

第2版

画像認識ユニット

NVP-Ax230SDK
SVP-Ax330SDK

Software Development Kit

Fine Vision Processor

ProGen2 操作マニュアル

maxell

マクセルシステムテック株式会社

はじめに

このたびは、VP-Axシリーズをお買い上げいただきまして、誠にありがとうございます。

本マニュアルはVP-Axシリーズ用ツール ProGen2 について記載しております。



ご注意

- システムの構築やプログラム作成などの操作を行う前に、本マニュアルの記載内容をよく読み、書かれている 指示や注意を十分理解してください。誤った操作によりシステムの故障が発生することがあります。
- 本マニュアルの記載内容について理解できない内容、疑問点または不明点がございましたら、弊社営業窓口までお知らせください。また、弊社ホームページのお問い合わせのページからも受け付けていますのでご利用ください。
<http://www.systemtech.maxell.co.jp/solution/vp/>
- お客様の誤った操作に起因する、事故発生や損害につきましては、弊社は責任を負いかねますのでご了承ください。
- 弊社提供のハードウェアおよびソフトウェアを無断で改造しないでください。この場合の品質および安全につきましては、弊社は責任を負いかねますのでご了承ください。
- 本マニュアルの内容について予告なく変更する場合があります。

※ Windowsは、米国Microsoft Corporationの米国およびその他の国における登録商標です。

※ その他記載の社名および商品名は、一般に各社の商標または登録商標です。

ご注意書き

1. 本ソフトウェアおよび本資料に記載されるこれらに関連する情報を使用する場合には、お客様の責任において行ってください。これらの使用に起因して、お客様または第三者に生じた損害に関し、当社は、一切その責任を負いません。
2. 本ソフトウェアおよび本資料に記載される情報は、正確を期するため慎重に作成したのですが、誤りがないことを保証するものではありません。万一、本ソフトウェアおよび本資料に記載される情報の誤りに起因する損害が生じた場合においても、当社は一切その責任を負いません。
3. 本資料に記載された製品データ、図、表、プログラム、アルゴリズム、応用例等の情報の使用に起因して発生した第三者の特許権、著作権その他知的財産権に対する侵害に関し、当社は、何らの責任を負うものではありません。当社は本資料に基づき当社または第三者の特許権、著作権その他知的財産権を何ら許諾するものではありません。
4. 本ハードウェアおよび本ソフトウェアを改造、改変、複製等しないでください。かかる改造、改変、複製により生じた損害に関し、当社は一切その責任を負いません。
5. 本ハードウェアおよび本ソフトウェアは、直接生命・身体に危害を及ぼす可能性のある機器・システム（生命維持装置、人体に埋め込み使用するもの等）、もしくは多大な物質的損害を発生させるおそれのある機器・システム（原子力制御システム・軍事機器等）、もしくは高品質水準が求められる機器・システム（自動車・電車・船舶等の輸送機器、交通用信号機器、防災・防犯装置、各種安全装置等）に使用されることを意図しておらず、使用することはできません。たとえ、意図しない用途に当社製品を使用したことによりお客様または第三者に損害が生じても、当社は一切その責任を負いません。なお、ご不明点がある場合は、当社営業にお問い合わせください。
6. 本ハードウェアおよび本ソフトウェアをご使用の際は、当社が指定する最大定格、動作電源電圧範囲、放熱特性、その他の保証範囲内でご使用ください。当社保証範囲を超えて当社製品をご使用された場合の故障および事故につきましては、当社は、一切その責任を負いません。
7. 当社は、本ハードウェアおよび本ソフトウェアの品質向上に努めておりますが、使用条件によっては誤動作したりする場合があります。本ハードウェアおよび本ソフトウェアのご使用にあたっては、お客様の責任において、冗長設計、延焼対策設計、誤動作防止設計等の安全設計およびエージング処理等、お客様の機器・システムとしての出荷保証、および安全検証を行ってください。
8. 本資料に記載されている当社のハードウェア製品、ソフトウェア製品および技術を国内外の法令および規制により、製造・使用・販売を禁止されている機器・システムに使用することはできません。また、本ハードウェア、本ソフトウェアおよび技術を大量破壊兵器の開発等の目的、軍事利用の目的その他軍事用途に使用しないでください。本ハードウェア、本ソフトウェアおよび技術を輸出する場合は「外国為替及び外国貿易法」その他輸出関連法令を遵守し、かかる法令の定めるところにより必要な手続きを行ってください。
9. お客様の転売等により、本ご注意書き記載の諸条件に抵触して当社製品が使用され、その使用から損害が生じた場合、当社は何らの責任も負わず、お客様にてご負担頂きますのでご了承ください。
10. 本資料の全部または一部を当社の文書による事前の承諾を得ることなく転載または複製することを禁じます。

目次

1.	概要.....	1
1.1	機能.....	1
2.	操作方法.....	2
2.1	SDK の選択.....	2
2.2	メインダイアログ.....	2
2.3	プロジェクト構成設定とディレクトリ構成.....	4
2.4	[ライブラリ]タブ.....	5
2.5	[ソースファイル]タブ.....	6
2.5.1	[映像表示設定]ダイアログ.....	7
2.5.2	[映像入力設定]ダイアログ.....	8
2.5.3	[画面確保]ダイアログ.....	9
2.6	[詳細設定]ダイアログ.....	10
2.7	プロジェクト作成.....	11
3.	注意事項.....	14
付録 A	変更履歴.....	15

図・表 目次

図 2-1 [ProGen2] ダイアログ	2
図 2-2 [ライブラリ] タブ	5
図 2-3 [ソースファイル] タブ	6
図 2-4 [映像表示設定] ダイアログ	7
図 2-5 [映像入力設定] ダイアログ (NVP-Ax230)	8
図 2-6 [画面確保] ダイアログ	9
図 2-7 [詳細設定] ダイアログ	10
表 1-1 Microsoft Visual Studio 2005 プロジェクトファイル	1
表 1-2 SuperH RISC engine C/C++ Ver9.0 プロジェクトファイル	1
表 2-1 [ProGen2] ダイアログ 設定項目	3
表 2-2 プロジェクト構成の設定とディレクトリ構成	4
表 2-3 [ライブラリ] タブ 設定	5
表 2-4 [ソースファイル] タブ 設定	6
表 2-5 [映像表示設定] ダイアログ 設定	7
表 2-6 [映像入力設定] ダイアログ 設定	8
表 2-7 [画面確保] ダイアログ 設定	9
表 2-8 [詳細設定] ダイアログ 設定	10
リスト 2-1 ソースファイル出力例①	11
リスト 2-2 ソースファイル出力例②	12
リスト 2-3 ソースファイル出力例③	13

1. 概要

1.1 機能

ProGen2 は VP-Ax シリーズ SDK を利用したアプリケーション開発用のプロジェクトを作成するツールです。作成するプロジェクトは Microsoft Visual Studio 2005、および、SuperH RISC engine C/C++ Ver 9.0 (HEW) の 2 種類です（表 1-1、表 1-2）。

表1-1 Microsoft Visual Studio 2005 プロジェクトファイル

ファイル	種類
[プロジェクト名].sln	Microsoft Visual Studio Solution
[プロジェクト名].vcproj	VC++ Project

表1-2 SuperH RISC engine C/C++ Ver9.0 プロジェクトファイル

ファイル	種類
[プロジェクト名].hwp	HEW hwp ファイル
[プロジェクト名].hws	HEW Project

2. 操作方法

2.1 SDK の選択

ProGen2 にて作成するプロジェクトは、ご使用の SDK により生成されるプロジェクトの内容が異なります。複数の SDK がインストールされる PC では VPSetReg2 により製品選択を行ってください。VPSetReg2 の詳細は環境設定マニュアルを参照してください。

なお、本マニュアルでは製品選択で NVP-Ax230SDK を選択していることを前提に説明します。

2.2 メインダイアログ

ProGen2 を起動すると [ProGen2] ダイアログが表示されます (図 2-1)。各設定項目の詳細は表 2-1 を参照してください。

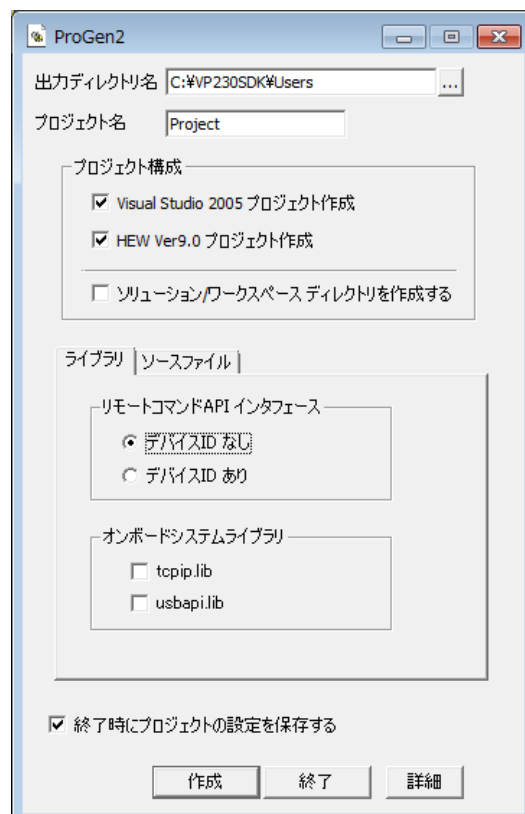


図2-1 [ProGen2]ダイアログ

表2-1 [ProGen2]ダイアログ設定項目

項目	説明
出力ディレクトリ名	作成するプロジェクトの出力ディレクトリを指定します。右の[...]ボタンからディレクトリ参照できます。
プロジェクト名	プロジェクト名を指定します。
プロジェクト構成	作成するプロジェクトを選択します(詳細は2.3章参照)。
[ライブラリ]タブ	作成するプロジェクトのライブラリ設定を指定します(詳細は2.4章参照)。
[ソースファイル]タブ	作成するプロジェクトのソースファイル設定を指定します(詳細は2.5章参照)。
[終了時にプロジェクトの設定を保存する]チェックボックス	ProGen2 終了時に現在の設定を保存するかを選択します。
[作成]ボタン	プロジェクトを作成します。
[終了]ボタン	ProGen2 を終了します。
[詳細]ボタン	プロジェクトの詳細設定を行います(詳細は2.6章参照)。

2.3 プロジェクト構成設定とディレクトリ構成

ProGen2 で作成するディレクトリ構成を表 2-2に示します。

表2-2 プロジェクト構成の設定とディレクトリ構成

プロジェクト構成の設定	出力結果
<input checked="" type="checkbox"/> Visual Studio2005 プロジェクト作成 <input checked="" type="checkbox"/> Hew Ver9.0 プロジェクト作成 <input type="checkbox"/> ソリューション/ワークスペースディレクトリを作成する	<出力ディレクトリ> └─ <プロジェクト名> ├─ <SH> └─ <プロジェクト名> hws, hwp ファイル ├─ <VC> └─ <プロジェクト名> sln, vcproj ファイル └─ <src> main.c
<input checked="" type="checkbox"/> Visual Studio2005 プロジェクト作成 <input checked="" type="checkbox"/> Hew Ver9.0 プロジェクト作成 <input checked="" type="checkbox"/> ソリューション/ワークスペースディレクトリを作成する	<出力ディレクトリ> └─ <プロジェクト名> ├─ <SH> ├─ <プロジェクト名> hws ファイル └─ <プロジェクト名> hwp ファイル ├─ <VC> ├─ <プロジェクト名> sln ファイル └─ <プロジェクト名> vcproj ファイル └─ <src> main.c
<input checked="" type="checkbox"/> Visual Studio2005 プロジェクト作成 <input type="checkbox"/> Hew Ver9.0 プロジェクト作成 <input type="checkbox"/> ソリューション/ワークスペースディレクトリを作成する	<出力ディレクトリ> └─ <プロジェクト名> sln, vcproj ファイル └─ <src> main.c
<input checked="" type="checkbox"/> Visual Studio2005 プロジェクト作成 <input type="checkbox"/> Hew Ver9.0 プロジェクト作成 <input checked="" type="checkbox"/> ソリューション/ワークスペースディレクトリを作成する	<出力ディレクトリ> └─ <プロジェクト名> sln ファイル ├─ <プロジェクト名> vcproj ファイル └─ <src> main.c
<input type="checkbox"/> Visual Studio2005 プロジェクト作成 <input checked="" type="checkbox"/> Hew Ver9.0 プロジェクト作成 <input type="checkbox"/> ソリューション/ワークスペースディレクトリを作成する	<出力ディレクトリ> └─ <プロジェクト名> hws, hwp ファイル └─ <src> main.c
<input type="checkbox"/> Visual Studio2005 プロジェクト作成 <input checked="" type="checkbox"/> Hew Ver9.0 プロジェクト作成 <input checked="" type="checkbox"/> ソリューション/ワークスペースディレクトリを作成する	<出力ディレクトリ> └─ <プロジェクト名> hws ファイル ├─ <プロジェクト名> hwp ファイル └─ <src> main.c

2.4 [ライブラリ]タブ

[ライブラリ]タブ(図 2-2)の設定を表 2-3に示します。

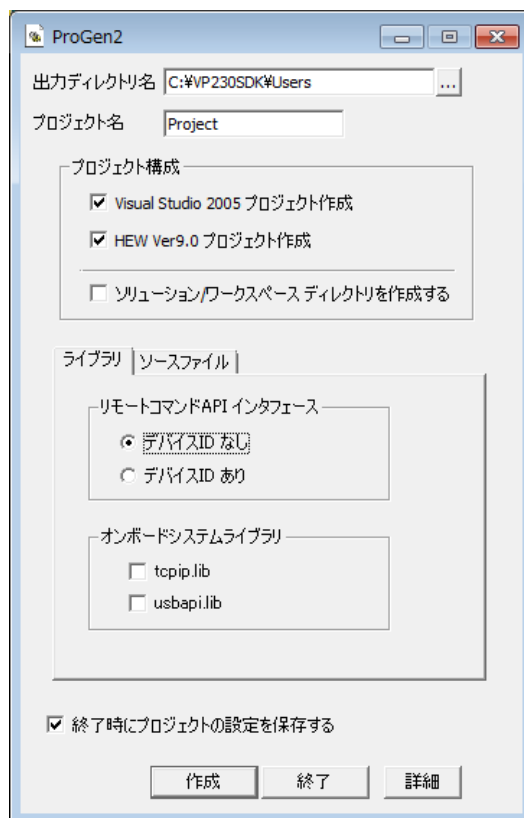


図2-2 [ライブラリ]タブ

表2-3 [ライブラリ]タブ設定

項目	説明
リモートコマンド [®] API インタフェース	デバイス ID の有無を選択します。 ・デバイス ID なし … ax200sg1.lib をリンクします。 ・デバイス ID あり … ax200mul.lib をリンクします。マクロ定義に “MULTIBOARD_CONFIG”を追加します。
オンボード [®] システム ライブラリ	TCP/IP コマント [®] 、USB、ATA アクセスコマント [®] を使用する場合、有効にします。 ・tcpip.lib … TCP/IP コマント [®] ・usbapi.lib … USB、ATA アクセスコマント [®] (*1)

(*1) SVP-Ax330SDK は未サポートです。

2.5 [ソースファイル]タブ

[ソースファイル]タブ(図 2-3)の設定を表 2-4に示します。

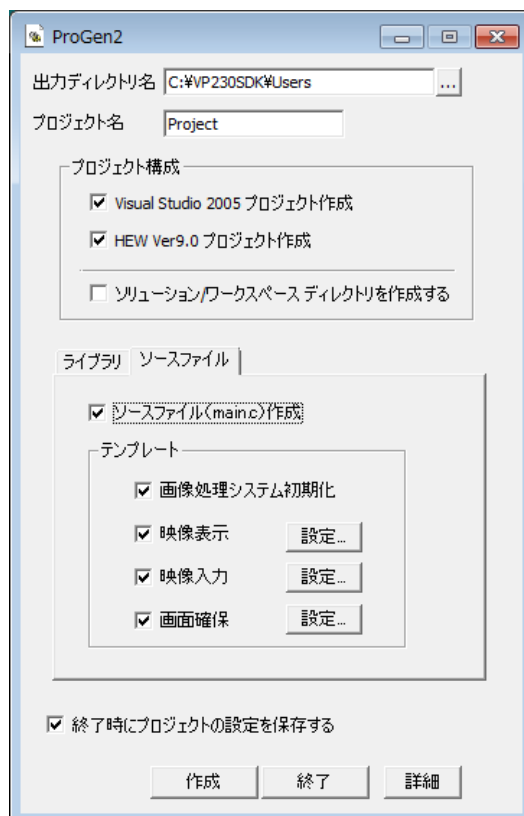


図2-3 [ソースファイル]タブ

表2-4 [ソースファイル]タブ設定

項目	説明
ソースファイル(main. c) 作成	ソースファイル(main. c)を作成するかを選択します。無効にするとソースファイルは作成されません。
画像処理システム 初期化	有効にすると画像処理システムの初期化コメントが記述されます。無効にすると空の main 関数が作成されます。
映像表示	有効にすると映像表示系の初期化コメントが記述されます。[設定]ボタンをクリックすると[映像表示設定]ダイアログが表示されます(詳細は2. 5. 1章参照)。
映像入力	有効にすると映像入力系の初期化コメントが記述されます。[設定]ボタンをクリックすると[映像入力設定]ダイアログが表示されます(詳細は2. 5. 2章参照)。
画面確保	有効にすると画面確保系の初期化コメントが記述されます。[設定]ボタンをクリックすると[画面確保]ダイアログが表示されます(詳細は2. 5. 3章参照)。

2.5.1 [映像表示設定]ダイアログ

[映像表示設定]ダイアログ(図 2-4)の設定を表 2-5に示します。

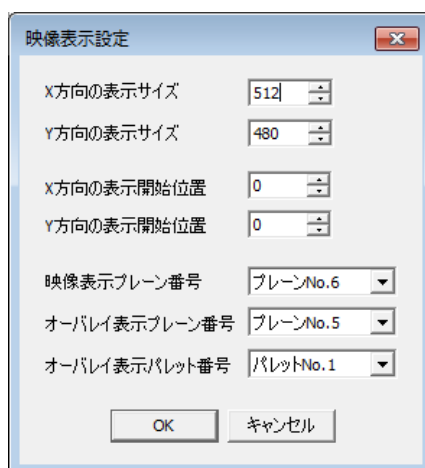


図2-4 [映像表示設定]ダイアログ

表2-5 [映像表示設定]ダイアログ設定

項目	説明
X方向の表示サイズ	SetConfigDisp パラメータ(ConfigDispPara xsize)として記述されます。
Y方向の表示サイズ	SetConfigDisp パラメータ(ConfigDispPara ysize)として記述されます。
X方向の表示開始位置	SetConfigDisp パラメータ(ConfigDispPara dpx)として記述されます。
Y方向の表示開始位置	SetConfigDisp パラメータ(ConfigDispPara dpy)として記述されます。
映像表示プレーン番号	SetConfigDisp パラメータ(ConfigDispPara disp_plane)として記述されます。 ^(*1)
オーバーレイ表示プレーン番号	SetConfigDisp パラメータ(ConfigDispPara overlay_plane)として記述されます。 ^{(*1) (*2)}
オーバーレイ表示パレット番号	SetConfigDisp パラメータ(ConfigDispPara overlay_palette)として記述されます。 ^(*2)

^(*1) オーバーレイ表示プレーン番号 < 映像表示プレーン番号 となる様設定してください。

^(*2) オーバーレイ表示プレーン番号、オーバーレイ表示パレット番号が共にオーバーレイ表示なし以外のとき、SetDispPalette が記述されます。

2.5.2 [映像入力設定]ダイアログ

[映像入力設定]ダイアログ(図 2-5)の設定を表 2-6に示します。

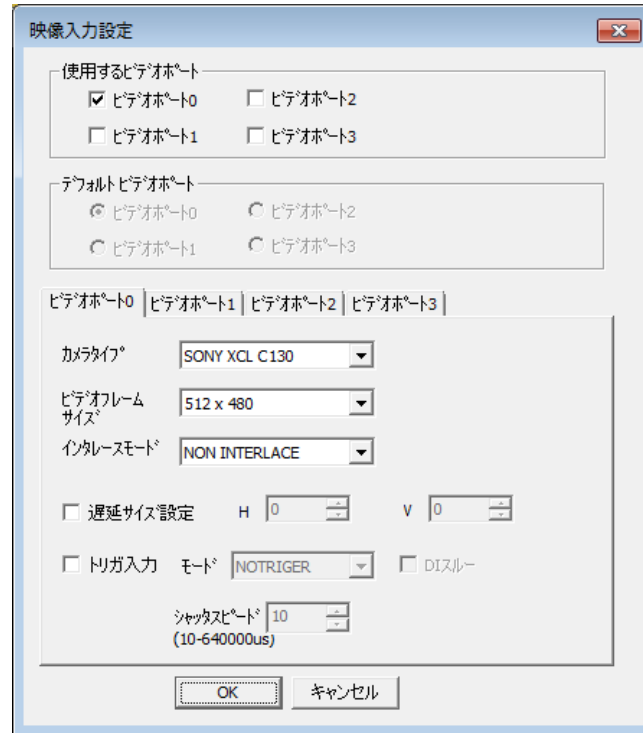


図2-5 [映像入力設定]ダイアログ (NVP-Ax230)

表2-6 [映像入力設定]ダイアログ設定

項目	説明
使用するビデオポート	ActiveVideoPort のパラメータとして記述されます。
デフォルトビデオポート	初期化後のアクティブビデオポートを指定します(指定したビデオポートが最後に記述されます)。
カメラタイプ	SelectCamera のパラメータとして記述されます。
ビデオフレームサイズ	SetVideoFrame パラメータとして記述されます。
インターレースモード	SetVideoFrame パラメータとして記述されます。
遅延サイズ設定	SetVFDelay のパラメータとして記述されます。
トリガ入力	SetTriggerMode のパラメータとして記述されます。

※ 製品選択 (NVP-Ax230 or SVP-Ax330)によりダイアログや設定可能なパラメータが異なります。

2.5.3 [画面確保]ダイアログ

[画面確保]ダイアログ(図 2-6)の設定を表 2-7に示します。



図2-6 [画面確保]ダイアログ

表2-7 [画面確保]ダイアログ設定

項目	説明
画面確保数	画面確保数分だけ確保する様に記述されます。
画面サイズ	AllocImg または AllocRGBImg のパラメータとして記述されます。
画面属性	BW 指定時は AllocImg (Y (モノクロ) 画面確保) が記述されます。 RGB 指定時は AllocRGBImg (RGB 画面確保) が記述されます。

2.6 [詳細設定]ダイアログ

[詳細設定]ダイアログ(図 2-7)の設定を表 2-8に示します。[追加]、[編集]、[削除]ボタンにより各項目について変更することができます。

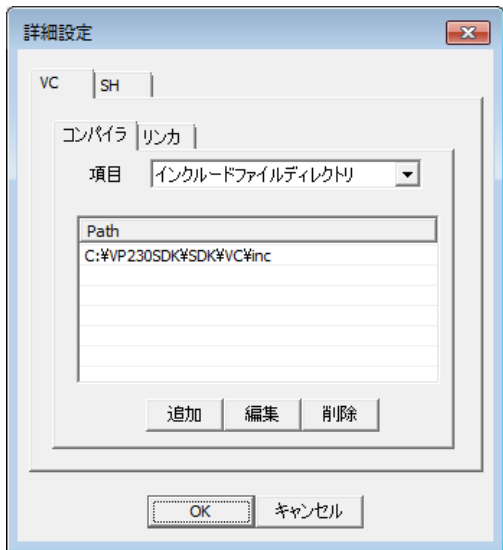


図2-7 [詳細設定]ダイアログ

表2-8 [詳細設定]ダイアログ設定

項目			説明
[VC] タブ	[コンパイラ] タブ	インクルードファイルディレクトリ	Visual Studio2005プロジェクト インクルードファイルを設定します。
		マクロ定義	Visual Studio2005プロジェクト マクロ定義を設定します。
	[リンカ] タブ	ライブラリファイル	Visual Studio2005プロジェクト ライブラリファイルを設定します。
[SH] タブ	[コンパイラ] タブ	インクルードファイルディレクトリ	Hew Ver9.0プロジェクト インクルードファイルを設定します。
		マクロ定義	Hew Ver9.0プロジェクト マクロ定義を設定します。
	[リンカ] タブ	ライブラリファイル	Hew Ver9.0プロジェクト ライブラリファイルを設定します。
		セクション	Hew Ver9.0プロジェクト セクションを設定します。

2.7 プロジェクト作成

メインダイアログで[作成]ボタンをクリックすると、プロジェクトを作成します。リスト 2-1、リスト 2-2、リスト 2-3にソースファイル出力例を示します。

リスト2-1 ソースファイル出力例①

```
#include <stdio.h>
#include "ipxdef.h"
#include "ipxsys.h"
#include "ipxprot.h"

#define RGB_BLACK      (16)
#define RGB_BLUE      (1)
#define RGB_GREEN      (2)
#define RGB_CYAN      (3)
#define RGB_RED        (4)
#define RGB_MAGENTA    (5)
#define RGB_YELLOW     (6)
#define RGB_WHITE      (7)
#define RGB_D_BLUE     (8)
#define RGB_D_GREEN    (9)
#define RGB_D_CYAN     (10)
#define RGB_D_RED      (11)
#define RGB_D_MAGENTA  (12)
#define RGB_D_YELLOW   (13)
#define RGB_GRAY       (14)
#define RGB_B_GRAY     (15)

const DURGB_Color lut01[17] = {
    { 0, 0, 0, 0 }, /* TRANSPARENT_INDEX */
    { 0, 0, 0, 255 }, /* RGB_BLUE */
    { 0, 0, 255, 0 }, /* RGB_GREEN */
    { 0, 0, 255, 255 }, /* RGB_CYAN */
    { 0, 255, 0, 0 }, /* RGB_RED */
    { 0, 255, 0, 255 }, /* RGB_MAGENTA */
    { 0, 255, 255, 0 }, /* RGB_YELLOW */
    { 0, 255, 255, 255 }, /* RGB_WHITE */
    { 0, 0, 0, 128 }, /* RGB_D_BLUE */
    { 0, 0, 128, 0 }, /* RGB_D_GREEN */
    { 0, 0, 128, 128 }, /* RGB_D_CYAN */
    { 0, 128, 0, 0 }, /* RGB_D_RED */
    { 0, 128, 0, 128 }, /* RGB_D_MAGENTA */
    { 0, 128, 128, 0 }, /* RGB_D_YELLOW */
    { 0, 128, 128, 128 }, /* RGB_GRAY */
    { 0, 192, 192, 192 }, /* RGB_B_GRAY */
    { 0, 0, 0, 0 } /* RGB_BLACK */
};

int gImgID[4]; /* 画面番号 */
```


リスト2-2 ソースファイル出力例②

```

/*****
[ 関数名 ] main
[ 書 式 ] void main( void );
[ 引 数 ]
[ 戻り値 ]
*****/
void main( void )
{
    int ret;
    int boardNo = 0;
    int opt = 0;
    ConfigDispPara cfgdisp;
    DispPaletteTbl PalTbl;
    int i;

    /*-----
    画像処理システムの初期化
    -----*/

    ret = StartIP( boardNo, opt );
    if( ret ){
        printf( "StartIP ERROR : %08X¥n", ret );
        return;
    }

    ret = InitIP();
    if( ret ){
        printf( "InitIP ERROR : %08X¥n", ret );
        return;
    }

    /*-----
    映像表示設定
    -----*/

    memset( &cfgdisp, 0, sizeof( ConfigDispPara ) );
    cfgdisp.xsize      = 512;          /* X方向の表示サイズ      */
    cfgdisp.ysize      = 480;          /* Y方向の表示サイズ      */
    cfgdisp.dpx        = 0;            /* X方向の表示開始位置    */
    cfgdisp.dpy        = 0;            /* Y方向の表示開始位置    */
    cfgdisp.disp_plane  = DISP_PLANE_6; /* 映像表示プレーン番号    */
    cfgdisp.overlay_plane = DISP_PLANE_5; /* オーバレイ表示プレーン番号 */
    cfgdisp.overlay_pallet = DISP_PALLET_1; /* オーバレイ表示パレット番号 */

    ret = SetConfigDisp( &cfgdisp );
    if( ret ){
        printf( "SetConfigDisp ERROR¥n" );
        return;
    }

    memset( &PalTbl, 0, sizeof( PalTbl ) );
    for(i=0; i< 17; i++){
        PalTbl.Color[i].Alpha = lut01[i].Alpha;
        PalTbl.Color[i].Red   = lut01[i].Red;
        PalTbl.Color[i].Green = lut01[i].Green;
        PalTbl.Color[i].Blue  = lut01[i].Blue;
    }
}

```

リスト2-3 ソースファイル出力例③

```

ret = SetDispPalette( DISP_PALETTE_1, &PalTbl );
if( ret ){
    printf( "SetDispPalette ERROR¥n" );
    return;
}

/*-----
  画像メモリ領域確保
-----*/
for( i=0; i<4; i++ ){
    gImgID[i] = AllocImg( IMG_FS_512H_512V );
    if( gImgID[i] < 0 ){
        printf( "AllocImg ERROR¥n" );
        return;
    }
}

/*-----
  映像入力設定
-----*/
ret = ActiveVideoPort( VIDEO_PORT0 );
if( ret ){
    printf( "ActiveVideoPort ERROR [VIDEO_PORT0]¥n" );
    return;
}

ret = SelectCamera( CAMERA_PORT0, SONY_XCL_C130 );
if( ret ){
    printf( "SelectCamera ERROR [VIDEO_PORT0]¥n" );
    return;
}

ret = SetVideoFrame( NONINTERLACE, VIDEO_FS_512H_480V );
if( ret ){
    printf( "SetVideoFrame ERROR [VIDEO_PORT0]¥n" );
    return;
}

/*-----
  リモートコマンドの使用停止
-----*/
StopIP( boardNo );
}

```

3. 注意事項

- (1) 本ツールは画像認識アプリケーション作成時の雛形プロジェクト生成を目的とするものです。本ツールにより生成したソースコード、および、プロジェクトは動作を保証するものではありません。コンパイルエラー、実行時エラー発生時は、SDKのユーザズマニュアル、コマンドリファレンスを確認の上、ソースファイル、および、プロジェクトの設定を修正してください。
- (2) 本マニュアルに記載した開発環境(Microsoft Visual Studio 2005, SuperH RISC engine C/C++ Ver9.0)とバージョンが一致しない場合、ご利用の開発環境ツールでプロジェクトをオープンしてください。互換性があれば自動的にプロジェクトが変換されます。正しくプロジェクトをオープンできない場合、前提条件となる開発環境を再度ご確認ください。
- (3) 本ツールにて生成した SuperH RISC engine C/C++ Ver9.0 用プロジェクトではセッションファイルを自動生成しません。そのため、プロジェクトをオープンするとセッションファイルがない旨のメッセージが出力される場合がありますが、そのままプロジェクトを開くことで問題なく利用できます。また、プロジェクトの終了時にセッションファイルを自動生成する旨のメッセージが出力される場合がありますが、そのまま保存、作成することで次回オープンからは本メッセージが表示されなくなります。

付録A 変更履歴

【第1版】
初版

【第2版】
社名を変更しました。

画像認識ユニット NVP-A x 2 3 0 SDK
SVP-A x 3 3 0 SDK
ProGen2 操作マニュアル (第2版)

(C) マクセルシステムテック株式会社

開発元

マクセルシステムテック株式会社

設計部 〒992-0021 山形県米沢市花沢 3091-6

営業部 〒244-0801 神奈川県横浜市戸塚区信濃町 549-2 三宅ビル

技術サポート窓口 URL <http://www.systemtech.maxell.co.jp/>
mail : vp-support@maxell.co.jp