, NULL, NULL);

ezxml\_free(faultxml);

SetThreadID(THREAD\_ID\_KEEP);

SetUnderButton(4, trr(String\_BACK), "\_\_BACKDEC\_\_");

return;

}

if(ezxml\_child(faultnode,"NDTC"))

{

ndtcxml = ezxml\_child(faultnode,"NDTC");

}

if(ezxml\_child(faultnode,"FMI"))

{

fmixml = ezxml\_child(faultnode,"FMI");

}

if(ezxml\_child(faultnode,"DTC"))

{

faultnode = ezxml\_child(faultnode,"DTC");

}

msgxml=ezxml\_child(m\_threaddata.xml,"MSG");

while(msgxml)

{

switch(getHex(ezxml\_attr(msgxml,"TYPE")))

{

case 0://发出70命令

number=pro\_SendRecv\_J1939(msgxml,buf);

break;

case 2://发出其他命令

number=pro\_SendRecv\_J1939(msgxml,buf);

break;

case 1:

number=pro\_SendRecv\_J1939(msgxml,buf);

if(number < 6)

{

msgxml=ezxml\_next(msgxml);

continue;

}

memset(strexpress,0,sizeof(strexpress));

//无故障码的时候第6~3位都为0

if((buf[2]|buf[3]|buf[4]|buf[5])==0)

{

break;

}

//显示故障码

dtcnumber = (number-2)/4;

for(i=0; i<dtcnumber; i++)

{

//SPN

//此处默认设置SPN对所有的19 位均采用英特尔格式

//buf[2+i\*4] SPN 的低8 位有效位

//buf[3+i\*4] SPN 的第2 个字节

//buf[4+i\*4]&0xE0 有效位中的高3 位

SPN = ((buf[4+i\*4]&0xE0)<<11)+(buf[3+i\*4]<<8)+buf[2+i\*4];

if(SPN&0x70000)

{

sprintf(strexpress,"X%05X",SPN);

}

else

{

sprintf(strexpress,"X%04X",SPN);

}

sprintf(strbuf,"X%0d",buf[4+i\*4]&0x1f);

faultp = ezxml\_child(ndtcxml, strexpress);

faultFmi = ezxml\_child(fmixml, strbuf);

if(faultp)

{

strcpy(strexpress,ezxml\_txt(faultp));

if (faultFmi)

{

strcpy(fmibuf,ezxml\_txt(faultFmi));

strcat(strexpress," | ");

strcat(strexpress,fmibuf);

}

}

else

{

//sprintf(strexpress,"X%02X%02X%02X",buf[2+i\*4],buf[3+i\*4],buf[4+i\*4]&0xE0);

sprintf(strexpress,"X%02X%02X%02X",buf[2+i\*4],buf[3+i\*4],buf[4+i\*4]);

if(strcmp(strexpress,"XFFFFFF")==0)

break;

if(strcmp(strexpress,"X000000")==0)

continue;

faultp = ezxml\_child(faultnode, strexpress);

if(faultp)

{

strcpy(strexpress,ezxml\_txt(faultp));

}

else

{

sprintf(strexpress,"X%02X%02X%02X SPN=%d FMI=%d ",buf[2+i\*4],buf[3+i\*4],buf[4+i\*4], SPN, buf[4+i\*4]&0x1f);

strcpy(strexpress,strexpress+1);

strcat(strexpress,trr(String\_Undefined));

}

}

//故障次数0～126,数值127 用于表明未知

if((buf[5+i\*4]&0x7f)!= 0x7f)

{

sprintf(strexpress,"%s %d",strexpress,buf[5+i\*4]&0x7f);

}

if (ezxml\_attr(msgxml,"ACTIVE") && getHex(ezxml\_attr(msgxml,"ACTIVE")) == 0x01)

{

sprintf(strexpress,"%s %s",strexpress,trr(String\_Activation));

}

if (ezxml\_attr(msgxml,"ACTIVE") && getHex(ezxml\_attr(msgxml,"ACTIVE")) == 0x00)

{

sprintf(strexpress,"%s %s",strexpress,trr(String\_Inactive));

}

InsertMenuToDialog(strexpress,NULL, NULL, NULL);

Debug\_vc("J1939\_READ\_DTC->%s\r\n", strexpress);

dtccount++;

}

break;

case 3:

number=pro\_SendRecv\_J1939\_SendSingle(msgxml,buf);

number=pro\_SendRecv\_J1939\_ReadSingle(msgxml,buf);

while(number > 0)

{

if(number < 12 || (buf[10] == 0xff && buf[11] == 0xff))

{

number=pro\_SendRecv\_J1939\_ReadSingle(msgxml,buf);

continue;

}

sprintf(strexpress,"X%02X%02X%02X%02X%02X",buf[6],buf[7],buf[8],buf[10],buf[11]);

faultp = ezxml\_child(faultnode, strexpress);

if (faultp)

{

if (buf[2] == 0xcb)

{

strcpy(stabuf,trr(String\_Activation));

}

else

strcpy(stabuf,trr(String\_Inactive));

sprintf(fmibuf,"%0d",buf[9]);

strcpy(strbuf,trr(String\_Occurrence\_Count));

strcat(strbuf,fmibuf);

strcpy(strexpress,ezxml\_txt(faultp));

strcat(strexpress," | ");

strcat(strexpress,stabuf);

strcat(strexpress," | ");

strcat(strexpress,strbuf);

}

else

{

strcpy(strexpress,strexpress+1);

strcat(strexpress," ");

strcat(strexpress,trr(String\_Undefined));

}

InsertMenuToDialog(strexpress,NULL, NULL, NULL);

number=pro\_SendRecv\_J1939\_ReadSingle(msgxml,buf);

dtccount++;

}

GUI\_X\_Delay(1000);//读取故障码之后等待1秒 否则下次无法读取

break;

default:

break;

}

if(!DTCSTOP)

break;

if(!dtccount)

{

msgxml=ezxml\_next(msgxml);

}

else

{

break;

}

}

removeMsg();

ezxml\_free(faultxml);

if(dtccount == 0)

{

InsertMenuToDialog(trr(String\_System\_OK),NULL, NULL, NULL);

}

SetThreadID(THREAD\_ID\_KEEP);

SetUnderButton(4, trr(String\_BACK), "\_\_BACKDEC\_\_");

return;

}