

Mono display

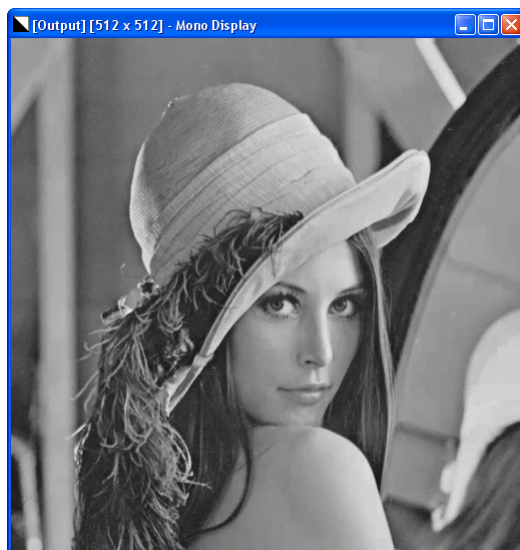
Obsah

1. Popis.....	1
2. Závislosti	2
3. Implementačné informácie	2
4. Piny	2
5. Konfigurácia	2
5.1 Konfiguračná štruktúra.....	2
5.2 Konfiguračný dialóg.....	3
6. Definícia triedy	3
7. Príklad použitia	4

1. Popis



Trieda *Mono display* predstavuje monochromatický display. Vstupom je šedo tónový obraz v štruktúre *TPlane*, ktorý sa naškáluje a oreže na interval $\langle 0, 255 \rangle$.



Okno zobrazuje obraz v odtieňoch šedej. V záhlaví okna sa zobrazuje nastavený názov a rozlíšenie obrazu. Kliknutím pravého tlačidla myši sa zobrazí kontextové menu okna.



Tu sú na výber dva módy okna, *Stretch* a *Fullsize*. V móde *Stretch* sa veľkosť obrazu prispôsobuje veľkosti okna. V móde *Fullsize* sa obraz zobrazuje v pôvodnej veľkosti, v prípade, že je obraz väčší ako veľkosť okna, uchopením a potiahnutím obrazu zmeníme pohľad naň. Voľbou *Close* okno zavrieme.

2. Závislosti

Trieda *Mono display* používa tieto externé definičné súbory a knižnice:

Headers: *filtergraph.h*, *datatypes.h*

Libs: *filtergraph.lib*

3. Implementačné informácie

Informácie o triede:

Názov: *Mono display*

Verzia: *1.0*

Magic: *60*

Informácie o definíciách:

Header: *monodisplay.h*

Konfiguračný header: *monodisplaycfg.h*

Lib: *monodisplay.lib*

Informácie o knižnici obsahujúcej triedu:

Názov knižnice: *Mono display Library*

Verzia knižnice: *1.0*

Dll súbor knižnice: *monodisplay.dll*

4. Piny

Piny sú popísané spôsobom: „*[index pinu na filtri] názov pinu: popis pinu*“.

[0] Cfg in: Konfiguračný vstup. Vstupom je serializovaná konfigurácia.

[1] Cfg out: Konfiguračný výstup. Výstupom je serializovaná konfigurácia.

[2] In: Vstup. Vstupom je šedo tónová rovina v štruktúre *TPlane*.

5. Konfigurácia

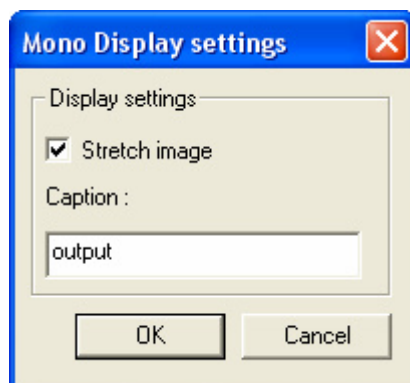
5.1 Konfiguračná štruktúra

Konfiguračná štruktúra je definovaná nasledovne:

```
typedef struct {  
    bool Stretch;           // mód okna, ak Stretch=TRUE, tak mód Stretch, inak mód Fullsize  
    char* Caption;         // názov okna  
} TMonoDisplayConfig;
```

Indikátor *Stretch* určuje mód okna. Ak *Stretch=TRUE*, tak je okno v móde *Stretch*, inak je v móde *Fullsize*. Reťazec *Caption* určuje názov okna, ktorý sa zobrazuje v záhlaví.

5.2 Konfiguračný dialóg



Konfiguračný dialóg poskytuje voľby popísané pri konfiguračnej štruktúre. Môžeme nastaviť mód okna voľbou *Stretch* a názov okna v kolonke *Caption*.

6. Definícia triedy

Trieda *Mono display* je definovaná nasledovne:

```
class TMonoDisplayFilter : public TFilter {
private:
    HINSTANCE      hInstance;           // identifikácia inštalácie knižnice
    HWND           hwnd;                // systémová identifikácia okna
    HBITMAP        hBitmap;            // systémová identifikácia bitmapy
    HMENU           hMenu;              // systémová identifikácia menu

    char* Caption;                      // názov okna

    TBuffer DataBuffer;                 // vstupný dátový buffer

    LONG Width;                          // šírka obrazu
    LONG Height;                        // výška obrazu

    TPin* CfgInPin;                     // vstupný konfiguračný pin
    TPin* CfgOutPin;                    // výstupný konfiguračný pin
    TPin* InPin;                        // vstupný pin

    bool StopFlag;                     // indikátor zastavenia

    int setConfigFromPin(TPin *pin);    // konfigurácia na pinoch...
    int putConfigOnPin(TPin *pin);
    int freeConfigOnPin(TPin *pin);
    int clearOutputPins();              // vyčistenie výstupov...
    int freeOutputData();

public:
    int ScreenMode;                    // mód okna
```

```

int ScreenState;                // stav okna
POINT APnt;                     // pomocné body...
POINT BPnt;

TMonoDisplayFilter();           // konštruktor
TMonoDisplayFilter(HINSTANCE hInst); // konštruktor s parametrami
~TMonoDisplayFilter();          // deštruktor

int setConfigData(TBuffer config,int type); // predefinované funkcie triedy TFilter...
TBuffer getConfigData(int type);
int initialize();
int run();
int reset();
int stop();
int finalize();
int showConfigDialog();

HBITMAP getBitmap();           // pomocné funkcie...
HMENU getMenu();
SIZE getSize();
};

```

7. Príklad použitia

