モデル検査に対応する上位ハードウェア記述言語 Melasy の VHDL コード生成

野村 達雄† 岩崎 直木†† 和崎 克己††

信州大学工学部情報工学科† 信州大学大学院工学系研究科†† 信州大学大学院工学系研究科††

1. まえがき

IT 技術の発展により，様々な環境で電子機器が動作しており，年々規模・数ともに増加傾向にある．また，様々な社会システムがこれらの技術と信頼性に依存しており，とりわけディジタルシステムにおける技術と信頼性は社会システムの維持に必要不可欠である．仕様通りに確実に動作する信頼性の高いシステムを，より安価に効率よく設計できる環境が望まれている．ハードウェア設計の正当性を形式的にチェックするツールとしては，SMV，NuSMV 等のモデル検査ツール(Model Checker)[1]が存在する．これらのツールを用いることで設計の正当性の評価を自動で行うことができる．

ハードウェア開発において，設計の正当性の形式的な検証する工程と動作の詳細を実装する工程の二つの開発工程が存在する．この二つの工程はそれぞれ別々の言語を用いてそれぞれ進められているため，開発コストの上昇や設計と実装の整合性を損ないやすいなどの問題がある．

この問題に対して一つのコードから様々な処理系に対応するように設計された上位ハードウェア記述言語 Melasy[2]が開発されている．Melasy は上位記述から XML 形式の中間コードを生成する．本報告は Melasy コードから生成された XML 形式の中間コードから，実装向けハードウェア記述言語 VHDL[3]のコードを生成する，コードジェネレータの提案と生成事例について述べたものである．コードジェネレータは Python で実装されており，Melasy の中間コードである.xml ファイルを入力とし，VHDL のコードである.vhd ファイルを出力する．実装事例として，ノード優先度付きバス調停器（アービタ）[4]を取り上げ，記述性と VHDL 生成を確かめた．

A Meta Hardware Description Language Melasy for Model-Checking Systems and its VHDL Code Generation

† Tatsuo Nomura, Dept. of Information Engineering, Faculty of Engineering, Shinshu University
†† Naoki Iwasaki and Katsumi Wasaki, Graduate School of Science and Technology, Shinshu University

2. XML 中間コードから VHDL への変換

Melasy は様々な処理系向けのコードを生成することを目的とした上位ハードウェア記述言語であるため，本報告では Python のクラスとして VHDL モジュールを実装し，中間コードから必要な部分を読み込みながら VHDL モジュールクラスに信号線や式などを追加し，最後に VHDL モジュールクラスによって VHDL コードを出力する手順をとっている（図 1）．

```
vhdl_module
  -signal: A   -signal: B
  -port: X
  -exp A = X
```

図 1 生成される VHDL コードモジュールの構成

3. バスアービタの Melasy での実現

実装事例として，ノード優先度付きバス調停器（アービタ）[4]を取り上げ，Melasy の記述性を確かめた．バスアービタは，データ転送するためのバスを複数のノードで共有するときに，データの衝突が起きないよう各々の上位モジュールからのバス使用要求に基づいて，バスの使用権をどのノードに与えるか決めるための回路である．アービタの仕様として，(1)優先度に応じて唯一のバスマスタが定まる，(2)リクエスト後，バスの使用権は有限時間内に得ることができる，などが挙げられる．これらの性質は，別に作成した Melasy 向け NuSMV コード生成の後，モデル検査される．

今回実装したバスアービタの 1 ノード分の回路図を図 2 に示す．各ノードは優先度を示す ID(*an*)を変えた回路を，ワイヤード OR で構成されたバス(*AB*)を共有する構成をとっている．バスに接続されたモジュールを識別するための ID の bit 数分のノード回路が各アービタに存在する．い各モジュールは接続されているバスアービタに対して *Comp* 信号をアサートすると，アービタ同士で調停が行われる．その結果モジュールがバス使用権を獲得できたならば，接続されたアービタから *Win* が返される．複数のモジュールから同時にバスの使用権を求められた時には，モジュール ID に応じて優先的にバス使用権を与えるモジュールを一つだけ決める．

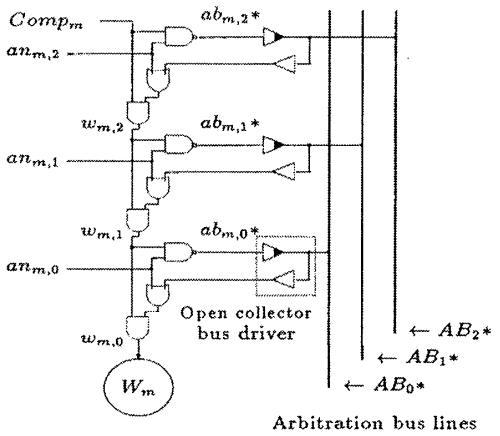

図2 ノード優先度付きバス調停器の回路

このバスアービタの Melasy による記述（一部抜粋）をコード例[A]に示す.

[A] バスアービタの Melasy による記述例(抜粋)

```
struct BusArbiterNode: public Component {
    Logic w, ab; // out
    Logic comp, an, ab_wired_back; // in
    BusArbiterNode() {
        in(comp);in(an);in(ab_wired_back);
        out(w); out(ab);
        async(ab, !(comp & an));
        Logic tmp = ab_wired_back|an;
        async(w,   comp & (ab_wired_back | an));
    }
};
template<int N> struct BusArbiter : public
 Component {
    Logic ab[N], w;
    Logic comp, an[N], ab_wired_back[N]; // in
    BusArbiterNode node[N];
    BusArbiter() {
        in(an); in(comp); in(ab_wired_back);
        out(ab); out(w);
        for(int i=0; i<N; i++) {
            PortMap pm[]= {
             node[i].comp<=(i==0?comp:node[i-1].w),
             node[i].an<=an[i],
             node[i].ab_wired_back<=ab_wired_back[i]
            };
            instance(node[i], pm);
            async(ab[i], node[i].ab);
        }
        async(w, node[N-1].w);
    }
};
```

このコードの上半分はバスアービタ内のノード回路のコードである．下半分はバスアービタ本体のコードであり，接続するモジュール数に応じてノード回路を N 個接続した構成になっている．今回は 5 個のモジュールを接続すると仮定した場合を確かめた．モジュールが 5 個であるので，モジュールを示す ID は 3bit で足りる．よってバスアービタは N=3 とし，ノード回路を 3 つ接続した形になる．

4. VHDL コード生成

実際に Melasy で生成される XML 形式の中間コードを今回作成した VHDL コード生成器に与えたときに得られたソースコード（一部抜粋）をコード例[B]に示す.

[B] 生成されたバスアービタの VHDL コード例

```
entity BusArbiterNode is
  port ( comp : in std_logic;
         ab : out std_logic;
         ab_wired_back : in std_logic;
         w : out std_logic;
         an : in std_logic);
end BusArbiterNode;
architecture melasy_generated of BusArbiterNode is
begin
  ab <= not (comp and an);
  w <= (comp and (ab_wired_back or an));
end melasy_generated;
entity BusArbiter_3 is
  port (ab_0 : out std_logic;
        .........中略.........
  BusArbiterNode_2 : BusArbiterNode port map (
    comp => BusArbiterNode_1_w,
    ab => BusArbiterNode_2_ab,
    ab_wired_back => ab_wired_back_2,
    w => BusArbiterNode_2_w,
    an => an_2);
end melasy_generated;
```

生成されたコードは論理合成ツールとターゲットレイアウト構成ツールによって遅延付シミュレーションを実施し動作を確かめた．バスアービタだけでは機能が確かめられないので，バスの使用権を要求する仮想的な上位モジュールを 5 つ別に作成し，各々バスアービタを接続した．シミュレーションの結果，実際に複数のモジュールから同時にバスの使用権を要求されたときでも，優先度に応じたバス調停動作が行われ，前節で示したアービタとしての性質が満たされていることを確かめた.

5. まとめ

Melasy から生成された XML 形式の中間コードから，VHDL のコード生成器を開発した．Melasy の記述性と VHDL コード生成の確認のため，ノード優先度付きバスアービタを記述，生成を試行した．各ノードは ID だけ異なった同一回路で構成されており，Melasy での上位記述でノード数を N で抽象化して大変簡素な記述に成功している．今後の課題として，(1)正しくない VHDL を生成したときの意味的エラー解析と報告，(2) Melasy 側へのバックアノテーション，などが挙げられる．更に，ハードウェア・ソフトウェア協調設計等への適用を試みたい.

参考文献

[1] E.M.Clarke, O.Grumberg, D.Peled, "Model Checking," MIT Press (2000).

[2] N.Iwasaki, K.Wasaki, "A Meta Hardware Description Language Melasy for Model-Checking Systems," Proc. of the 5th Int'l Conf. on Information Technology : New Generations (ITNG2008), pp.273-278 (2008).

[3] P.J.Ashenden,"The Designer's Guide To VHDL Second Edition"Morgan Kaufmann (2002).

[4] 時藤，黒川，古賀，"セルフテスティングバスアービタの一実現法について，" 電子情報通信学会論文誌 D-I, Vol.75, No.1, pp.30-40 (1992).