

# TOE40G-IP コア

2018/10/29

データシート

Rev1.0J



本社: 〒184-0012  
東京都小金井市中町 3-23-17  
電話/FAX: 050-3588-7915  
E-mail: sales@dgway.com  
URL: [www.dgway.com](http://www.dgway.com)

## 特長

- 純ハードワイヤードの TCP/IP プロトコル・スタック
- IPv4 に対応
- 単一セッション接続 (複数インスタンスすることでマルチ・セッションに対応可能)
- サーバおよびクライアントの両モード (パッシブ/アクティブのオープン/クローズに対応)
- 送受信同時の高速データ転送(Full Duplex)対応
- ジャンボ・フレームに対応
- 制御インターフェイスは汎用的なレジスタ・インターフェイス
- データ・インターフェイスはシンプルかつ一般的な 256 ビット幅の FIFO インターフェイス
- 32 バイト(256 ビット)単位での送受信パケット・サイズ (パケット・サイズと総転送サイズは 32 バイト単位とする必要があります)
- 送信/受信バッファ・サイズを消費リソースとパフォーマンスから最適点に調整可能
- Xilinx 製 40-Gbps イーサネット MAC と接続する 256 ビット FIFO インターフェイス
- クロック周波数は 200MHz 以上を推奨
- ZCU102/ZCU106 による実機動作リファレンス・デザインを用意(Full Duplex の同時送受信デザインを含みます)

コア概要	
コア納品同梱物	
技術資料	データシート、リファレンスデザイン資料
コア納品形態	暗号化した VHDL
制約ファイル	リファレンス・デザインで提供
検証環境	ユーザ I/F 部のシミュレーション 実機動作リファレンス・デザイン
インスタンス化サンプル	VHDL で提供
リファレンス・デザイン	VIVADO プロジェクト コア以外は全てソースコード提供
その他	ZCU102/ZCU106 での実機デモ環境
シミュレーション・ツール	
ModelSim SE	
技術サポート	
デザイン・ゲートウェイ・ジャパンによる日本語サポート	

表 1: Ultrascale シリーズ・ファミリ コンパイル結果例

ファミリ	ターゲット・デバイス	Fmax (MHz)	CLB Regs	CLB LUTs	CLB <sup>1</sup>	BRAMTile <sup>2</sup>	Design Tools
Zynq-Ultrascale+	XCZU9EG-FFVB1156-2	300	5268	5271	996	34.5	Vivado2017.4

備考:

1. 実際のリソース消費カウントはユーザロジックやフィット条件等に依存します。
2. ブロックメモリの消費リソース数は送信データ・バッファ 64K バイト、受信データ・バッファ 64K バイト時の値となります。バッファ容量の最小設定はいずれも 16K バイトです。

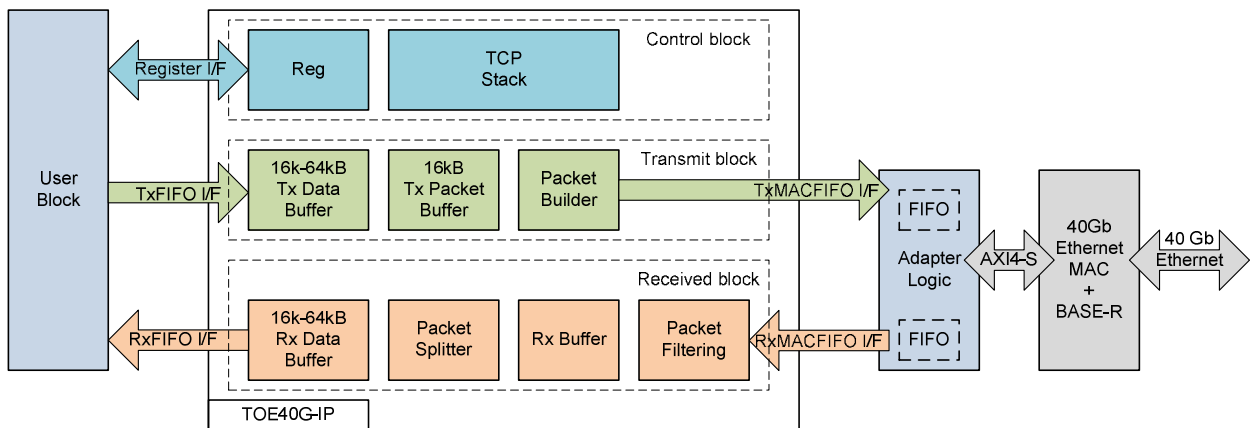


図 1: TOE40G-IP ブロック図

## アプリケーション情報

本 TOE40G-IP コア (以下本 IP コアと略します) は 40Gbit イーサネットにて TCP/IP プロトコルによりデータ信頼性を維持しながら同時に高速転送が要求されるネットワーク・アプリケーション向けのコアです。本コアは CPU なしのハード・ロジックのみで TCP/IP による 40Gb イーサネット通信を可能とします。また、データ送信とデータ受信は独立して動作するため、同時送受信 (フル・デュプレックス) に対応します。

## コア概要

本 IP コアは TCP/IP スタックを内蔵します。ネットワーク・プロトコル層にて、より低層レイヤを構築する Xilinx 製 40GbE EMAC IP コアおよび 40GbE イーサネット PCS/PMA(BASE-R)と接続します。ユーザ回路はシンプルな回路で本 IP コアと接続するにより 40Gb イーサネット・データの送受信が可能となります。

ユーザ・インターフェイスは 2 つのグループに分かれます、すなわちひとつは制御信号でもう一つはデータ信号です。制御信号は一般的なレジスタ・インターフェイスで実装され、データ信号は汎用の 256 ビット FIFO インターフェイスで接続されます。

レジスタ・インターフェイスは MAC アドレス、ポート番号、IP アドレス等のネットワーク・パラメータを設定するために使われます。また、IP コアに対してアクティブ・モードでのコネクション・オープン、データ送信、アクティブ・モードでのコネクション・クローズなどを指示するコマンドを送るためにも使われます。データ送信時の転送サイズやパケット・サイズも同じくレジスタ・インターフェイスを介して設定されます。

ユーザ回路とのデータ・インターフェイスは送信・受信とも 256 ビット幅の FIFO インターフェイスを使います。ユーザ回路は本 IP コアへのデータ送信やコアからのデータ受信に一般的な FIFO タイミングに従ったシンプルな回路でデザイン可能です。

TCP/IP プロトコルに従い 2 デバイス間でのデータ転送の前にコネクションをオープンする必要があります。本 IP コアはアクティブ・モード (ポートが本 IP コアによりオープンされる) とパッシブ・モード (ポートが外部デバイスのよりオープンされる) の両方をサポートします。新たなコネクションが確立された後本 IP コアは外部デバイスと新たにオープンしたポート番号を使ってデータの送信あるいは受信が可能となります。全データの転送が完了した後にコネクションは本 IP コア (アクティブ・モード) あるいは外部デバイス (パッシブ・モード) でクローズできます。

## データ・インターフェイス

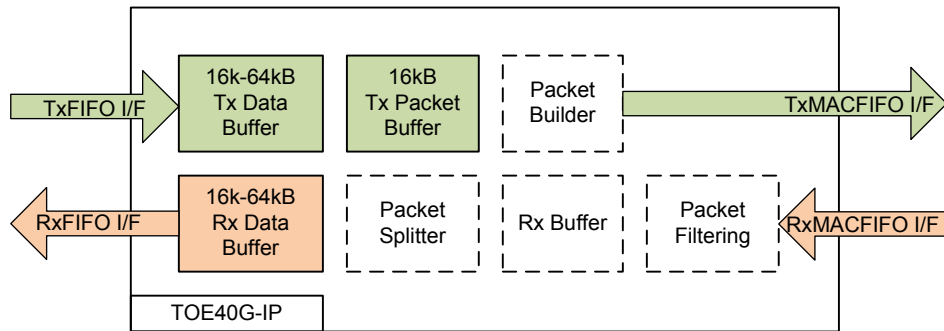


図 2: サイズが調整可能な送信/受信バッファ

転送速度を加速するため本 IP コアには 3 種類のバッファが内蔵されています、すなわち図 2 に示す送信データ・バッファ、送信パケット・バッファ、受信データ・バッファです。このうち送信データ・バッファと受信データ・バッファのサイズは本 IP コアのパラメタライズ変数で選択可能です。データ・バッファのサイズを大きくすると転送パフォーマンスが向上します。転送パフォーマンスと FPGA 内部メモリ・リソース消費量はトレード・オフ関係にあります。

送信データ・バッファ・サイズは本 IP コアから外部デバイスへの送信パフォーマンスに影響し、受信データ・バッファ・サイズは外部デバイスから本 IP コアへの受信パフォーマンスに影響します。本 IP コアの送信パケット・バッファのサイズは 16K バイトに固定されています。

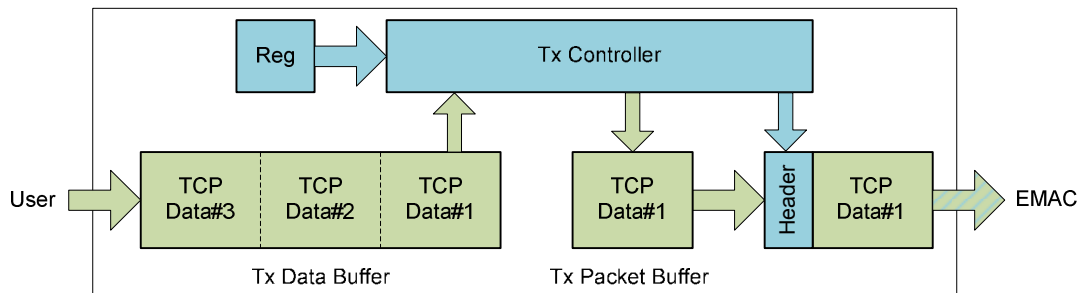


図 3: データ送信フロー

TCP パケットにデータを内蔵するため、送信データ・バッファから 1 パケット分のデータが送信パケット・バッファへ転送されます。そしてネットワーク・パラメタがレジスタからロードされ送信コントローラ内にて各パケットのヘッダが生成されます。ヘッダ生成処理が完了するとヘッダはパケットの先頭部に挿入され連続したパケット・データとなります。こうして完成した TCP パケットは EMAC へと転送されます。また送信コントローラでは TCP チェック・サムと IP チェック・サムが計算されます。本 IP コアのビジー信号(レジスタ・インターフェイスを通してリード可能)は外部デバイスへのデータ転送が全て完了すると '0' ネゲートします。全データ転送サイズとパケット・サイズはレジスタ・インターフェイスを介してユーザ回路から設定できます。

本 IP コアから外部デバイスへのデータ送信中 EMAC からの受信パケットがモニタされ外部デバイスでの受信バッファ状態が監視されます。本 IP コアは外部デバイス側の受信バッファに十分な空きスペース( 1 パケット・サイズ以上の空き領域)がある場合に限ってデータを送信します。送信済みデータに対応して返送された ACK 番号も常時監視されており、外部デバイス側で完全に受信したデータ位置がチェックされます。TCP/IP プロトコル規格に従い重複 ACK(同じ ACK 番号を前のパケットと連続して受信)を検出した場合、本 IP コアは外部デバイスへロスト・パケットを再送します。ロストした位置は重複 ACK パケットの ACK 番号から抽出されます。

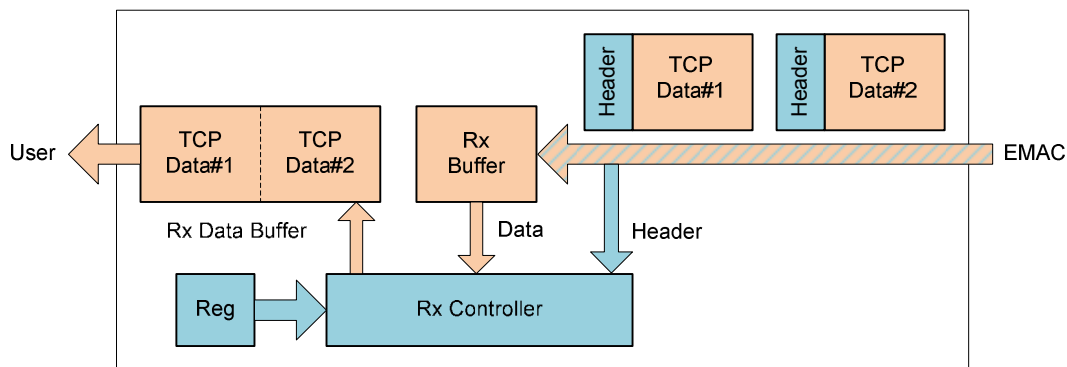


図 4: データ受信フロー

本 IP コアが外部デバイスからのデータを受信すると受信パケットは受信バッファに格納され、最初にヘッダとチェックサムが評価されます。受信パケット内のネットワーク・パラメータはユーザ回路により設定されていた設定値と合致する必要があります。合致していない場合かチェックサムがエラーであった場合、そのパケットは破棄(無視)されます。有効なパケットの場合、パケット内のデータは抽出され受信データ・バッファへと転送されます。受信パケットのヘッダはパケットの有効性評価には使われますが受信データ・バッファへは転送されません。

有効なパケットを受信すると本 IP コアは完全に受信できたデータ位置を示す ACK 番号を返送します。受信パケットのデータ位置が正しい順番に並んでいない場合やパケットのロストが検出された場合、本 IP コアは重複 ACK を発生することでロスト・パケットの再送を要求します。本 IP コアのビジー信号は重複 ACK パケット生成中'1'にアサートされます。

ただし本 IP コアにはパケット・シーケンスが 1 パケット分逆転した場合に限って受信パケットの順番を入れ換える機能があります。たとえば受信パケットのシーケンスが#1→#3→#2→#4 のようなケースでは、自動的に受信パケットの順番を正しく#1→#2→#3→#4 に入れ換えます。しかし受信パケット順番の逆転が 2 パケット以上たとえば#1→#3→#4→#2 のケースでは、本 IP コアは処理できないためパケット#3 と#4 の受信後に重複 ACK を送信してパケット#2 の再送を要求します。

## コアの機能ブロック

本 IP コアは、制御ブロック、送信ブロック、受信ブロックの3ブロックに分かれています。

### 制御ブロック

- レジスタ

ユーザ回路は TCP/IP オペレーションに関するパラメータをレジスタ・インターフェースにより設定できます。レジスタ・アドレスは全 4 ビット(最大レジスタ数=16)が実装されています。それぞれのレジスタのアドレス・マッピングを表 2 に示します。RST レジスタによるリセットが解除されると、各パラメータを設定したレジスタを初期値として動作が開始されます。

- TCP スタック

TCP スタックは TCP/IP 規格に沿ってパケット送信および受信シーケンスを制御するステート・マシンを内蔵します。データ送信時はネットワーク・パラメータや送信データ・サイズ、パケット・サイズなどのコマンド・パラメータが TCP スタックにより送信ブロックへ設定されます。その後 TCP スタックは全データの送信が完了するまで送信ブロックの状態を監視します。

データ受信時 TCP スタックは受信ブロックからの受信パケット・タイプを監視します。ACK パケットの返送が必要な場合、TCP スタックは送信ブロックに対してパケット返送を指示します。

表 2: レジスタ・マップ定義

レジスタ・アドレス [3:0]	レジスタ名	方向	ビット	説明
0000b	RST	Wr/ Rd	[0]	IP リセット。'1'でリセットし'0'でリセットを解除する。初期状態は'1'(リセット状態)。ユーザ回路が動作に必要な全パラメータをレジスタにセットしてから本レジスタに'0'を書き込むことでコアの動作が開始する。ユーザ回路が SML, SMH, DIP, SIP, DPN, SPN レジスタの値を変更する必要が生じた場合、本レジスタを一旦'1'としコアをリセット状態に移行させてから変更しなくてはならない。
0001b	CMD	Wr	[1:0]	アクティブ・モードでのユーザ・コマンド。"00":データ送信,"10":オープン・コネクション(アクティブ),"11":クローズ・コネクション(アクティブ),"01":未定義(指定しないこと)。本レジスタによってアクティブ・コマンドを開始する前に、ユーザ回路は本レジスタの bit[0]をチェックしてコアは動作中でないことを確認しなくてはならない。本レジスタをセットするとコアは指定されたアクティブ・コマンド動作を自動的に開始する。
			Rd	[0]
			[3:1]	現在の動作状態,"000":データ送信中,"001":アイドル状態,"010":アクティブ・オープン・コネクション,"011":アクティブ・クローズ・コネクション,"100":データ受信中,"101":初期化中,"110":パッシブ・オープン・コネクション,"111":パッシブ・クローズ・コネクション
0010b	SML	Wr	[31:0]	コアの MAC アドレスの下位 32bit 定義レジスタ。RST レジスタをクリアする前に本レジスタで MAC アドレスを指定する必要がある。
0011b	SMH	Wr	[15:0]	コアの MAC アドレスの上位 16bit 定義レジスタ。RST レジスタをクリアする前に本レジスタで MAC アドレスを指定する必要がある。
0100b	DIP	Wr	[31:0]	接続ターゲット・デバイス側の IP アドレス 32bit を指定する。RST レジスタをクリアする前に本レジスタで IP アドレスを指定する必要がある。
0101b	SIP	Wr	[31:0]	本システム側の IP アドレス 32bit を指定する。RST レジスタをクリアする前に本レジスタで IP アドレスを指定する必要がある。
0110b	DPN	Wr	[15:0]	コネクションを行う接続ターゲット・デバイス側のポート番号を 16bit で指定する。アクティブ・オープンでコネクションを行う場合は RST レジスタをクリアする前に本レジスタでターゲット側のポート番号を指定する必要がある。パッシブ・オープンの場合、本 IP コアに設定されたネットワーク・パラメータに合致したオープン・パケットの受信にて、ターゲットのポート番号は自動的に決定される。(注 2)
0111b	SPN	Wr	[15:0]	本システム側のポート番号を 16bit で指定する。RST レジスタをクリアする前に本レジスタで自身のポート番号を指定する必要がある。
1000b	TDL	Wr	[31:0]	送信データ数をバイト単位で指定する、ただし値は 32 バイトの倍数でなくてはならない。有効な値は 32~0xFFFFFEE0 (bit[4:0]の最下位 5 ビットは無視される)。CMD レジスタに"00"をセットする前に本レジスタで送信データ数をセットする必要がある。この値はコア内部ロジックにて CMD レジスタがセットされた瞬間にサンプリングされるため、現在の送信が完了していない場合でも次の送信用の値をセットできる。また、再度前回と同じデータ数を送信する場合は本レジスタに再セットする必要はない。
		Rd	[31:0]	まだ送信されていない残りデータ数をバイト単位で表示する。

レジスタ・アドレス [3:0]	レジスタ名	方向	ビット	説明
1001b	TMO	Wr	[31:0]	全てのコマンドにて、受信パケットの待ち時間タイムアウト値を設定する。タイムアウト時間は Clk 信号周波数に依存する。たとえば Clk 周波数が 200MHz の場合タイマーの時間設定は 5nsec 単位となる。この値は実際のネットワーク環境にてパケットが通常状態では発生せずロストした場合にタイムアウトを発生するレイテンシ時間以上の値に調整する必要がある。
		Rd		各タイムアウト等のステータス、それぞれのビット定義は以下の通り [0] ARP で返信パケットをタイムアウト時間内に受信しなかった タイムアウト後、本 IP コアは ARP 応答を受信するまで ARP 要求を再送し続ける [1] アクティブ・オープン時に SYN と ACK フラグをタイムアウト時間内に受信しなかった タイムアウト後、本 IP コアは SYN パケットを最大 16 回まで再送し、その後は FIN パケットを送信してコネクションをクローズする [2] パッシブ・オープン時に ACK フラグをタイムアウト時間内に受信しなかった タイムアウト後、本 IP コアは SYN ACK パケットを最大 16 回まで再送し、その後は FIN パケットを送信してコネクションをクローズする [3] アクティブ・クローズ時に FIN と ACK フラグをタイムアウト時間内に受信しなかった タイムアウト後、本 IP コアは直ちに RST パケットを送信しコネクションをクローズする [4] パッシブ・クローズ時に ACK フラグをタイムアウト時間内に受信しなかった タイムアウト後、本 IP コアは FIN ACK パケットを最大 16 回まで再送信し、その後は RST パケットを送信してコネクションをクローズする [5] データ送信時に ACK フラグをタイムアウト時間内に受信しなかった タイムアウト後、本 IP コアは前回送信したパケットを再送する [6] 受信パケットをロストした、受信データ FIFO が一杯になった、あるいは誤ったシーケンス番号などの要因により、データ受信中タイムアウトとなった この場合本 IP コアは重複 ACK パケットを送信し再送を要求する [22] データ送信が完了していない状況で FIN パケットを受信した [23] 受信データ FIFO が一杯のため受信パケットが受信できない致命的なエラー [21],[27] 受信パケットのロストを検出した [30] 受信パケット中に RST フラグが検出された [31],[29:28],[26:24] 内部テスト用信号
1010b	PKL	Wr	[15:0]	バイト単位で指定する送信パケットのデータ長、ただし値は 32 バイトの倍数でなくてはならない。有効な値は 32~8,960 (bit[4:0]の最下位 5 ビットは無視される)。デフォルト値は 1440 バイト(非ジャンボ・フレームにて 32 の倍数での最大サイズ)。 <b>本レジスタの値はデータ転送中(システム・ビジー・フラグ='1')は変更してはならない。次の転送でも同じパケット・サイズの場合、ユーザは本レジスタの値を再セットする必要はない。</b>
1011b	PSH	Wr/ Rd	[1:0]	送信モードの指定 (デフォルト値は全ビット'0')、本レジスタはアイドル中(CMD レジスタの bit0='0')のみ変更可能。注 3 参照 [0] '0': TDL ≤ PKL の場合(1 送信指示が 1 パケット)に同一パケットを自動再送する '1': TDL ≤ PKL の場合の自動再送機能を禁止する (1 パケットのみの送信とする) [0] '0': 全ての送信パケットで PSH フラグを'0'とする '1': 全ての送信パケットで PSH フラグを'1'とする)

レジスタ・アドレス [3:0]	レジスタ名	方向	ビット	説明
1100b	WIN	Wr/ Rd	[5:0]	<p>ウインドウ・アップデート・パケットの自動送信を決める 1K バイト単位での閾値 初期値は 0(ウインドウ・アップデート・パケットを自動送信しない) 本 IP コアは受信データ・バッファの空き容量が、前回パケットを送信した時点のウインドウ・サイズより本レジスタで設定した容量分の空きが増えた時点でウインドウ・アップデート・パケットを自動送信する 例えば WIN="000001b"つまり 1K バイトの閾値を設定し前回送信パケットでウインドウ・サイズが 2K バイトで報告した場合、ユーザ回路から受信データを 1K バイト分リードしてその分受信データ・バッファの空き容量が増えた時点で本 IP コアはウインドウ・サイズを 3K バイトとしてウインドウ・アップデート・パケットを自動送信する</p>
1101b	ETL	Wr	[31:0]	<p>拡張された総送信データ数をバイト単位で指定する。本サイズは 32 の倍数でなくてはならず本 IP コアは bit[4:0]を無視する。 ユーザ回路は CMD='00'のデータ送信実行中に総送信長を増加するため本レジスタを設定できる、このため総転送サイズは本 IP コアに新たなコマンドを発行せずに送信データ・サイズを追加できる。 たとえばデータ転送で TDL=4GB に設定したと仮定する。そして転送残りサイズが 1GB となった時点でユーザ回路は本 ETL に 2GB をセットすることで、6GB 分の連続データ送信が可能となる。(4GB+2GB) ただし以下に注意すること 1) ETL レジスタは TDL のリード値(転送残りバイト数)が 128K バイト以上の状態でセットする必要がある。 2) ETL レジスタの設定値は(0xFFFF-FFE0 – TDL のリード値)よりも小さい値とし、コア内データ・カウンタがオーバーフローしないようにすること(データ・カウンタは 32 ビットで実装されている)</p>
1110b	SRV	Wr/ Rd	[0]	<p>'0': クライアント・モード。本 IP コアはリセット状態から解放されると、DIP レジスタで設定されたターゲットの IP アドレスから ARP 要求を送信しターゲットの MAC アドレスを取得する '1': サーバー・モード。本 IP コアはリセット状態から解放されると、ターゲットから自分に向けた ARP 要求パケットを待機する。ARP 要求を受信し ARP 応答パケットを送信した後、本 IP コアのビジー状態は'0'に解除される。<b>すなわち ARP 要求パケットを受信しない限り本 IP コアは初期化が完了しないので注意すること。</b> デフォルト値は'0'(クライアント・モード)</p>

注意:

- 1 ターゲットの MAC アドレスはクライアント・モードの場合 ARP の返信パケットにて、サーバー・モードの場合受信した ARP パケットから本 IP コアが自動検出・設定するため、ユーザ回路側でこのパラメータを設定する必要はありません。
- 2 ターゲットのポート番号はパッシブ・オープンの場合は受信パケットから自動設定されます。
- 3 PSH の bit0 における自動再送機能(デフォルトで機能 ON)は、1 パケットのみの送信の場合、受信側の PC の ACK 応答が非常に遅くなる現象を回避するための機能です。その機能が不要な場合このビットを'1'として自動再送機能を禁止してください。

表 3: 送信データ/受信データの各バッファで設定できる FIFO サイズ(アドレス・サイズで指定)

有効なビット幅	バッファ・サイズ	送信データ・バッファ	送信パケット・バッファ	受信データ・バッファ
9	16kByte	有効	有効	有効
10	32kByte	有効	有効	有効
11	64kByte	有効	有効	有効

## 送信ブロック

### • 送信データ・バッファ

送信データ・バッファのサイズは本 IP コアの "TxBufBitWidth" パラメータで指定します。有効な値の範囲は表 3 に示すように 9-11 で 256 ビット・バッファのアドレス・サイズを意味します。

本バッファのサイズは PKL レジスタで設定した送信パケット・サイズより2倍以上大きいサイズでなくてはなりません。ユーザ回路からの送信データは一旦本バッファに格納されます。このバッファ内データは送信先ターゲットから ACK パケットを受け取りデータ転送が確認されるとフラッシュされます。本バッファからのデータは送信パケット・バッファに転送され次のパケット送信用に保持されます。

本バッファ・サイズは送信パフォーマンスに影響します。より大きなサイズに設定するとターゲットからの ACK パケット受信を待たずに連続してデータが送信できます。このためネットワーク・デバイスからの遅延時間を低減する効果もあります。

ユーザ回路が総転送サイズの設定数より多くのデータを本バッファに書き込んだ場合、バッファ内に残ったデータは次の転送に使われます。ポートがクローズされるかあるいはリセットを検出すると、格納されていたデータはフラッシュ(クリア)されます。逆にユーザ回路から送信したデータ数が現在のトランザクションに不足していた場合、本 IP コアはパケットを送信せずユーザ回路から必要なデータが書き込まれるまで待機します。

### • 送信パケット・バッファ

このバッファのサイズは 16K バイト固定で 1 データ・パケットを格納します。送信パケット・バッファ内のデータは TCP スタックからの要求があり EMAC が新たなデータを受信できる場合に転送されます。

### • パケット・ビルダ

TCP パケットはヘッダ部とデータ部から構築されます。パケット・ビルダはレジスタ・モジュールからのネットワーク・パラメータを受け取り TCP ヘッダを生成します。更に IP チェック・サムおよび TCP チェック・サムが計算され TCP ヘッダを完成させます。そして内部ロジックからのヘッダとそれに続くデータの送信パケットは EMAC へと転送されます。



## 受信ブロック

- **受信バッファ**

このバッファはヘッダ・チェックで処理される前のイーサネット MAC からの全ての受信パケットを一時的に保持します。有効なパケットのうち TCP データのみが抽出され受信データ・バッファへ転送されます。

- **パケット・フィルタリング**

受信パケット内のヘッダを比較します。ヘッダの内容は Reg モジュールで設定されたネットワーク・パラメータと適合する必要があります。また、本モジュール内で IP および TCP チェック・サムが計算されます。パラメータが合致、かつチェック・サムが正しいパケットのみパケット・スプリッタへ転送されます。

- **パケット・スプリッタ**

受信パケットから TCP データのみを抽出し受信データ・バッファへ転送します。また、受信データが重複した場合(受信済みと同一のパケットを再受信した場合)はそのパケットは無視されます。

- **受信データ・バッファ**

受信データ・バッファのサイズは IP コアの "RxBufBitWidth" パラメータで指定します。有効な値の範囲は表 3 に示すように 9-11 です。バッファ・サイズは TCP コネクションのウインドウ・サイズに割り当てられます。本バッファ・サイズを大きくすると接続先ターゲット・デバイスにて本 IP コアからの ACK を待たずに連続してデータを送信できるため、受信パフォーマンスが改善されます。また、受信データ・バッファのサイズを増加すると、ネットワーク・デバイス側での遅延時間を低減する効果があります。

## ユーザ回路

ユーザ回路はレジスタ I/F を通してパラメータの設定やコア状態のモニタを行い、また、送信 FIFO I/F を介した送信データの書き込みや受信 FIFO I/F を介して受信データの読み出しを行います。ユーザ回路はシンプルなハードウェア・ロジックで実装できます。

## 40Gb/50G イーサネット MAC および BASE-R

リファレンス・デザインにおいては Xilinx から提供される EMAC および BASE-R 向け PCS/PMA から構成される 40Gb/50Gb イーサネット・サブシステムが使われます。より詳細については以下のウェブサイトを参照してください。(MAC についてはお客様にて Xilinx 製 IP コアを手配して頂く必要があります)

<https://www.xilinx.com/products/intellectual-property/ef-di-50gemac.html>

## コアの I/O 信号

コアに設定するパラメータおよび I/O 信号をそれぞれ表 4 と表 5 で説明します。本 IP コアの MAC インターフェイスは FWFT モードでの 256 ビット FIFO 標準です。

表 4: コアのパラメータ

パラメータ名	設定範囲	説明
TxBufBitWidth	9-11	送信データ・バッファのサイズをバッファのアドレス・バスのビット幅で指定します。
RxBufBitWidth	9-11	受信データ・バッファのサイズをバッファのアドレス・バスのビット幅で指定します。

表 5: コアの I/O 信号

信号名	方向	説明
<b>共通 I/F 信号</b>		
RstB	In	IP コアのリセット: ロウ・アクティブ信号である。
Clk	In	Xilinx ブロックの PHY 層からの 156.25 MHz 固定クロック
<b>ユーザ I/F</b>		
RegAddr[3:0]	In	レジスタの 4bit アドレスバス
RegWrData[31:0]	In	ライト・レジスタの 32bit 書込みデータ・バス
RegWrEn	In	レジスタのライト・イネーブル、アドレスおよびデータに有効な値をセットし本信号にパルスを与えることで書込みを実行する。
RegRdData[31:0]	Out	レジスタの 32bit 読み出しデータ・バス、レジスタアドレスをセットしてから1クロックのレイテンシ後に有効なリード・データが本バス上に現れる。
ConnOn	Out	コネクション状態('1':コネクションはオープン状態、'0':コネクションはクローズ状態。
TimerInt	Out	タイマ割り込み、タイムアウト発生時に本信号が 1 クロック期間分 H アサートされる。ユーザ回路は TMO[6:0]レジスタを読むことで割り込み要因を確認することができる。
RegDataA1[31:0]	Out	32 ビットの CMD レジスタ(RegAddr=0001b)リード値、Bit[0]は本 IP コアのビジー・フラグ
RegDataA8[31:0]	Out	32 ビットの TDL レジスタ(RegAddr=1000b)リード値、ETL レジスタを使う場合に残り転送サイズをモニタするために使う
RegDataA9[31:0]	Out	32 ビットの TMO レジスタ(RegAddr=1001b)リード値、タイムアウト発生時の割り込みステータスとして使う
<b>送信 FIFO I/F</b>		
TCPTxFfFlush	Out	コアの送信データ・バッファがクリアされたことを示す。コネクションのクローズ時やリセット実行時に、1 クロック期間 H アサートされる。
TCPTxFfFull	Out	コアの送信データ・バッファの Full フラグ。ユーザ回路は本信号が H アサートされてから 4 クロック期間以内に送信データの書込みを停止しなくてはならない。
TCPTxFfWrEn	In	送信データ・バッファのライト・イネーブル。送信データを書き込む際にアサートする。
TCPTxFfWrData[255:0]	In	送信データ・バッファの 256bit 書込みデータ・バス、TOETxFfWrEn に同期する。
<b>受信 FIFO I/F</b>		
TCPRxFfFlush	Out	コアの受信データ・バッファがクリアされたことを示す。コネクションのオープン時に、1 クロック期間 H アサートされる。
TCPRxFfRdCnt[10:0]	Out	受信データ・バッファ内の受信データ総量を 256bit 単位で示す FIFO データ・カウンタ
TCPRxFfLastRdCnt[4:0]	Out	総受信データ・バイト数が 32 の倍数ではない場合に受信バッファ内の最終 256 ビット (32 バイト)データで有効なバイト数を示す。
TCPRxFfRdEmpty	Out	受信データ・バッファの FIFOEmpty フラグ。ユーザ回路は本信号が H アサートされたら直ちにデータの読み出しを停止しなくてはならない。
TCPRxFfRdEn	In	受信データ・バッファの読み出しイネーブル。受信データを読み出す際にアサートする。
TCPRxFfRdData[255:0]	Out	受信データ・バッファの 256bit 読み出しデータ・バス、TCPRxFfRdEn をアサートしてから 1 クロック期間のレイテンシ後に有効なリードデータが出力される。

信号名	方向	説明
<b>Mac FIFO I/F (FWFT モード)</b>		
rx_axis_tdata[63:0]	In	MAC からの 64bit 受信データ・バス
rx_axis_tvalid	In	受信データの有効信号、rx_axis_tdata に同期
rx_axis_tlast	In	フレームの最終データであることを示す信号
rx_axis_tuser	In	受信フレームの最後にそのフレームがエラーを含むかどうかを示す信号、'1'は正常パケットで'0'はエラー・パケットを意味する
rx_axis_tready	Out	ハンドシェイク信号。rx_axis_tdata が受領されたときにアサートされる。
tx_axis_tdata[63:0]	Out	MAC への 64bit 送信データ・バス
tx_axis_tkeep[7:0]	Out	送信データのバイト・イネーブル信号、tx_axis_tdata に同期
tx_axis_tvalid	Out	送信データが EMAC に対して有効であることを示す、tx_axis_tdata に同期
tx_axis_tlast	Out	フレームの最終バイトであることを示す信号
tx_axis_tuser	Out	エラーが発生していることを示す信号。本 IP コアにおいては常に'0'を出力する
tx_axis_tready	In	ハンドシェイク信号。tx_axis_tdata が正常に受信したときにアサートする
MacTxFfWrCnt[15:0]	In	MacTxFIFO のライト・データ・カウンタ数を 256bit 単位で表す。この信号は FIFO フル状態を検出するために使う。FIFO カウンタが 16bit 以下の場合、上位ビットには'1'を埋める必要がある。
MacTxFfWrData[255:0]	Out	MacTxFIFO への 256bit ライト・データ、MacTxFfWrEn='1'と同時に有効
MacTxFfWrEn	Out	MacTxFIFO ヘデータをライトするときに'1'アサート。本信号は送信 1 パケット分連続して'1'アサートされる。
MacTxLastByteEn[31:0]	Out	各送信パケット最終データにおけるデータ・バイト・イネーブル。本信号は MacTxEnd='1'かつ MacTxFfWrEn='1'のときに有効である。IP コア内では本信号は 2 つの値すなわち 0x0FFF_FFFF または 0x003F_FFFF のいずれかとなる。
MacTxSizeData[15:0]	Out	256bit 単位による送信パケットの総サイズ この値は MacTxFfWrEn='1'アサート期間と同じ値に維持される。Bit[15:11]は常に'0'である。
MacTxEnd	Out	'1'アサートすると各送信パケットの最終データであることを示す。
MacRxFfEmpty	In	MacRxFIFO のエンプティ・フラグ、MacRxFIFO 内のデータがない場合に'1'アサート。
MacRxFfRdEn	Out	'1'アサートで MacRxFIFO からのデータをリードする この信号は MacRxFfEmpty='0'の時点でのみアサート可能
MacRxFfRdData[257:0]	In	MacRxFIFO からのリード・データ、この信号は MacRxFfRdEn='1'と同じタイミングで有効となる。つまり MacRxFIFO は FWFT モードで動作するよう実装しなくてはならない。全 257bit の各信号定義は以下の通り。 [255:0]: EMAC からの 256 ビット受信データ [256]: EMAC からの各受信パケットで最終データであるときに'1'アサート [257]: '0'-通常パケット, '1'-エラー・パケット このビットは最終データであることを示す bit[256]='1'と同じタイミングで'1'アサートする必要がある。

## タイミング・チャート

### IP コアの初期化

RST レジスタ=0'としたあとの初期化プロセスとしては、本 IP コアは SRV レジスタの設定すなわちクライアント・モードかサーバー・モードかによって 2 モードで動作します。

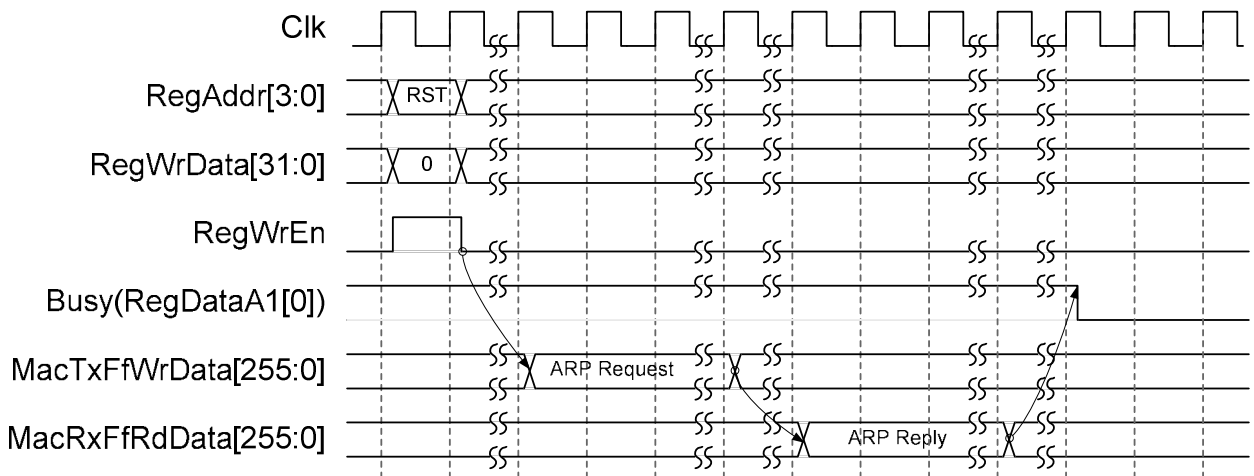


図 6: クライアント・モードの IP コア初期化

クライアント・モードの場合本 IP コアは ARP 要求を送信し、ターゲットからの ARP 応答を待ちます。ターゲット側 MAC アドレスは ARP 応答パケットから抽出します。その後ビジー信号は'0'にネゲートされます。ビジー信号は RegDataA1 出力の bit10 でモニタできます。

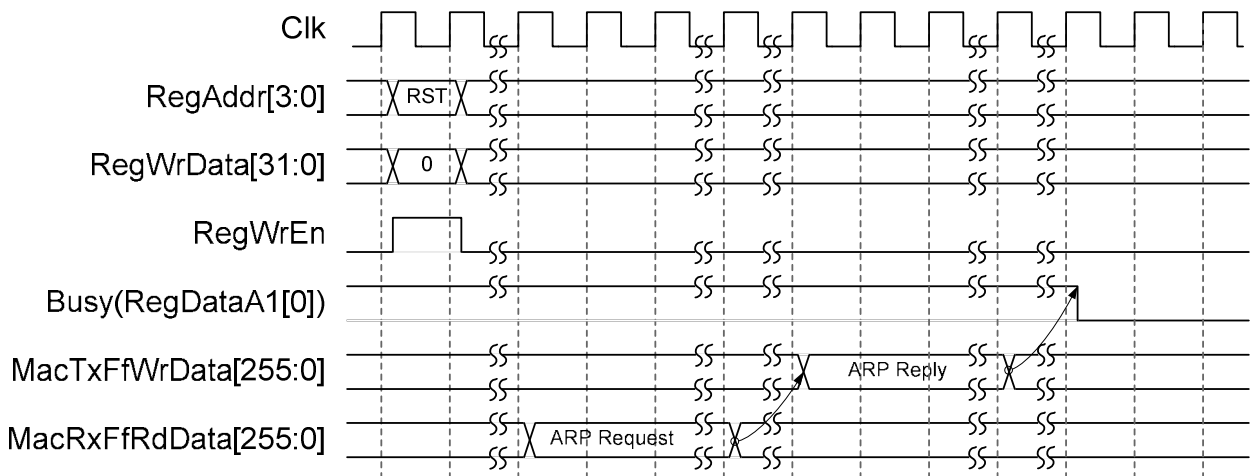


図 7: サーバー・モードの IP コア初期化

サーバー・モードの場合、リセット状態から解放された本 IP コアはターゲットからの ARP 要求を待ちます。ヘッダ情報が本 IP コアに設定されたネットワーク・パラメータに合致した ARP 要求を受け取った場合、本 IP コアは ARP 応答をターゲットに向けて送信します。ターゲット側 MAC アドレスは ARP 要求パケットから抽出します。最後にビジー信号を'0'ネゲートします。

## レジスタ・インターフェイス

ユーザ回路からの本 IP コア内部レジスタへのリード・ライト・アクセスは図 7 に示すタイミングにより実行します。アクセス先レジスタのアドレスマップは表 2 で示されます。レジスタへの書込みは RegAddr と RegWrData にそれぞれ有効なライト先のアドレスとデータをセットし 1 クロック期間 RegWrEn='1' とします。レジスタからの読み出しの際は、有効な RegAddr がコアに与えられた次のクロック期間に RegRdData にリードデータが出力されます。CMD レジスタにおいてはビジーフラグ(RegAddrA1[0]信号)を確認し、'0'ネゲート(IP コアがアイドル状態)である場合のみセットすることができます。図 8 に示すように CMD レジスタをセットするとビジー・フラグは'1'アサートされます。このビジー・フラグはコマンドが完了すると'0'ネゲートします。

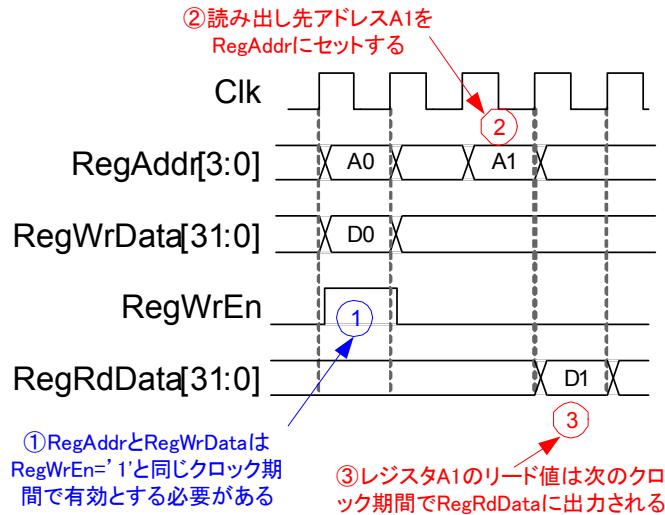
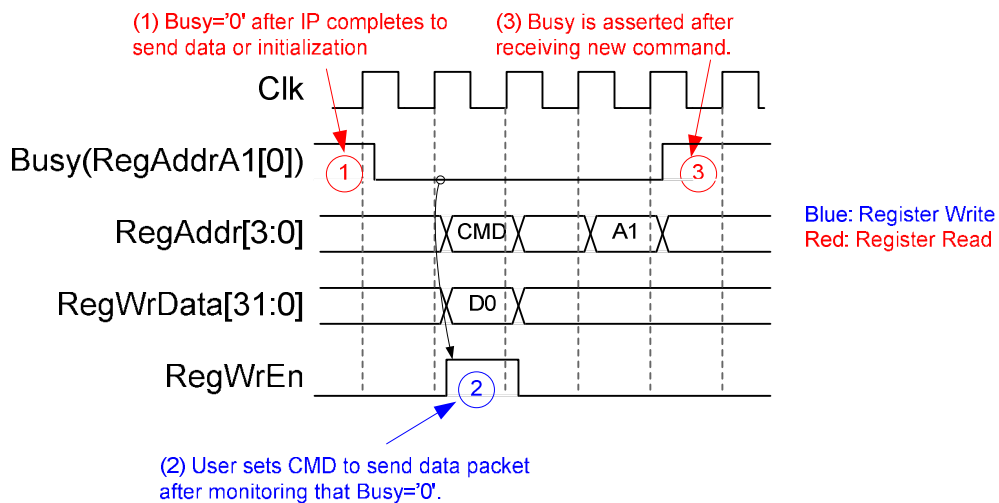


図 7: レジスタ I/F のタイミング・チャート

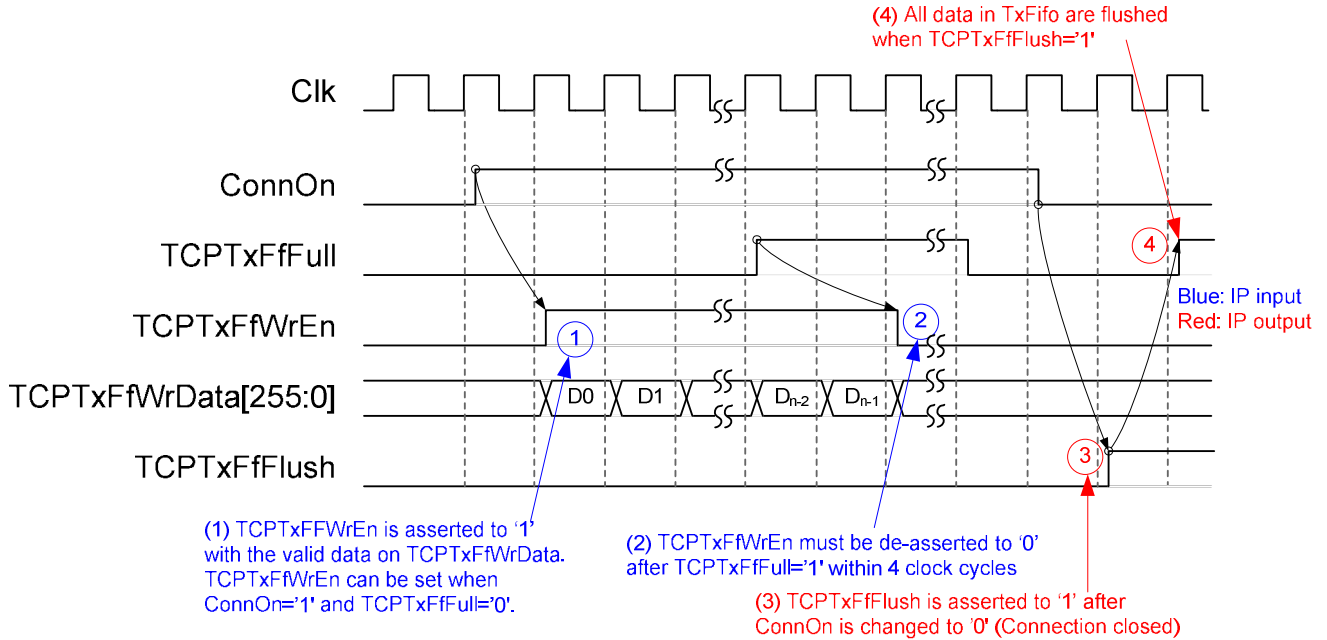


- (1) IP コアのデータ送信または初期化の完了後ビジーが'0'ネゲートされる
- (2) ユーザ回路はビジーをモニターし'0'ネゲートされたことを確認してから送信コマンドが発行できる
- (3) IP コアは新たなコマンドを受信するとビジーをアサートする

図 8: CMD レジスタのライトはビジーがネゲート時のみ可

## 送信 FIFO インターフェイス

ユーザ回路から本 IP コアを介して送信するデータは図 9 のタイミングに示す FIFO インターフェイスで書き込みます。データを送信する前にユーザ回路は Full フラグ(TCPTxFfFull)が'1'にアサートされておらず、かつ、ConnOn が'1'アサートされていることを確認する必要があります。そして TCPTxFfWrEn='1'とし送信データを TCPTxFfWrData に出力します。TCPTxFfFull が'1'にアサートされた場合は 4 クロック以内にデータの送信を停止しなくてはなりません。TCPTxFfFull は送信 FIFO がフラッシュされた場合も'1'アサートしますが、コネクションのクローズや IP コア・リセットといった 2 つのケースでもアサートします。



- (1) TCPTxFfWrData は TCPTxFfWrEn='1'と同じクロック期間で有効にする必要がある。TCPTxFfWrEn は ConnOn='1'かつ TCPRxFfFull='0'の場合にイネーブルできる。
- (2) TCPTxFfWrEn は TCPTxFfFull='1'となってから 4 クロック以内にネゲートしなくてはならない。
- (3) TCPTxFfFlush は ConnOn='0'となってから(コネクションがクローズされてから)アサートされる。
- (4) TCPTxFfFlush='1'で送信 FIFO の全データをクリアするとともに TCPTxFfFull='1'として誤ったライト動作から保護する。

図 9: 送信 FIFO I/F のタイミング・チャート

## 受信 FIFO インターフェイス

本 IP コアがターゲットから有効なパケットを受信するとデータのみを抽出して受信データ・バッファに保存します。ユーザ回路は FIFO インターフェイスを介してバッファ内の受信データを図 10 に示すタイミングで読み出すことができます。ユーザ回路は TCPRxFfEmpty をモニタしてデータの読み出しが可能かどうかを確認します。TCPRxFfEmpty が '0' にクリアされていればデータを読み出すことができます。FIFO 内受信データの読み出しは TCPRxFfRdEn を '1' にセットして行いますが、受信データは次のクロック期間に TCPRxFfRdData に出力されます。データの読み出し中に TCPRxFfEmpty が '1' にアサートされた場合、ユーザ回路は TCPRxFfRdEn を同じクロック期間内で '0' としなくてはなりません。受信データ・バッファ内の全データはコネクシヨンのオープン時にクリアされ、それは TCPRxFfFlush が '1' にアサートされることで検出できます。

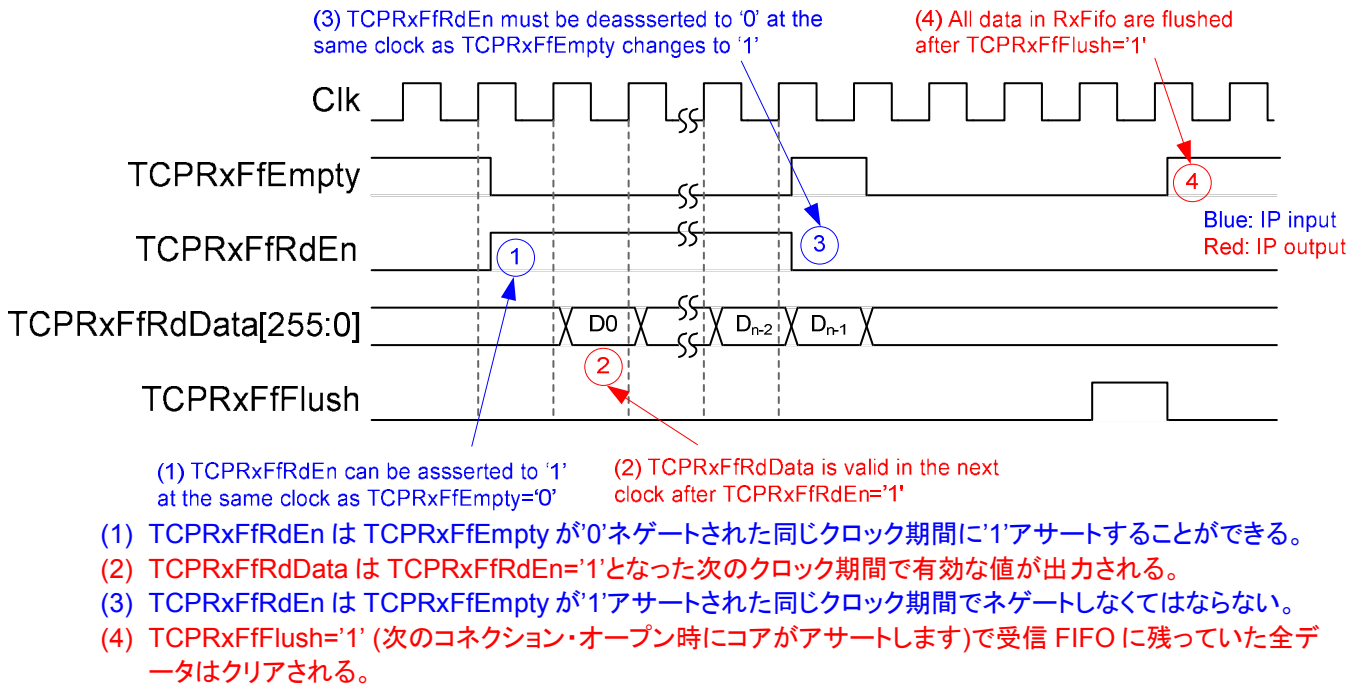


図 10: 受信 FIFO I/F のタイミング・チャート

データをバースト転送でリードする場合、TCPRxFfRdCnt を使って受信データ・バッファ内の総データ量を調べることができます。ユーザ回路は図 11 に示すように連続したクロック期間中 TCPRxFfRdEn='1' としてバースト転送を実行できます。

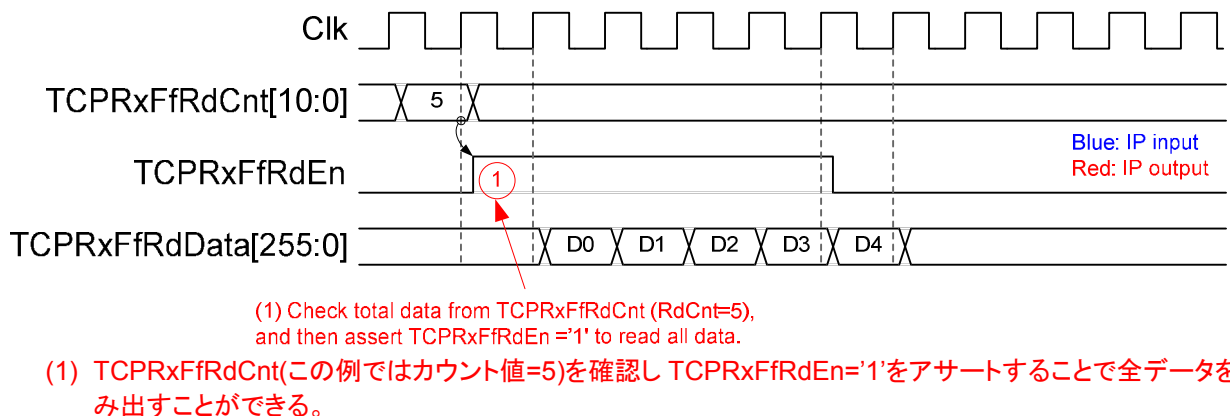


図 11: FIFO 残量カウンタ情報による受信データの読み出し

## EMAC FIFO インターフェイス

本 IP コアの MAC インターフェイスは FWFT モードの FIFO インターフェイスとなります。本 IP コアを 40Gb EMAC と接続するため図 12 に示す FWFT FIFO を含む接続ロジックが必要となります。

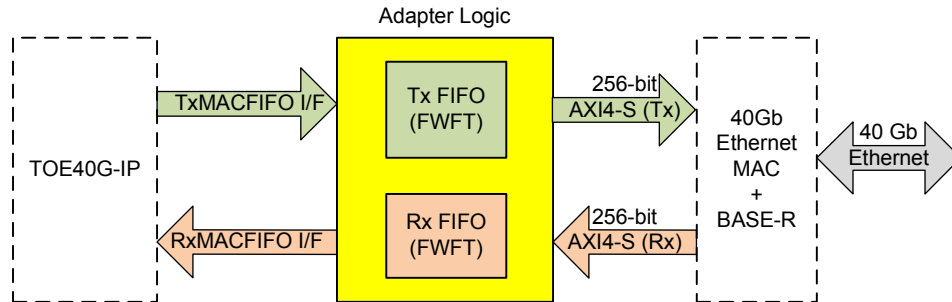
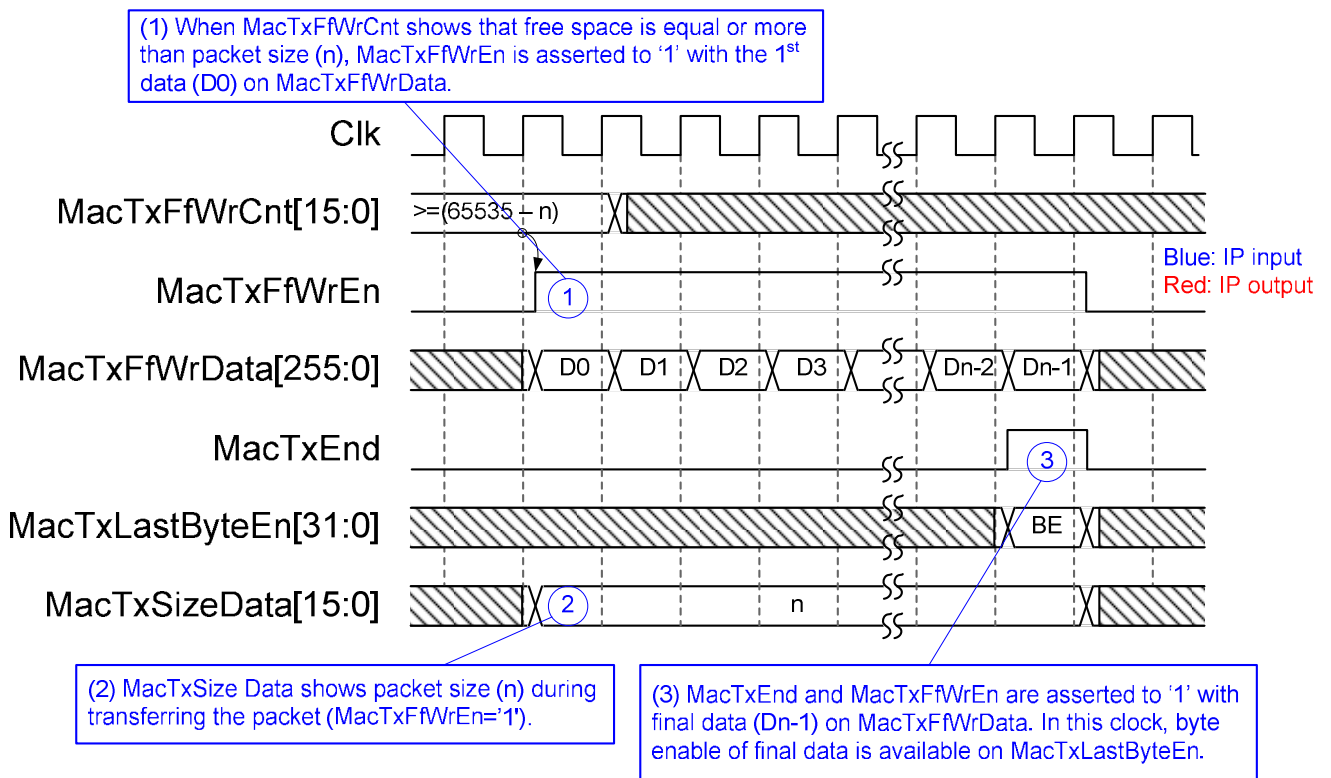


図 12: TOE40G-IP コアと EMAC 間の接続

パケットを送信する前に本 IP コアはまず MacTxFfWrCnt をモニタし 1 パケット分でデータを保持するのに十分な領域を持つ MacTxFfWrCnt 内に十分な空きがあることを確認します。そして 1 パケットの末尾データまで連続して MacTxFfWrCnt をアサートします。その MacTxFfWrCnt='1' と同一のクロック期間で MacTxFfWrData は有効となります。パケットの最終データでは MacTxEnd が有効な MacTxLastByteEn と一緒に '1' アサートします。各パケットの転送中 MacTxSizeData はその値を変化させずパケットの総データ数が 256 ビット単位であることを示します。本 IP コアがパケット・サイズ=n で送信する際のタイミング波形例を図 13 に示します。



- (1) MacTxFfWrCnt がパケット・サイズ(n)と同じかそれ以上の空き空間を示すと先頭データ(D0)と同時に MacTxFfWrCnt が '1' アサートされる
- (2) MacTxSizeData 値はパケット転送中(MacTxFfWrCnt='1')パケット・サイズ(n)を示す
- (3) MacTxFfWrData 上に最終データ(Dn-1)とともに MacTxEnd と MacTxFfWrCnt が '1' アサートされる。このクロック期間では最終データのバイト・イネーブルが MacTxLastByteEn 上で示される

図 13: EMAC Tx FIFO インターフェイスの送信タイミング・チャート



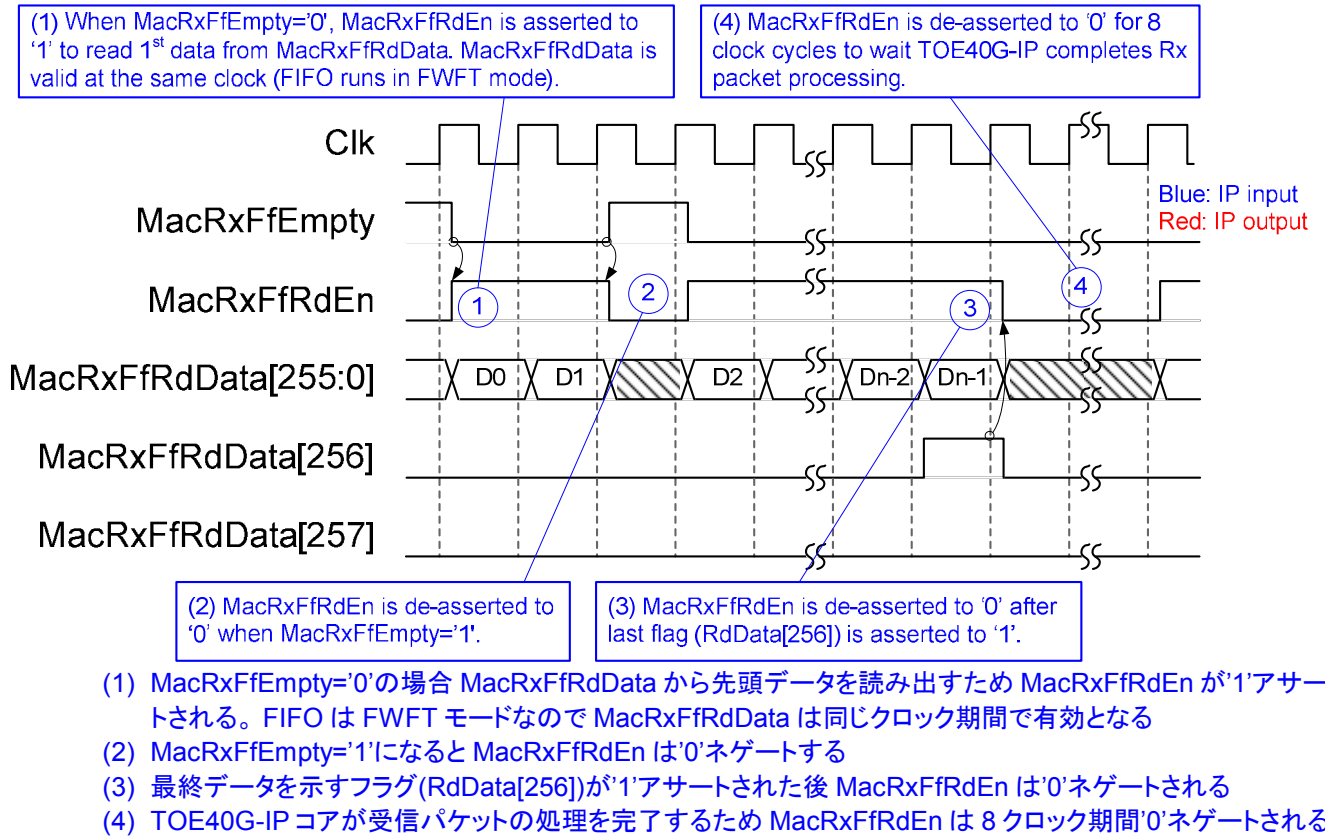


図 14: MACRx FIFO インターフェイスの受信タイミング・チャート

図 14 に受信パケット・サイズが  $n$  である場合の MacRx FIFO インターフェイスのタイミング波形例を示します。本 IP コアは MacRxFfEmpty で MacRx FIFO 内のデータ状態をモニタします。そして MacRxFfEmpty='0' の場合に MacRxFfRdEn='1' として MacRx FIFO からデータを読み出します。FWFT モードの特性により、MacRxFfRdEn='1' とした同じクロック期間で MacRxFfRdData に有効なリード・データが現れます。MacRxFfRdData の bit[256] はパケットの最終データであることを示します。その最終データを受信すると MacRxFfRdEn は 8 クロック期間 MacRxFfRdEn='0' ネゲートし本 IP コアが受信パケットを処理するのを待ちます。その後再び MacRxFfEmpty='0' であった場合は次のパケット処理を開始するため MacRxFfRdEn を '1' アサートします。

上記のタイミング波形から本 IP コアの MacFIFO インターフェイスを 40Gb EMAC の 256 ビット AXI4 ストリーム・インターフェイスに変換する接続ロジックが必要となります。接続ロジックの実装例は本 IP コア同梱のリファレンス・デザインにて VHDL ソースコードで提供されます。

---

## コアの使用例

### クライアント・モードの場合(SRV[0]='0')

データ送信とクライアント・モードでの受信における本コアのレジスタ設定シーケンス例を以下に示します。(より詳細についてはリファレンス・デザインのソースコードを参照してください。)

- 1 RSTレジスタを'1'にセットしIPコアをリセット状態とします。
- 2 MACアドレスをSML/SMHに、IPアドレスをDIP/SIPに、ポート番号をSPN/DPN (DPNは本IPコアによるアクティブ・オープンの場合に設定する必要がある)にセットします。
- 3 RSTレジスタを'0'にクリアしリセット状態を解除します、するとIPコアはARP要求を送信し相手側のMACアドレスを受信したARP応答から抽出して獲得します。初期化が完了するとビジー信号(RegAddrA1[0])は'0'にクリアされます。
- 4 コネクションは以下2つのモードで確立されます。
  - a. アクティブ・オープンの場合: CMDレジスタをセットしポートをオープン(IPコアよりSYNパケットが送信)します。
  - b. パッシブ・オープンの場合: "ConnOn"が'1'となる(通信相手からSYNパケットを受信)のを待ちます。
- 5
  - a. データ送信の場合: TDL/PKLレジスタに総転送長およびパケット・サイズをセットしCMDレジスタをセットしてデータ送信を開始します。ユーザ回路は全転送データを送信FIFOに書き込み、CMDレジスタをモニタしてビジー・フラグが'0'にネゲートされるのを待ちます。次の転送を行う場合ユーザ回路はIPコアをリセット状態にせずにTDL/PKLレジスタを変更しCMDレジスタをセットすることができます。
  - b. データ受信の場合: 受信FIFOのステータスをチェックし受信FIFOが空になるまでデータを読み出します。
- 6 コネクションの確立と同様コネクションは以下2モードでクローズできます。

アクティブ・クローズの場合: CMDレジスタをセットしてポートをクローズ(IPコアよりFINパケットが送信)します。

パッシブ・クローズの場合: "ConnOn"が'0'となる(通信相手からFINパケットを受信)のを待ちます。

### サーバー・モードの場合(SRV[0]='1')

サーバー・モードとクライアント・モードの相違点は初期化プロセスにおける通信相手MACアドレスの取得方法です。クライアント・モードではMACアドレスは本IPコアが送信したARP要求に対して受信したARP応答から抽出します。サーバー・モードではコアに設定されたターゲットIPアドレスと合致したARP要求パケットから抽出します。データの送信・受信プロセスはクライアント・モードと同一です。サーバー・モードでの初期化シーケンスを以下に説明します。

- 1 RSTレジスタを'1'にセットしIPコアをリセット状態とします。
- 2 MACアドレスをSML/SMHに、IPアドレスをDIP/SIPに、ポート番号をSPN/DPNにセットします。
- 3 RSTレジスタを'0'にクリアしリセット状態を解除します、するとIPコアは相手側のMACアドレスを取得するためARP要求の受信を待ちます。そして接続ターゲットに対してARP応答を送信します。初期化が完了するとビジー信号(RegAddrA1[0])は'0'にクリアされます。
- 4 以降のプロセスはクライアント・モードのステップ4~6と同一です。

## コアの検証方法

本 IP コアのユーザ I/F 部についてはシミュレーションによる検証が可能であり、ZCU102/ZCU106 ボードにより実機動作での検証が可能です。

## 必要とされる環境と設計スキルについて

本コアの実機動作確認やデザイン・ゲートウェイ社へのサポート依頼には Xilinx 純正の評価ボード ZCU102 または ZCU106)が必要となるため、ユーザ側でコア購入時に手配してください。

また、本コアを使ってユーザ・システムを設計・実装するためには、HDL 言語設計技術・TCP/IP プロトコル知識および Vivado/ISE/EDK/SDK によるデザイン実装経験を必要とします。

## 注文情報

本製品は Xilinx 代理店から、あるいはデザイン・ゲートウェイ社から直接購入することが可能です。また、デバイス・ファミリに応じて以下のコアのラインナップが用意されています。

コア型番	対応ファミリ	検証用評価ボード	説明
TOE40G-IP-ZUP	Zynq Ultrascale+	ZCU102	Zynq Ultrascale+対応 TOE40G-IP コア

※ 上記以外のデバイス・ファミリ対応状況につきましては以下の問い合わせ URL 経由にてお問い合わせください。

<http://www.dgway.com/Inquiry.html>

## 更新履歴

リビジョン	日付	説明
1.0	Oct-2-2018	New release
1.0J	2018/10/29	日本語初期版の作成