EMIF IP ユーザー・インターフェース補足
- アクセス波形説明

株式会社マクニカ アルティマカンパニー

Rev.1　2020/5

Agenda

- 本資料の位置づけ
- ライトリクエスト波形
- リードリクエスト波形

本資料の位置づけ

- 本資料では Intel® FPGA の EMIF IP のユーザー・インターフェイス信号の波形を説明します
  - Avalon® Interface Specification で Memory-Mapped Interface (Avalon-MM)として定義されたプロトコルが使用され、IP は Slave Component になります
- 既存の資料として下記がありますが、ウェイトがかかる場合の例が少ないため、その部分を補完する資料になります
  - EMIF ハンドブック：
    https://www.intel.com/content/www/us/en/programmable/support/literature/lit-external-memory-interface.html
  - Avalon® Interface Specification (v18.1)
    https://www.intel.com/content/dam/www/programmable/us/en/pdfs/literature/manual/archives/mnl_avalon_spec-18-1.pdf
  - EMIF 説明資料(Macnica 技術情報 Web)
    「EMIF リード/ライト・シーケンスの概要…」
    https://macnicago.zendesk.com/hc/ja/articles/360024575571

本資料の位置づけ（続き）

- 本資料の説明は、ALTMEMPHY、UniPHY および Generation 10 FPGA の EMIF IP に共通して適用できます
  - 本資料では Cyclone® V(UniPHY IP) の信号名を使用しています
  - 信号名の変更や一部信号の廃止など変更は各世代の関連資料をご確認下さい

ライトリクエスト波形

- ライト(リクエスト)波形として、以下の４つの場合について説明します
  (1) avl_size = 1, avl_ready = H (バースト長 1、ウェイトが無い場合)
  (2) avl_size = 1, avl_ready = L (バースト長 1、ウェイトがある場合)
  (3) avl_size ≧ 2, avl_ready = H (バースト長 2 以上、ウェイトが無い場合)
    - バーストの全てのライトデータ転送リクエストで avl_ready=H の場合です
  (4) avl_size ≧ 2, avl_ready = L (バースト長 2 以上、ウェイトがある場合)
    - バースト先頭では avl_ready = H でも途中のリクエストで avl_ready = L の場合を含みます

※ avl_ready = L (リクエスト受付のウェイト) を禁止する事はできないためウェイトが無い場合とある場合の両方に対応する必要があります

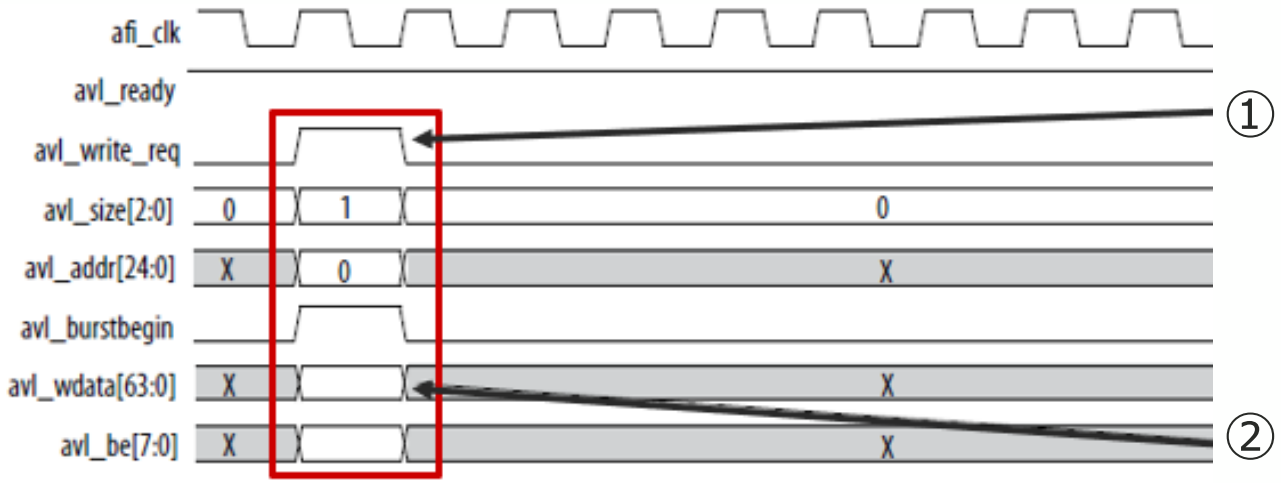
ライト波形(1)：avl_size = 1, avl_ready = H (バースト長 1、ウェイトが無い場合)

- 転送は、 1 サイクル(①,②)で完了します

- ユーザーは、書き込みリクエストとして、①、②の赤枠内の信号を発行します

① avl_write_req とavl_burstbegin をHigh にすると同時にavl_size とavl_addr を指定します (avl_ready = H の時)

② 同時に、書き込みデータとそのバイトイネーブル信号 avl_wdata, avl_be をコントローラーに送信します



ライト波形(1)-(4) 共通：

a. 転送期間終了後、直後を含む任意のサイクルで、ユーザーは次のライト/リード・リクエストを開始することができます
b. リクエストが連続しない(待機)場合は avl_write_req, avl_read_req, avl_burstbegin は Low に駆動し、他のユーザー駆動の信号は任意(don't care) です
c. 転送期間が完了するまで他のライト/リード・リクエストを発行することはできません
d. ２つ以上のライト転送期間が重複することはありません（ライト転送期間にリード転送期間が重なることは許容されます）

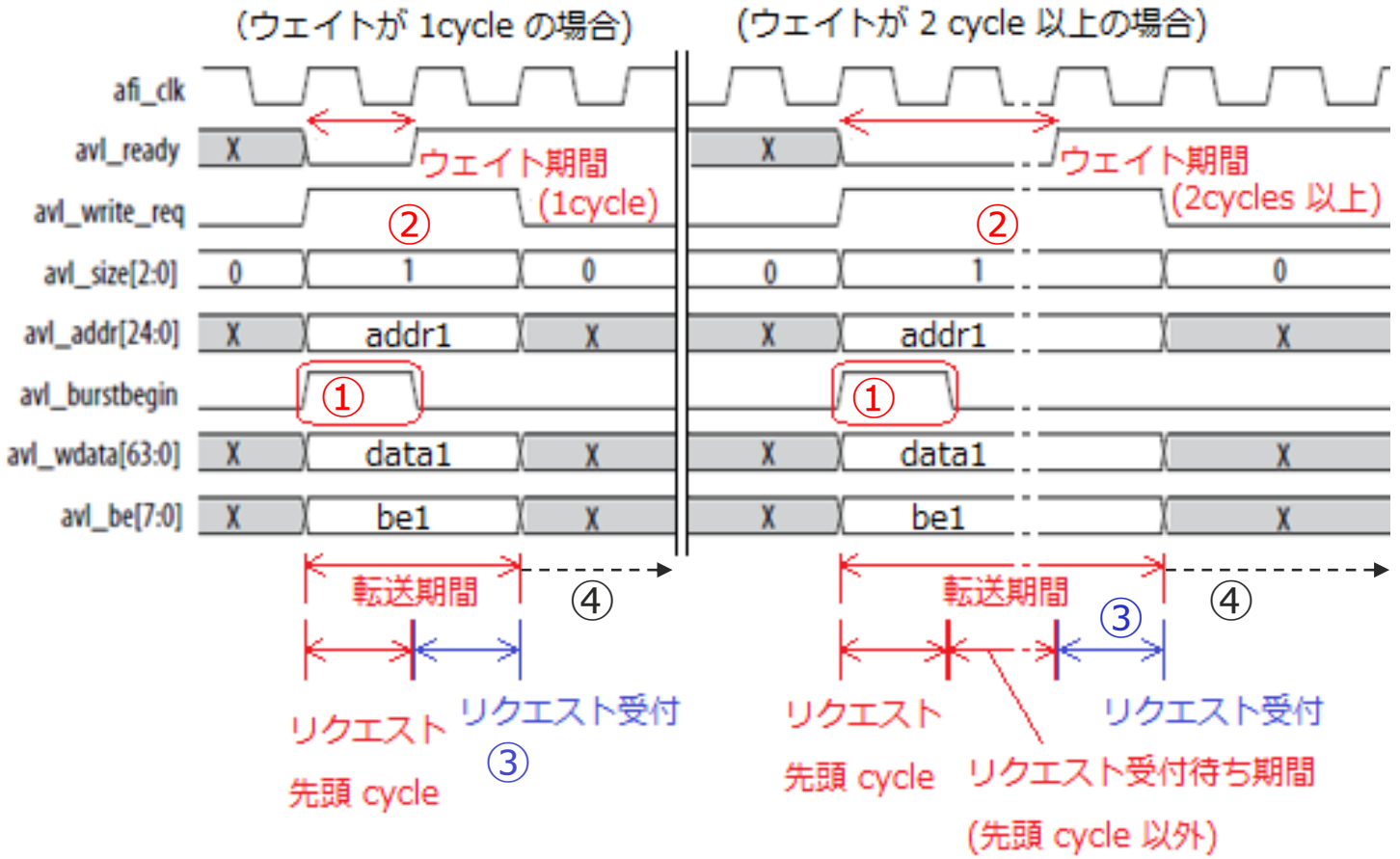
ライト波形(2)：avl_size = 1, avl_ready = L (バースト長 1、ウェイトがある場合)

- 転送期間はウェイト期間だけ増え、2サイクル以上必要となります

① avl_burstbegin = High となるのは
リクエスト先頭 の 1 cycle のみで、
リクエスト中の後の期間は Low となります

② リクエスト信号と、リクエスト内容となる
情報の信号 (avl_write_req, avl_size,
avl_addr, avl_wdata, avl_be) は、
転送期間中維持される必要があります
(size = 1 の場合)

③ avl_ready = Low から初めて High となった
サイクルでリクエストが受付けられ、転送
期間は終了します



(ライト波形(1)-(4) 共通：転送期間終了後、直後を含む任意のサイクルで、ユーザーは次のライト/リード・リクエストを開始することができます (④))

ライト波形(3)：avl_size ≧ 2, avl_ready = H (バースト長 2 以上、ウェイトが無い場合)

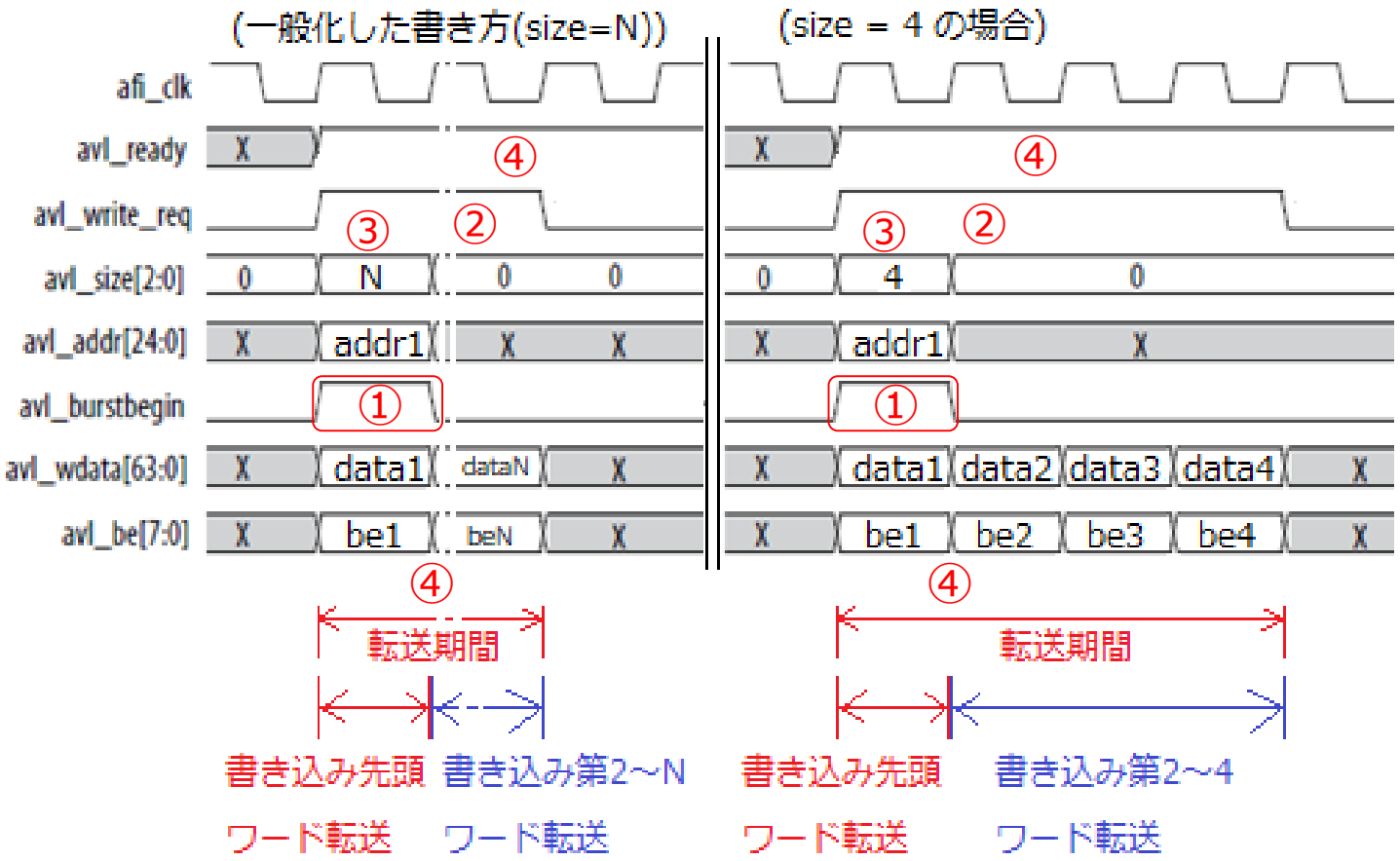
- 転送期間は avl_size で指示されたバースト長(サイズ)となります

① avl_burstbegin = High となるのはリクエスト先頭 の 1 cycle のみで、1 つのリクエスト中で後の期間は Low となります

② ライトバーストでは、リクエスト信号(avl_write_req)は期間中ずっと High となります(転送ワード毎に発行されます)

③ リクエスト内容(サイズ、転送アドレス)と書き込み先頭ワードのデータが最初に転送されます
(avl_size, avl_addr, avl_wdata, avl_be)

④ ウェイト(avl_ready =L)サイクルがないので、それ以降順次第2ワード以降のデータが受け付けられます(avl_wdata, avl_be)
つまり、リクエストの内容はどの情報も必要な 1 サイクルのみ維持すれば十分です



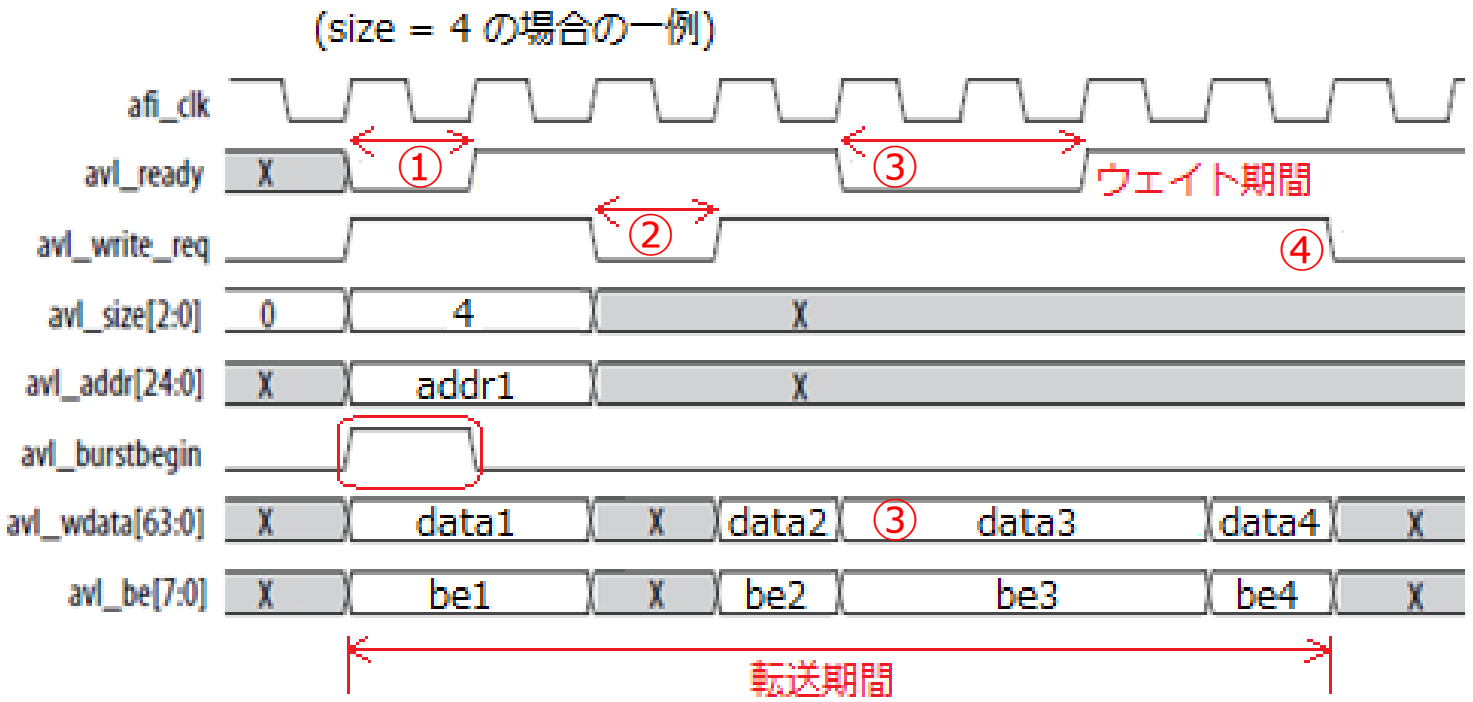
ライト波形(4)：avl_size ≧ 2, avl_ready = L (バースト長 2 以上、ウェイトがある場合)

- 転送期間は(3)の場合(サイズ)から、挿入されたウェイトだけ増えます
  - (2)(3) の場合の波形を組み合せ、バースト化とウェイトの挿入を行った形です

① 先頭ワードの書き込み転送で avl_ready=L によりウェイトがある場合は、図の様に先頭ワードの波形は (2) と同様です
(先頭ワードで avl_ready=H によりウェイトがない場合はその部分の波形は (1)(3) と同様です)

② avl_write_req=L の期間を挿入してユーザー側でウェイトをかけることができます
(データ信号は don't care)

③ 2ワード目以降の転送は、ウェイトが無い転送ワードは (3) と同様ですが、ウェイトがある転送ワードでは avl_write_req = H とデータ信号(avl_wdata, avl_be) の値は (2) 同様保持される必要があります



(ライト波形(1)-(4) 共通：転送期間が完了するまで他のライト/リード・リクエストを発行することはできません(④))

リードリクエスト波形

- リード(リクエスト)波形として、以下の４つの場合について説明します
  - リードの場合、リードリクエスト開始時の avl_ready の値がウェイトの有無に対応します

(1) avl_size = 1, avl_ready = H (バースト長 1、ウェイトが無い場合)
(2) avl_size = 1, avl_ready = L (バースト長 1、ウェイトがある場合)
(3) avl_size ≧ 2, avl_ready = H (バースト長 2 以上、ウェイトが無い場合)
(4) avl_size ≧ 2, avl_ready = L (バースト長 2 以上、ウェイトがある場合)

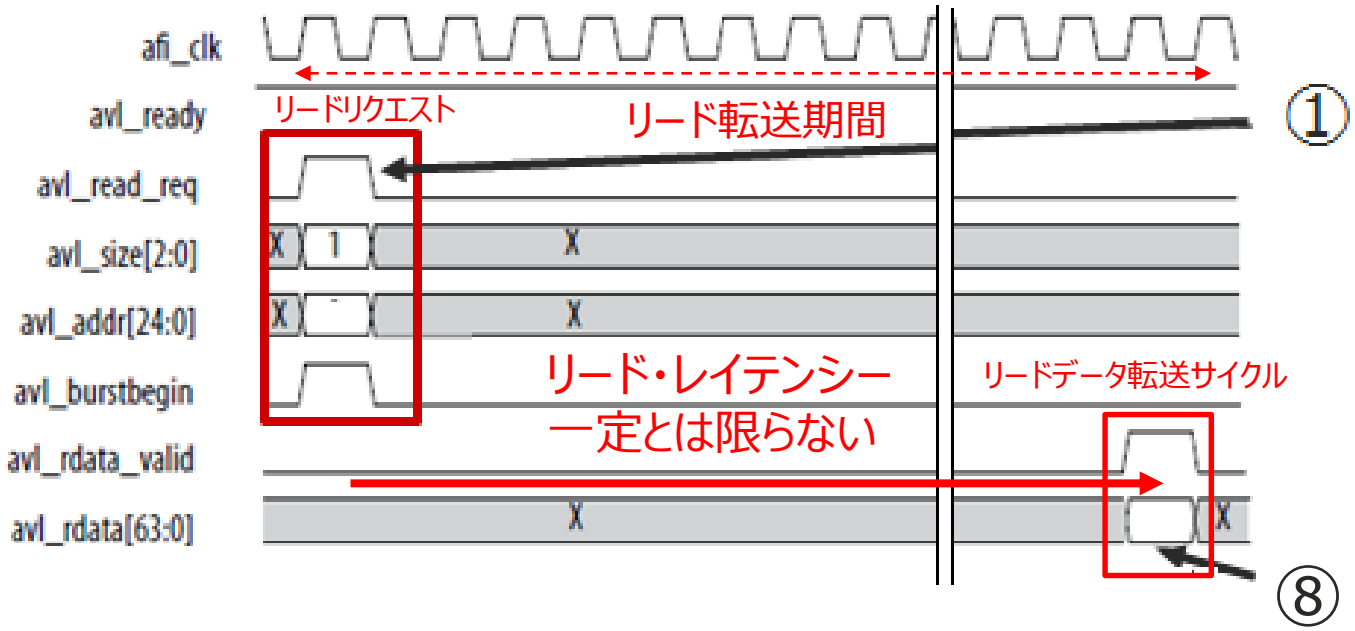
※ avl_ready = L (リクエスト受付のウェイト) を禁止する事はできないため
ウェイトが無い場合とある場合の両方に対応する必要があります。

リード波形(1)：avl_size = 1, avl_ready = H (バースト長 1、ウェイトが無い場合)

- リードリクエストは 1 サイクル(①)で完了、転送期間は読み出しデータが 1 ワード転送されれば終了します

① リードリクエスト信号の発行：
avl_read_req とavl_burstbegin を High にすると同時に avl_size = 1 と avl_addr を指定します
(avl_ready = H 、1サイクルで完了)

⑧ リードデータ信号によるリードデータ転送サイクル：
リード・レイテンシー経過後、コントローラーより avl_rdata_valid = H が出力されると同時に、読み出しデータが avl_rdata に出力されます
（リード・レイテンシーは一定ではない事に注意してください：次スライド共通事項）

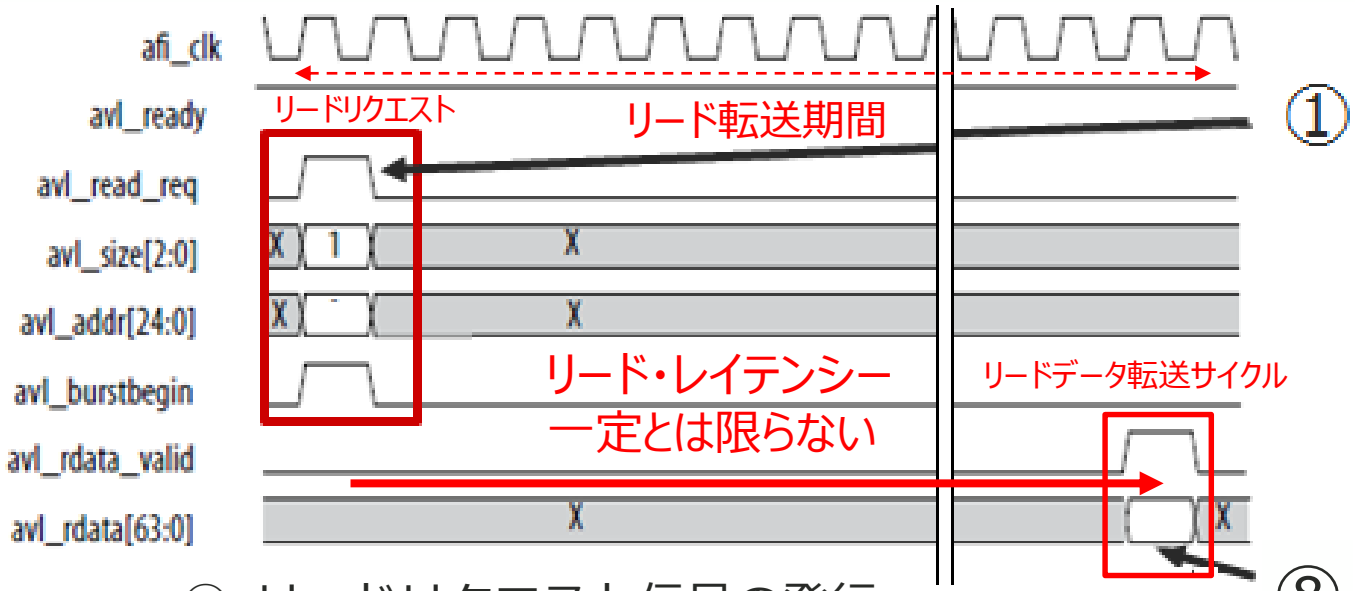


次スライドのリード波形(1)-(4) 共通事項にも注意してください

リード波形(1)-(4) 共通事項

- 下記はリード波形(1)-(4) の共通事項です（図はリード波形(1)）

a. リードリクエスト発行(開始)から、リードデータ転送サイクル完了までをリード転送期間とします
b. リードリクエスト発行(完了)後、直後を含む任意のサイクルで、ユーザーは次のライト/リード・リクエストを開始することができます。
転送期間の終了を待つ必要はありません。
c. リード転送期間は、リクエスト信号の発行を除いて、ライトを含む他の転送期間と重複することができます（ただし、リードデータ転送サイクルが重複することはありません）
d. リクエストが連続しない(待機)場合は avl_write_req, avl_read_req, avl_burstbegin は Low に駆動し、他のユーザー駆動の信号は任意(don't care) です



① リードリクエスト信号の発行
⑧ リードデータ信号によるリードデータ転送サイクル

e. リード・レイテンシーはいくつかの要因で増大することがあります（固定値ではありません）。そのためリードリクエストを連続で受け付けてもリードデータ転送サイクル (avl_rdata_valid = H) が同様に連続するとは限りません
f. リードデータ転送サイクルの読み出しデータの到来順は必ずリードリクエストの順になります（バースト中はアドレス順）
g. バースト長が 2 以上の場合、リードデータ転送サイクルがそれだけ順に行われますが、中断する場合があります

リード波形(2)：avl_size = 1, avl_ready = L (バースト長 1、ウェイトがある場合)

- リクエストの受付がウェイト期間だけ遅れ(②,③)、転送期間は読み出しデータが 1 ワード転送されれば終了します(④)

リードリクエスト信号の発行(①～③)：

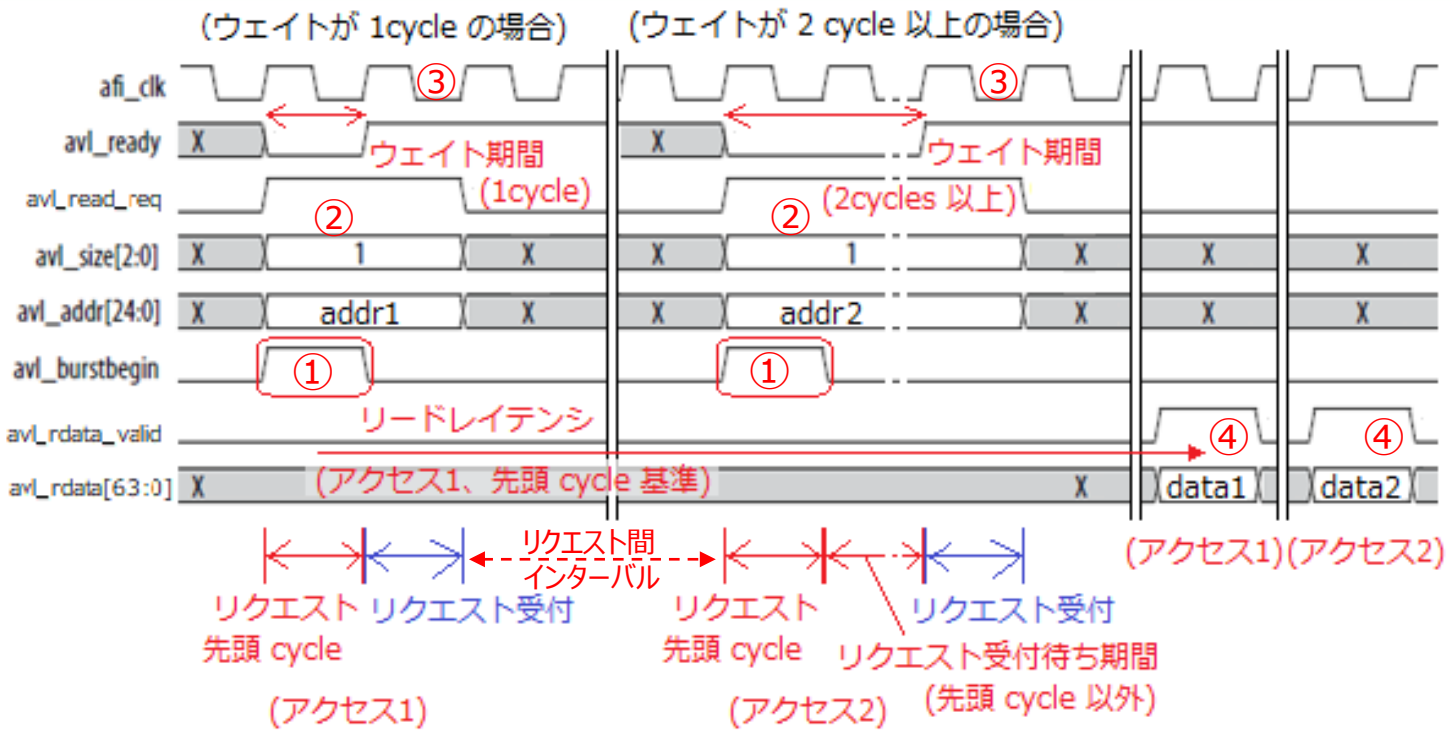
① avl_burstbegin = High となるのは
リクエスト先頭 の 1 cycle のみで
後はリクエスト受付まで Low となります

② リードリクエスト信号 (avl_read_req=H) と
リクエスト内容となる情報の信号
(avl_size=1, avl_addr) は、リクエスト
受付まで維持される必要があります。

③ avl_ready = L から初めて High となった
サイクルでリクエストが受付けられます

リードデータ転送サイクル：

④ リード・レイテンシー経過後、コントローラー
より avl_rdata_valid = H が出力される
と同時に、読み出しデータが avl_rdata に
出力されます



※ 図は2つのリードリ転送期間(アクセス 1, アクセス2)を重複して描いています

スライド12 の共通事項にも注意してください

リード波形(3)：avl_size ≧ 2, avl_ready = H (バースト長 2 以上、ウェイトが無い場合)

- リードリクエストは 1 サイクル(①)で完了、転送期間は読み出しデータが avl_size で指定されたサイズ(N)分転送されれば終了します(③)

<u>リードリクエスト信号の発行：</u>

① avl_read_req とavl_burstbegin をHigh にすると同時に avl_size=N とavl_addr (バースト先頭アドレス)を指定します (avl_ready = H 、1サイクルで完了)

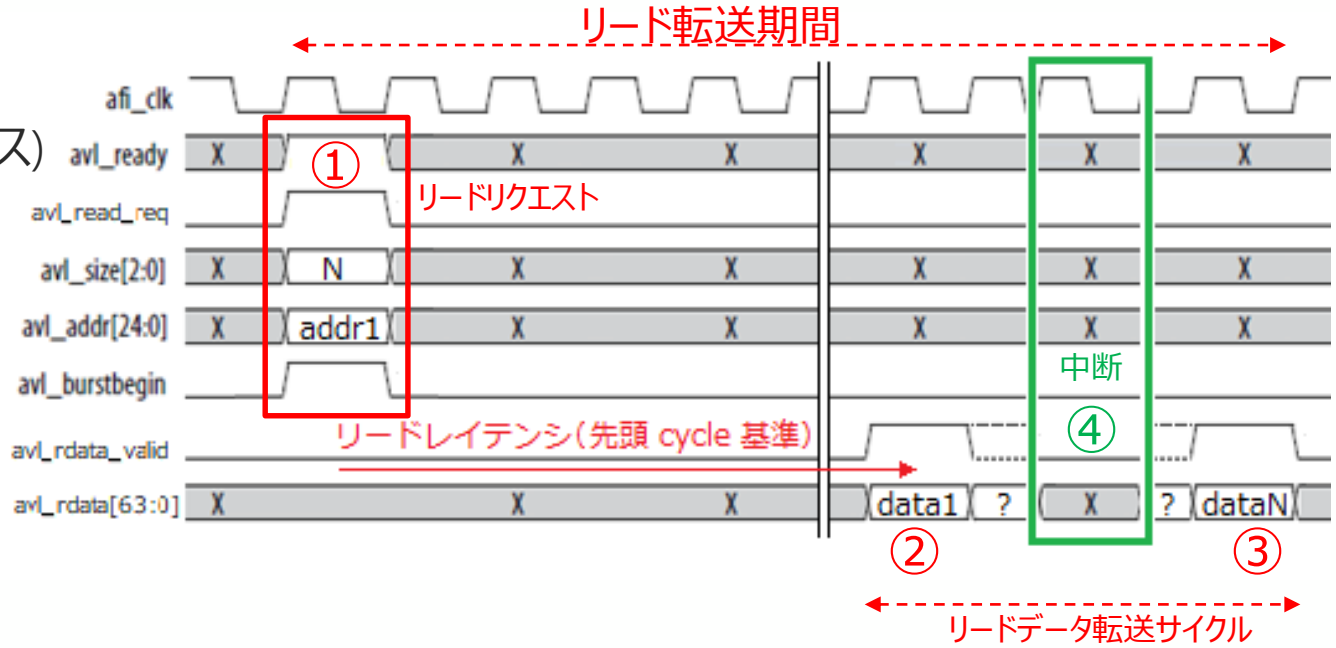
<u>リードデータ転送サイクル(②～③, ④)：</u>

② リード・レイテンシー経過後、コントローラーより avl_rdata_valid = H が出力され、同時にバーストの先頭の読み出しデータが avl_rdata に出力されます（以後、中断以外は avl_rdata_valid = H）

③ バーストサイズとして指定した数(N)、順次読み出しデータが転送されて、転送期間が終了します

④ リードデータ転送が中断する場合があり、その場合中断期間は avl_rdata_valid = L となります（avl_rdata は don't care）

スライド12 の共通事項にも注意してください



（リード波形(1)-(4) 共通：リード・レイテンシーはいくつかの要因で増大することがあります（固定値ではありません） ）

リードデータ転送サイクルの中断は、データ転送のウェイトととらえることができますが、混乱を避けるため中断と表現しています

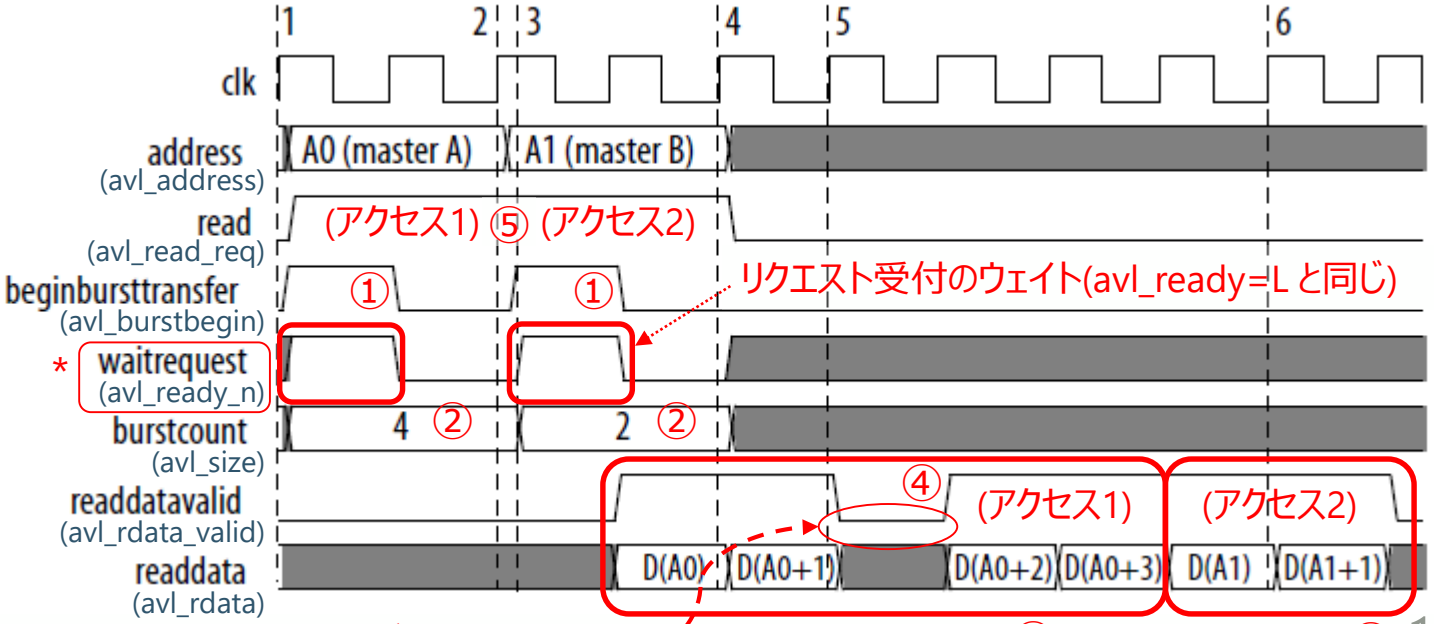
リード波形(4)：avl_size ≧2, avl_ready = L (バースト長 2 以上、ウェイトがある場合)

- リクエストの受付がウェイト期間だけ遅れ(②)、転送期間は読み出しデータが avl_size で指定されたサイズ(N)分転送されれば終了します(③)

リードリクエスト信号の発行(①,②)（リード波形 (2) と同様）：
① リクエスト先頭 1 cycle のみ avl_burstbegin = High となり、後はリクエスト受付まで Low となります
② avl_read_req=H, avl_size=N, avl_addr(バースト先頭アドレス) は、avl_ready = H で受付けられるまで維持されます
リードデータ転送サイクル(③, ④)（リード波形 (3) と同様）：
③ N サイクルの間、avl_rdata_valid =H となり順次読み出しデータが avl_rdata に出力されます
④ 中断する場合があり、その場合、中断期間は avl_rdata_valid = L となります（avl_rdata は don't care (任意)）

スライド12 の共通事項にも注意してください

リード波形の例 (Size = 4, 2 の 2 リクエスト）→
(Avalon Interface Specification より)
(IP の信号名は ( ) 内に青色で併記しています)
* waitreqiest は avl_ready の反転に相当します
アクセス1(size=4), アクセス2(size=2) が連続してリクエストされた場合(⑤) の図となっています
なお、一般的なバーストリードの例のため、
レイテンシーは比較的小さく描かれています
DDRx EMIF の場合この図よりレイテンシーは大きくなります



Appendix

- Avalon® Interface Specification の信号名との照応
- ウェイトがかかる要因の例

Avalon® Interface Specification の信号名との照応

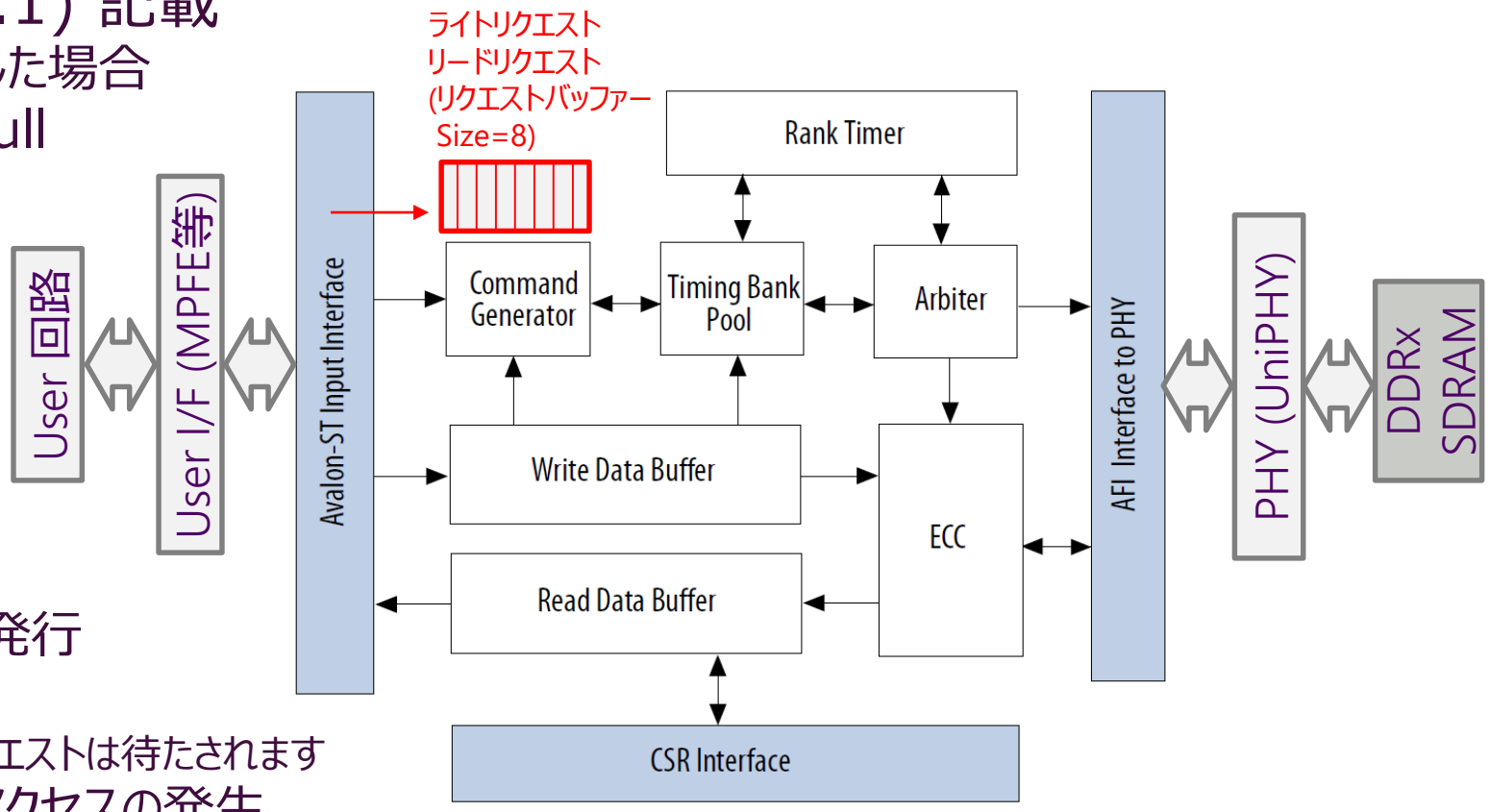
- Avalon® Interface Specification (資料) 掲載の信号名と EMIF IP (IP Catalog より生成) の信号名の対照表

EMIIF IP	Avalon Interface	備考
avl_ready	(waitrequest_n)	レディ信号
(avl_ready_n)	waitrequest	ウェイト信号(レディ信号の反転)
avl_write_req	write	ライトリクエスト信号
avl_read_req	read	リードリクエスト信号
avl_size	burstcount	バースト長(サイズ、カウント)
avl_addr	address	アドレス(スレーブ)
avl_burstbegin	beginbursttransfer	バーストの最初を示す信号
avl_wdata	writedata	書き込みデータ
avl_be	byteenable	バイト・イネーブル信号
avl_rdata_valid	readdatavalid	読み出しデータの有効を示す信号
avl_rdata	readdata	読み出しデータ

(参考) IP 本体の HDL ファイルでは信号名に Avalon I/F の名前もコメントで付記されています

ウェイトがかかる要因の例

- avl_ready が Low になる(新しいリクエストを受け付けない)要因の例
  - リード・レイテンシーの増大や、リードデータ転送の中断の原因になる場合もあります
  - EMIF Handbook (v17.1) 記載
    - リクエストを 8 つバッファーした場合
    - Timing bank pool が full
    - Read FIFO が full
    - Write FIFO が full
    - Read-modify-write (ECC) の処理待ち
  - 上記以外、または重なると上記につながる要因
    - リクエストを多数発行
    - サイズの大きなリクエストを発行
    - リフレッシュの発生
      優先的に処理されるためリクエストは待たされます
    - Raw アドレスを更新するアクセスの発生
      PCH, ACT 命令が挿入され時間がかかります



メモリー・コントローラー(コア, HPCII)
のブロック図 (UniPHYと連携)

Thank you !

MACNICA

弊社より資料を入手されたお客様におかれましては、下記の使用上の注意を一読いただいた上でご使用ください。

1. 本資料は非売品です。許可なく転売することや無断複製することを禁じます。
2. 本資料は予告なく変更することがあります。
3. 本資料の作成には万全を期していますが、万一ご不明な点や誤り、記載漏れなどお気づきの点がありましたら、弊社までご一報いただければ幸いです。
4. 本資料で取り扱っている回路、技術、プログラムに関して運用した結果の影響については、責任を負いかねますのであらかじめご了承ください。
5. 本資料は製品を利用する際の補助的な資料です。製品をご使用になる場合は、英語版の資料もあわせてご利用ください。

改版履歴

Revision	年月	概要
1	2020年5月	新規作成