

PythonベースのPolyphonyを使って 高位合成でコードを分かりやすくする

鈴木 量三朗

カメラ・モジュールの設定は比較的可変することが多い箇所です。

例えば、画像の出力フォーマットをベイヤ・パターンからRGB565へ変更することもあるでしょう。

サンプル・プロジェクトではVerilog HDLでリスト1のように設定が書かれており、sregという変数に16進数の値で代入されています。

バイナリで書かれた設定値は 分かりづらい

例えばaddressの178では、16'hDA_04という値が代入され、その意味がコメントに書かれています。

しかし、その数字の前半の8ビットであるhDAが、イメージセンサOV2640(オムニビジョン)のレジスタ番号であり、そこにh04を代入する処理であることを読み取ることは難しいでしょう。

さらにh04の意味がRAW10というOV2640の出力フォーマットの指定であり、フォーマットをRGB565に変更するためにはh08に変えればよいということは、コメントがなければ全く分かりません。

これらの数字の列を確信を持って変更するには、仕様書を片手にその前後の設定項目や他の関連するレジ

スタのチェックなどの作業を強いられることとなります。

仕様書に書かれている意味がコードからは消失してしまっているため、後からパラメータを変更する作業が困難になります。

コードから意味を読み取れるようにするにはコメントを付け加えるだけではなく、関数にするなどの工夫が必要です。

カメラの設定処理を Pythonで書く

● Pythonを使ってコードの可読性をよくする

Pythonの高位合成であるPolyphonyを使用すれば、Verilog HDLのコードと同じインターフェースを持ったモジュールを作成できます。最初はPythonのリストとして、数字の列による表現を使って100%互換性のある安全なモジュールを作り、次のステップで関数化するなど可読性の高いコードに変換していくことができます。

関数に適切な名称を付けることで、コードを理解しやすくし、仕様変更などのメンテナンスにも安全に対応できるモジュールを作ることもできるでしょう。

Polyphonyの環境構築方法などは、Appendix1を参

リスト1 OV2640の設定をするVerilog HDLコード(OV2640_Registers.v)

```

case (address)
000 : sreg <= 16'hFF_01;
001 : sreg <= 16'h12_80;

中略
address 178
177 : sreg <= 16'hE0_04;
178 : sreg <= 16'hDA_04; //08:RGB565 04:RAW10
179 : sreg <= 16'hD7_03;
180 : sreg <= 16'hE1_77;
181 : sreg <= 16'hE0_00;
182 : sreg <= 16'hFF_00;
183 : sreg <= 16'h05_01;
184 : sreg <= 16'h5A_A0; //(w>>2)&0xFF //28:w=160
//A0:w=640 //C8:w=800
185 : sreg <= 16'h5B_78; //(h>>2)&0xFF
//1E:h=120 //78:h=480 //96:h=600
186 : sreg <= 16'h5C_00;
//(((h>>8)&0x04)|((w>>10)&0x03))
187 : sreg <= 16'hFF_01;
188 : sreg <= 16'h11_80; //clkrcc=0x83 for
resolution <= SVGA
189 : sreg <= 16'hFF_01;
190 : sreg <= 16'h12_40; /* DSP input image
resolution and window size control */
191 : sreg <= 16'h03_0A;

/* UXGA=0x0F, SVGA=0x0A,CIF=0x06 */
192 : sreg <= 16'h32_09;
/* UXGA=0x36, SVGA/CIF=0x09 */
193 : sreg <= 16'h17_11;
/* UXGA=0x11, SVGA/CIF=0x11 */
194 : sreg <= 16'h18_43;
/* UXGA=0x75, SVGA/CIF=0x43 */
195 : sreg <= 16'h19_00;
/* UXGA=0x01, SVGA/CIF=0x00 */
196 : sreg <= 16'h1A_4b;
/* UXGA=0x97, SVGA/CIF=0x4b */
197 : sreg <= 16'h3d_38;
/* UXGA=0x34, SVGA/CIF=0x38 */
198 : sreg <= 16'h35_da;

中略
225 : sreg <= 16'hFF_00;
226 : sreg <= 16'hE0_04;
227 : sreg <= 16'hDA_04; //08:RGB565 04:RAW10
228 : sreg <= 16'hD7_03;
229 : sreg <= 16'hE1_77;
230 : sreg <= 16'hE0_00;
default : sreg <= 16'hFF_FF; // End configuration
endcase

```