Block diagram – 13.12

להלן עיקרי הדברים שעלו מהמפגש עם בארי היום (13.12):

1. ראשית, הצלחנו להריץ את המערכת במלואה (mds\_top\_tb), על בסיס קבצי הפעלה של בארי.

מצוין. התקדמות חשובה. מזכיר כי עליכם להצליח לבצע סימולציות של:

1. כתיבת image לזיכרון החיצוני.
2. כתיבת רגיסטר לבלוק מסוים.
3. בקשת קריאה (והחזרת מידע בהתאם ל- host...) של מידע מהזיכרון החיצוני (ל- DEBUG).
4. בקשת קריאה (והחזרת מידע בהתאם ל- host...) של מידע מרגיסטר בבלוק מסוים (ל- DEBUG).

מטלות א' ו- ב' הן must לפני התחלת הקידוד שלכם, ומטלות ג' ו- ד' אפשרי שיבוצעו בהמשך.

1. הוצע לעבוד עם ה-SDRAM על 4 כל ארבעת הבנקים, לפי החלוקה הבאה:  
   בנק א'+ב': אחסון תמונות מקור (מיושרות).

בנק ג'+ד': לתוכם תכתב תמונה לאחר עיבוד.

מטרת הצעה זו הינה פישוט תהליך הכתיבה והקריאה מהזכרון- בשיטה זו ישנן כתובות/אזורים קבועים לכל פעולה.

האם הרעיון ראוי, וכדאי להמשיך בגישה זו?

זה מתאים רק לעבודה עם התמונה הראשונה. לא יהיה ניתן לקבל עוד תמונות לאחר-מכן, מכיוון שבזמן שלאחר שאחסנתם את התמונה השנייה, אסור לדרוס באמצעות כתיבת התמונה המסובבת הראשונה, את התמונה המסובבת הנוכחית, שכרגע, יוצאת לצג.

צריך פשוט לחדד:

בנק א' – תמונת מקור 1 מוכנסת.

בנק ב' – תמונה מקור 2 מוכנסת.

בנק ג' – תמונת מסובבת של תמונת מקור 1 נכתבת. ברגע, שמסתיימת הכתיבה, ניתן להוציא מבנק ג' את התמונה המסובבת ולהציג לצג.

בנק ד' – תמונת מסובבת של תמונת מקור 2 נכתבת. ברגע, שמסתיימת הכתיבה, ניתן להוציא מבנק ד' את התמונה המסובבת ולהציג לצג.

כאשר 1 ו- 2 זה אינדקסים מתגלגלים.

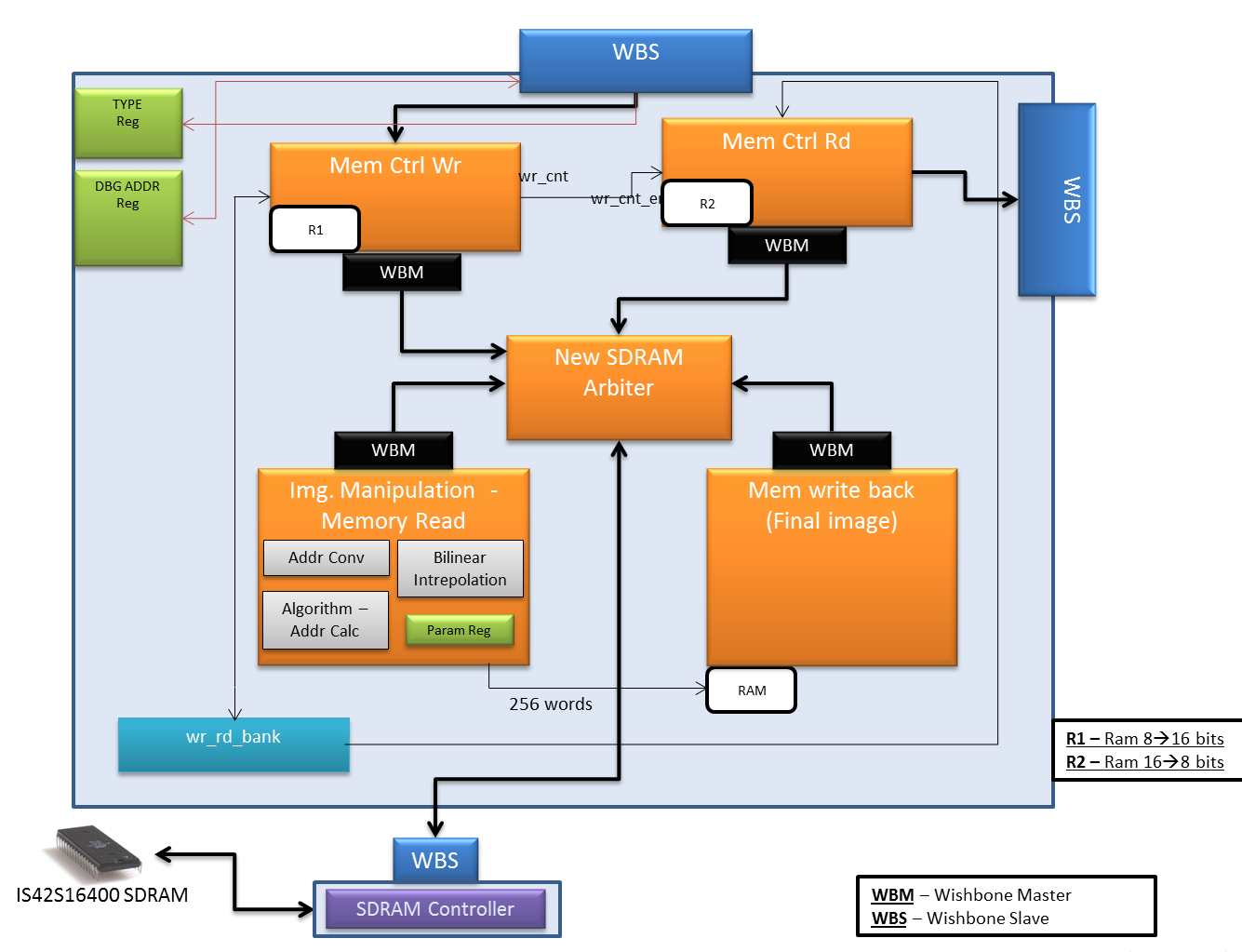
1. בארי הציע שבשלב הבא נלמד את פרוטוקול ה-WISHBONE ונתרגל כתיבה וקריאה ל-SDRAM (על בסיס שני הקבצים sdram\_controller ו-sdram\_model).

בשילוב עם מה שמוצע בסעיף 1.

נציין כי קראנו את ההערות שרשמת לנו בסופ"ש. לחלקן התייחסנו במהלך הפגישה, וחלקן פחות רלוונטיות לאור החלטת 4 הבנקים.

אולי לא הבנתם, אבל 4 בנקים התכוונו (התכוונתי) שיהיו. זה שאמרנו שנחלק ל- 3 חלקים, זה לא אומר 3 בנקים, אלא הכוונה ל- 3 חלקים פונקציונאליים (אחסון תמונת מקור, אחסון סיבוב תמונה, קריאה לצג).

דיאגרמת הבלוקים+הסברים:



* Mem Ctrl Wr - יחידת הבקרה מהפרויקט של בארי, תפקידה לכתוב לSDRAM- תמונה ישרה במלואה. **נותרת ללא שינוי**.
* Mem Ctrl Rd - יחידת הבקרה מהפרויקט של בארי, תפקידה לקרוא מהSDRAM- תמונה מוכנה (לאחר מניפולציות) להוצאה ל-VESA. **נותרת ללא שינוי**.
* New SDRAM Arbiter – על בסיס הארביטר הקיים, ישונה אלגוריתם התזמון והתעדוף של פעולות הקריאה והכתיבה.
* Mem Write Back (final image) – יחידה שתפקידה לכתוב תמונה מסובבת לתוך ה-SDRAM. היחידה כותבת 256 "מילים" (512 Byte) מתוך ה-RAM אל ה-SDRAM.  
  תמומש לפי פרוטוקול wishbone כ-master ליחידה הארביטר.
* Image Manipularion – Memory Read – יחידה בעלת 4 תתי יחידות :
  + Address Converter - היחידה ממירה כתובת מכתיב מטריצי [I,j] לכתובת זכרון SDRAM.
  + Bilinear Interpolation - היחידה מחשבת את רמת האפור של פיקסל בודד בתמונת היעד בעזרת אינטרפולציה בילינארית (בין 4 פיקסלים).
  + Algorithm – Address Calculator – היחידה מחשבת כתובת "מתמטית" של 4 הפיקסלים שמתאימים לתא הנתון בתמונת היעד (כתובת "מתמטית" - אינדקס [I,j] במטריצת התמונה).
  + Parameters Register – רגיסטר שמאחסן את פרמטרי המשתמש:  
    זוית, זום, תחום עניין. (ייתכן שלרגיסטר תכתבנה הזויות לאחר חישוב טריגונומטרי)

הערה : בלוק הwr\_rd\_bank- ייערך מחדש לצורך ניהול 4 הבנקים.

מצוין. יפה מאוד.

אורי ורן.