14.03.2013 - Integration Status

**הסבר מתומצת של האינטגרציה**

* קוד מטלב שיוצר קובץ UART לטעינת הסמלים לזיכרון ה-SDRAM.
* יכולת כתיבה מ-Display Controller לרגיסטרים של Memory Management (type ו-address debug) והתאמת הקבצים של בארי לצורך כך.
* בלוק SG\_WBM\_IF שמתחבר ל- Symbol Generator Topויוזם קריאות ב-debug mode מה-SDRAM.

**הסבר מפורט של האינטגרציה**

הוספנו יכולת של כתיבה בממשק WB שבין Display Controller ל- Memory Management (ממשק Read WBS), דרך Intercon Y. לשם כך הוספנו את הסיגנלים WE ו- DAT\_O מתוך ה- Display Controller(מוצא), ל- Intercon Y (כניסה ומוצא) ול- Memory Management (כניסה).  
המטרה היא יכולת של כתיבה לרגיסטר type ו-address debug מתוך Display Controller. לשם כך, הרחבנו את הקוד ב- Memory Management, כך שניתן לכתוב לרגיסטרים אלה כעת גם מתוך ממשק ה-Write WBS וגם מתוך ה-Read WBS.

בבלוק של mem\_ctrl\_rd, שבתוך Memory Management, חיווטנו את הסיגנל rd\_wbs\_we\_i, ויצרנו תלות של הבלוק בסיגנל זה, כך שיפעל רק במקרה של קריאה כאשר rd\_wbs\_we\_i='0'. הסיבה היא שכעת ניתן לכתוב מתוך ממשק Read WBS, ולכן צריך היה שהבלוק הזה לא יגיב במקרה של כתיבה לרגיסטרים.  
בנוסף, במכונת המצבים, בתוך מצב wbs\_wait\_ram\_rdy\_st: במקרה שבו cyc יורד, אז מסיימים את מחזור הקריאה וחוזרים ל-idle. הסיבה היא שלא ניתקע במצב זה, כאשר קוראים מה-SDRAM עוד לפני שהיו לשם כתיבות.

בבלוק mem\_ctrl\_wr, ביטלנו את האופציה לכתיבה ב- debug mode. הסיבה לכך היא שבפרויקט שלנו, אנחנו כותבים ב-mode רגיל וקוראים רק ב- debug mode. וכאשר יש ערבוב בין ה-modes, אז קיבלנו בלגן ורצינו להימנע ממנו. סיטואציה בעייתית לדוגמה הייתה כאשר התחילה כתיבה ב-mode רגיל, המשיכה קריאה ב-debug mode ואז חזרה לכתיבה ב-debug mode, למרות שרוצים כתיבה ב-mode רגיל...

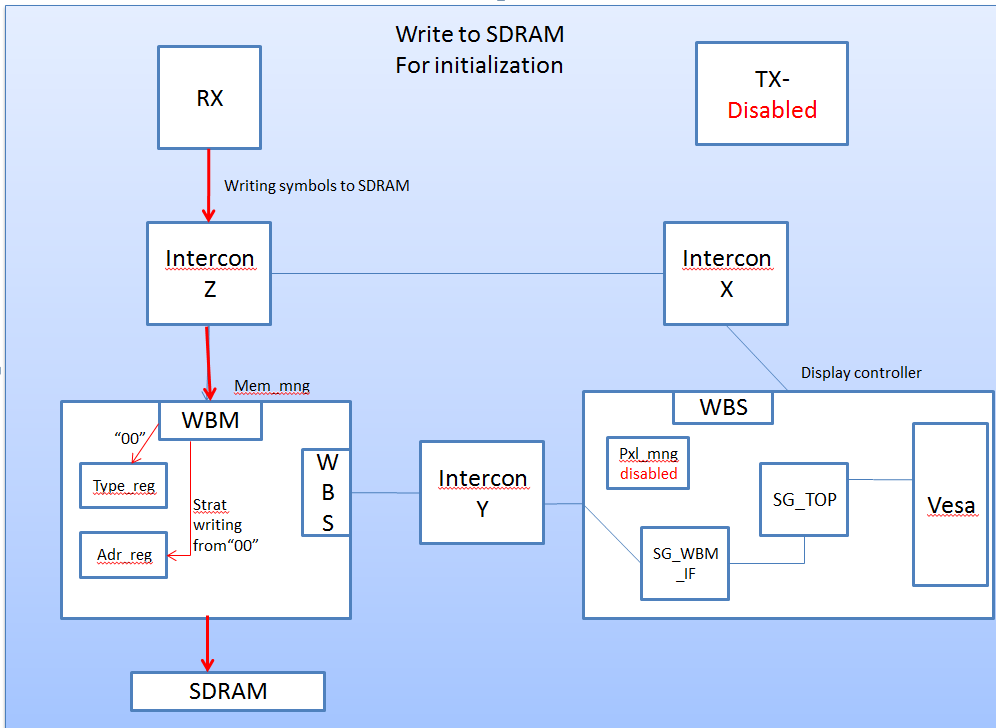
ב- Display Controller:

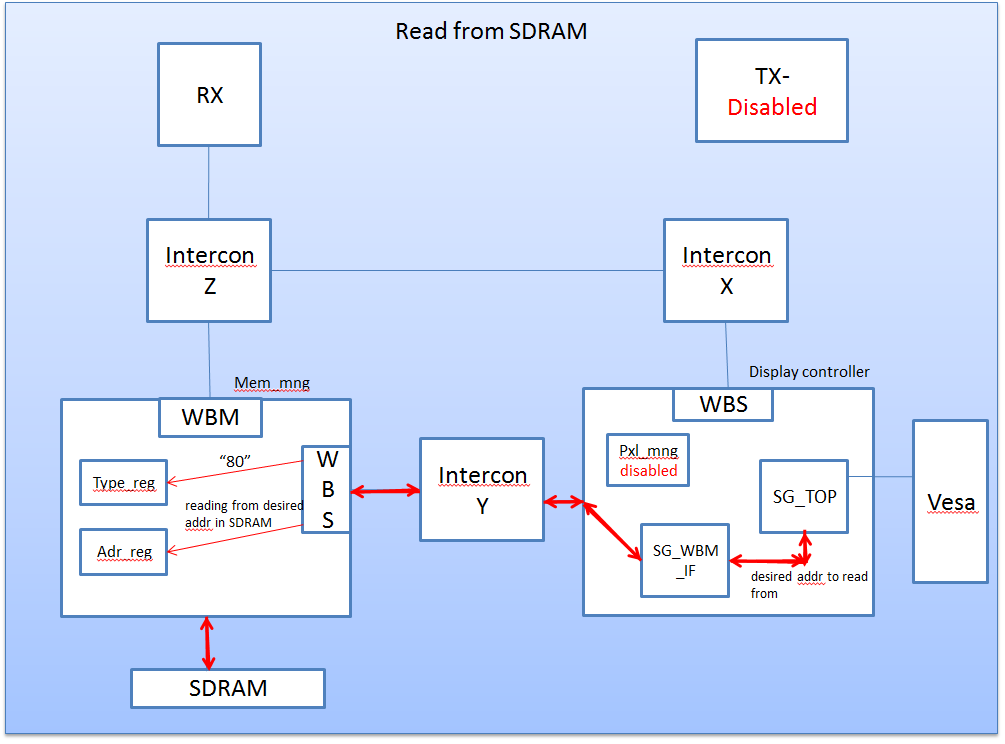
ניתקנו את ה-pixel manager, ובמקומו הוספנו את הבלוק שלנו SG\_WBM\_IF, שתפקידו הוא להתחבר ל- Symbol Generator Top ולהוות ממשק WBM החוצה מה- Display Controller. בלוק זה מקבל בקשות קריאה ב-SDRAM מה- Symbol Generator Top, מתרגם אותן לפרוטוקול WB ויוזם טרנזקציה החוצה, מקבל את ה-data מה-SDRAM ומעביר אותו חזרה אלינו ל- Symbol Generator Top.

אופן פעולת SG\_WBM\_IF (מימוש ע"י FSM):

1. כתיבת לרגיסטר debug address (ב- Memory Management) את כתובת הקריאה הרצויה מה-SDRAM.
2. כתיבה לרגיסטר type (ב- Memory Management) את מצב ה-debug באמצעות ביט ה-LSB ברגיסטר.
3. טרנזקציית קריאה על WB, מ- Display Controller ל- Memory Management ול-SDRAM, שמתורגמת לקריאה ב-debug mode.

**Top Block Diagram**





**Display Controller Block Diagram**

