



FL 1274
NFi 88/35

104883-1

**EXTENSÃO DA INTERFACE DE SAÍDA
COM DISPOSITIVOS GRÁFICOS**

por

Sílvia Delgado Olabarriaga *
João Luiz Dihl Comba **

RP no. 84

DEZEMBRO 1987

| Nota técnica do projeto "Banco de Dados e |
Ferramentas para CAD de Sistemas Digitais"

Orientação : **Profa. Carla M. Dal Sasso-Freitas ****

* trabalho realizado com apoio da FINEP

** trabalho realizado com apoio do CNPQ

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
POS GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO
Av. Osvaldo Aranha, 99
90001 - Porto Alegre - RS - BRASIL
Telefone : (0512) 21-8499
Telex : (051) 2680 - CUCF BR

Correspondência : UFRGS - CPGCC
Caixa Postal 1501
90001 - Porto Alegre - RS - Brasil

CPD/PGCC
BIBLIOTECA
UFRGS

UFRGS
BIBLIOTECA
CPD/PGCC

Comissão Editorial: Taisy Silva Weber
Carla Maria Dal Sasso Freitas

UFRGS

Reitor: Prof. FRANCISCO FERRAZ

Pró-reitor de Pesquisa e Pós-Graduação: Prof. HELGIO TRINDADE

Coordenador do CPGCC: Profa INGRID J. PORTO

Comissão Coordenadora do CPGCC: Prof. Carlos A. Heuser
Prof. Dalcídio M. Claudio
Prof. Flavio Wagner
Profa. Ingrid J. Porto
Prof. Roberto T. Price
Prof. Ricardo Reis

Bibliotecária CPGCC/CPD: Margarida Buchmann

RESUMO

Este relatório apresenta novas funções da interface de saída com dispositivos gráficos do projeto AMPLO.

PALAVRAS-CHAVE : Computação gráfica, interface de saída, funções gráficas

ABSTRACT

This report presents new functions appended to the interface with graphical output devices used in AMPLO.

KEYWORDS : computer graphics, output interface, graphical functions.

SUMARIO

1. INTRODUÇÃO	5
2. NOVAS FUNÇÕES	5
3. MANUAL DE REFERENCIA DAS NOVAS FUNÇÕES	7
4. UTILIZAÇÃO DA NOVA VERSÃO DA IS	20
REFERENCIAS BIBLIOGRAFICAS	21

1. INTRODUÇÃO

A fim de facilitar (e otimizar) a utilização da interface de saída (IS) [OLA 87] pelo pacote gráfico dos editores do AMPLD (PG), foram definidas novas funções no nível dependente de dispositivo. Estas são numeradas a partir de 50 e foram incluídas por dois motivos :

a) necessidade de utilização da IS sem considerar seu estado : como o PG já tem um "estado" em que são definidos seus atributos correntes, é desnecessário (e por vezes inconveniente) que a IS também tenha um. Assim, foram definidas algumas funções que executam os mesmos procedimentos das anteriormente implementadas, porém sem considerar os atributos correntes. Recebem como parâmetro todos os dados de que precisam para realizarem seus procedimentos, formando um conjunto de rotinas que pode ser usado isoladamente das demais;

b) necessidade de realizar operações gráficas mais sofisticadas, como desenho de círculos, arcos e salvamento/restauração de porções da tela para sobreposição de janelas. Tais rotinas foram incluídas na IS a fim de obter máximo desempenho através do aproveitamento de características dependentes de dispositivo.

2. NOVAS FUNÇÕES

As funções introduzidas na IS na presente versão são as seguintes :

a) is50_inic - inicializa o dispositivo gráfico de saída e carrega os conjuntos de texto com matrizes que definem os caracteres de tipo COMUM;

b) is51_linha - traça uma linha entre dois pontos quaisquer da tela;

c) is52_linhah - traça uma linha horizontal;

d) is53_linhav - traça uma linha vertical;

e) is54_retang - desenha um retângulo, dada sua diagonal;

f) is55_ponto - desenha um ponto;

g) is56_area - preenche uma área retangular da tela, dada sua diagonal;

h) is57_texto - exibe um string em determinada posição

da tela;

i) is58_circulo - traça um círculo, dados seu centro e raio;

j) is59_arco - traça um arco de círculo, dados seu centro e octantes das extremidades;

k) is61_tamcarac - informa tamanho dos caracteres contidos em determinado conjunto de texto;

l) is62_settipotx - permite troca do conjunto de caracteres para outro de tipo diferente;

m) is63_setfattx - permite troca do conjunto de caracteres para outro de tamanho diferente;

n) is64_salva_tela - salva o conteúdo da tela em uma área de memória;

o) is65_rest_tela - restaura a tela conforme conteúdo salvo anteriormente.

p) is66_salva_janela - salva o conteúdo de uma janela determinada pelo usuário (através da diagonal) na área de trabalho;

q) is67_rest_janela - restaura uma janela salva anteriormente, liberando ou não a área;

A maior parte das funções incluídas executam os mesmos procedimentos das funções da versão anterior [OLA 87], porém sem utilizar ou modificar os atributos do pacote.

Os códigos dos erros detectados pelas novas funções foram mantidos e são armazenados na variável "iserro".

3. MANUAL DE REFERENCIA DAS NOVAS FUNÇÕES DA IS

is50_inic

RESUMO

```
#include "iserros.h"
```

```
is50_inic ()
```

DESCRIÇÃO

Carrega conjuntos de texto com o arquivo de matrizes para tipo COMUM, fator 1, e passa a tela para o modo gráfico.

Retorna em iserro o código de término :

- ZERO : execução OK.
- ERRDISCO : erro de acesso ao arquivo de matrizes.

is51_linha

RESUMO

```
#include "istipos.h"
```

```
#include "iserros.h"
```

```
is51_linha ( xi, yi, x, y, cor )
```

```
int xi, yi, x, y;
```

```
COR cor;
```

DESCRIÇÃO

Traça uma linha simples e cheia (estilo=CHESIMPL) entre o ponto (xi,yi) e (x,y).

Verifica se os parâmetros são corretos e atribui o código de erro à variável "iserro" :

- ZERO : parâmetros OK.
- COORDINV : coordenadas inválidas (ver is30_coordisp).
- CORINV : cor inválida.

OBSERVAÇÃO :

Ver is21_linha.

is52_linhah

RESUMO

```
#include "istipos.h"  
#include "iserros.h"
```

```
is52_linhah ( y, xi, x, estilo, cor )  
int y, xi, x;  
ESTILO estilo;  
COR cor;
```

DESCRIÇÃO

Traça uma linha horizontal entre (xi,y) e (x,y), utilizando cor e estilo indicados.

Verifica se os parâmetros são corretos e atribui o código de erro à variável "iserro" :

- ZERO : parâmetros OK.
- COORDINV : coordenadas inválidas (ver is30_coordisp).
- ESTINV : estilo de linha inválido.
- CORINV : cor inválida.

OBSERVAÇÃO :

Ver is22_linhah.

is53_linhav

RESUMO

```
#include "istipos.h"  
#include "iserros.h"
```

```
is53_linhav ( x, yi, y, estilo, cor )  
int x, yi, y;  
ESTILO estilo;  
COR cor;
```

DESCRIÇÃO

Traça uma linha vertical entre (x,yi) e (x,y), utilizando cor e estilo indicados.

Verifica se os parâmetros são corretos e atribui o código de erro à variável "iserro" :

- ZERO : parâmetros OK.
- COORDINV : coordenadas inválidas (ver is30_coordisp).
- ESTINV : estilo de linha inválido.
- CORINV : cor inválida.

OBSERVAÇÃO :

Ver is23_linhav.

is54_retang

RESUMO

```
#include "istipos.h"  
#include "iserros.h"
```

```
is54_retang ( xi, yi, x, y, estilo, cor )  
int xi, yi, x, y;  
ESTILO estilo;  
COR cor;
```

DESCRIÇÃO

Desenha um retângulo determinado pela diagonal (xi,yi) a (x,y), na cor e estilo indicados.

Verifica se os parâmetros são corretos e atribui o código de erro à variável "iserro" :

- ZERO : parâmetros OK.
- COORDINV : coordenadas inválidas (ver is30_coordisp).
- ESTINV : estilo de linha inválido.
- CORINV : cor inválida.

OBSERVAÇÃO :

Ver is24_retang.

is55_ponto

RESUMO

```
#include "istipos.h"  
#include "iserros.h"
```

```
is55_ponto ( x, y, cor )  
int x, y;  
COR cor;
```

DESCRIÇÃO

Desenha um ponto em (x,y) na cor indicada.

Verifica se os parâmetros são corretos e atribui o código de erro à variável "iserro" :

- ZERO : parâmetros OK.
- COORDINV : coordenadas inválidas (ver is30_coordisp).
- CORINV : cor inválida.

is56_area

RESUMO

```
#include "iserros.h"
#include "istipos.h"

is56_area ( xi, yi, x, y, cor )
int xi, yi, x, y;
COR cor;
```

DESCRIÇÃO

Preenche um retângulo com a cor indicada. A diagonal do retângulo é determinada pelos pontos (xi,yi) e (x,y).

Verifica se os parâmetros são corretos e atribui o código de erro à variável "iserro" :

- ZERO : parâmetros OK.
- COORDINV : coordenadas inválidas (ver is30_coordisp).
- CORINV : cor inválida.

OBSERVAÇÕES

Ver is26_area.

is57_texto

RESUMO

```
#include "istipos.h"  
#include "iserros.h"
```

```
is57_texto ( x, y, string, cor, conjunto )  
int x, y;  
char *string;  
COR cor;  
int conjunto;
```

DESCRIÇÃO

Exibe o string na posição indicada, usando a cor e o conjunto de texto indicados.

Verifica se os parâmetros são corretos e atribui o código de erro à variável "iserro" :

- ZERO : parâmetros OK.
- COORDINV : coordenadas inválidas (ver is30_coordisp).
- CORINV : cor inválida.
- CONJINV : número de conjunto inválido.

OBSERVAÇÕES

Ver is27_texto.

is58_circulo

RESUMO

```
#include "istipos.h"
#include "iserros.h"

is58_circulo (xc, yc, raio, cor)
int xc, yc, raio ;
COR cor ;
```

DESCRIÇÃO

Traça um círculo com centro em (xc, yc), de raio e na cor recebidos como parâmetro.

Verifica se os parâmetros são corretos e atribui o código de erro à variável "iserro" :

- ZERO : parâmetros OK
- CORINV : cor inválida
- COORDINV : coordenadas inválidas (ver is30_coordisp)
- RAIONV : raio inválido. Os valores que o raio pode assumir são os mesmos da coordenada x (ver is30_coordisp) .

is59_arco

RESUMO

```
#include "istipos.h"  
#include "iserros.h"
```

```
is59_arco (xc, yc, octinic, octfinal, raio, cor)  
int xc, yc, octinic, octfinal, raio ;  
COR cor ;
```

DESCRIÇÃO

Traça um arco de círculo do octante inicial até o octante final . Este círculo e demarcado pelo centro (xc, yc), e raio recebidos como parâmetro. O arco é traçado na cor informada.

Os octantes são numerados a partir do ângulo zero, no sentido anti-horário.

Verifica se os parâmetros são corretos e atribui o código de erro à variável "iserro" :

- ZERO : parâmetros OK
- CORINV : cor inválida
- COORDINV : coordenadas inválidas
(ver is30_coordisp)
- RAIONV : raio inválido. Os valores que o raio
pode assumir são os mesmos da
coordenada x (ver is30_coordisp) .
- OCTINV : octante inicial ou final inválido. Os
octantes variam de 1 a 8 .

is61_tamcarac

RESUMO

```
#include "iserros.h"
```

```
is61_tamcarac ( tx, ty, conjunto )  
int *tx, *ty;  
int conjunto;
```

DESCRIÇÃO

Informa o tamanho dos caracteres do conjunto de texto indicado.

Verifica se os parâmetros são corretos e atribui o código de erro à variável "iserro" :

- ZERO : parâmetros OK.
- CONJINV : número de conjunto inválido.

OBSERVAÇÕES

Ver is32_tamcarac.

is62_settipotx

RESUMO

```
#include "istipos.h"
#include "iserros.h"

is62_settipotx ( tipo, conjunto )
TIPO tipo;
int conjunto;
```

DESCRIÇÃO

Troca as matrizes que definem os caracteres do conjunto de texto indicado, carregando o arquivo correspondente ao tipo especificado.

Retorna em iserro o código de término :

- ZERO : execução OK.
- ERRDISCO : erro de acesso ao arquivo de matrizes.
- TIPOINV : tipo de texto inválido.
- CONJINV : conjunto de texto inválido.

OBSERVAÇÕES

Ver is06_settipotx.

is63_setfatty

RESUMO

```
#include "istipos.h"  
#include "iserros.h"
```

```
is63_setfatty ( fator, conjunto )  
FATOR fator;  
int conjunto;
```

DESCRIÇÃO

Troca as matrizes que definem os caracteres do conjunto de texto indicado, carregando o arquivo correspondente ao tamanho especificado.

Retorna em iserro o código de término :

- ZERO : execução OK.
- ERRDISCO : erro de acesso ao arquivo de matrizes.
- FATINV : fator de texto inválido.
- CONJINV : conjunto de texto inválido.

OBSERVAÇÕES

Ver is07_setfatty

is64_salva_tela

RESUMO

is64_salva_tela ()

DESCRIÇÃO

Copia o conteúdo da memória gráfica em área interna à interface de saída.

Retorna em iserro o código de término :

- ZERO : execução OK.
- ALOCINV : erro de alocação de memória para armazenamento da tela gráfica.

is65_rest_tela

RESUMO

is65_rest_tela ()

DESCRIÇÃO

Copia o conteúdo salvo em área interna à interface de saída na memória gráfica.

Retorna em iserro o código de término :

- ZERO : execução OK.
- ALOCINV : não há tela salva.

is66_salva_janela

RESUMO

```
#include "istipos.h"  
#include "iserros.h"
```

```
is66_salva_janela (ind_jan, xi, yi, xf, yf)  
int ind_jan, xi, yi, xf, yf ;
```

DESCRIÇÃO

Copiar para a memória a janela definida pelas coordenadas indicadas, associando uma nova área alocada ao índice de janela recebido como parâmetro.

Retorna em iserro o código de término :

- ZERO : execução OK.
- ALOCINV : erro de alocação de memória para armazenamento da janela.
- JANINV : índice de janela está inválido (ver número máximo de janelas) ou tentativa para salvar uma janela já salva e não restaurada.

is67_rest_janela

RESUMO

```
#include "istipos.h"  
#include "iserros.h"
```

```
is67_rest_janela (ind_jan, liberar)  
int ind_jan, liberar ;
```

DESCRIÇÃO

Restaurar uma janela previamente salva com a área associada a ind_jan, podendo liberar (1) ou reter (0) esta área.

Retorna em iserro o código de término :

- ZERO : execução OK.
- ALOCINV : erro de alocação de memória para armazenamento da janela.
- JANINV : índice de janela está inválido (ver número máximo de janelas) ou tentativa para restaurar uma janela não salva anteriormente.

4. UTILIZAÇÃO DA NOVA VERSÃO DA IS

A IS (versão 2) apresenta-se de duas formas :

a) completa : a biblioteca contém todas as funções (antigas e novas);

b) reduzida : contém apenas as funções que não consideram os atributos da interface. São elas :

- is28_fim
- is30_coordisp
- is31_aspecto
- novas rotinas

A forma de ligação permanece a mesma, havendo agora também disponível uma biblioteca para TURBO C.

As bibliotecas disponíveis são as seguintes :

- ISCPTC.LIB : versão completa para o TURBOC
- ISCPMS.LIB : versão completa para o MICROSOFT 4.0
- ISSETC.LIB : versão sem estado para o TURBOC
- ISSEMS.LIB : versão sem estado para o MICROSOFT 4.0

Exemplo : para utilização da biblioteca completa, compilador da Microsoft :

```
LINK EXEMPLO,EXEMPLO,,ISCPMS
```

REFERENCIAS BIBLIOGRAFICAS

[OLA 87] OLABARRIAGA S., PINHO M. & COMBA J. Interface de saída com dispositivos gráficos. Porto Alegre, CPGCC da UFRGS, setembro 1987. (RP no. 79)