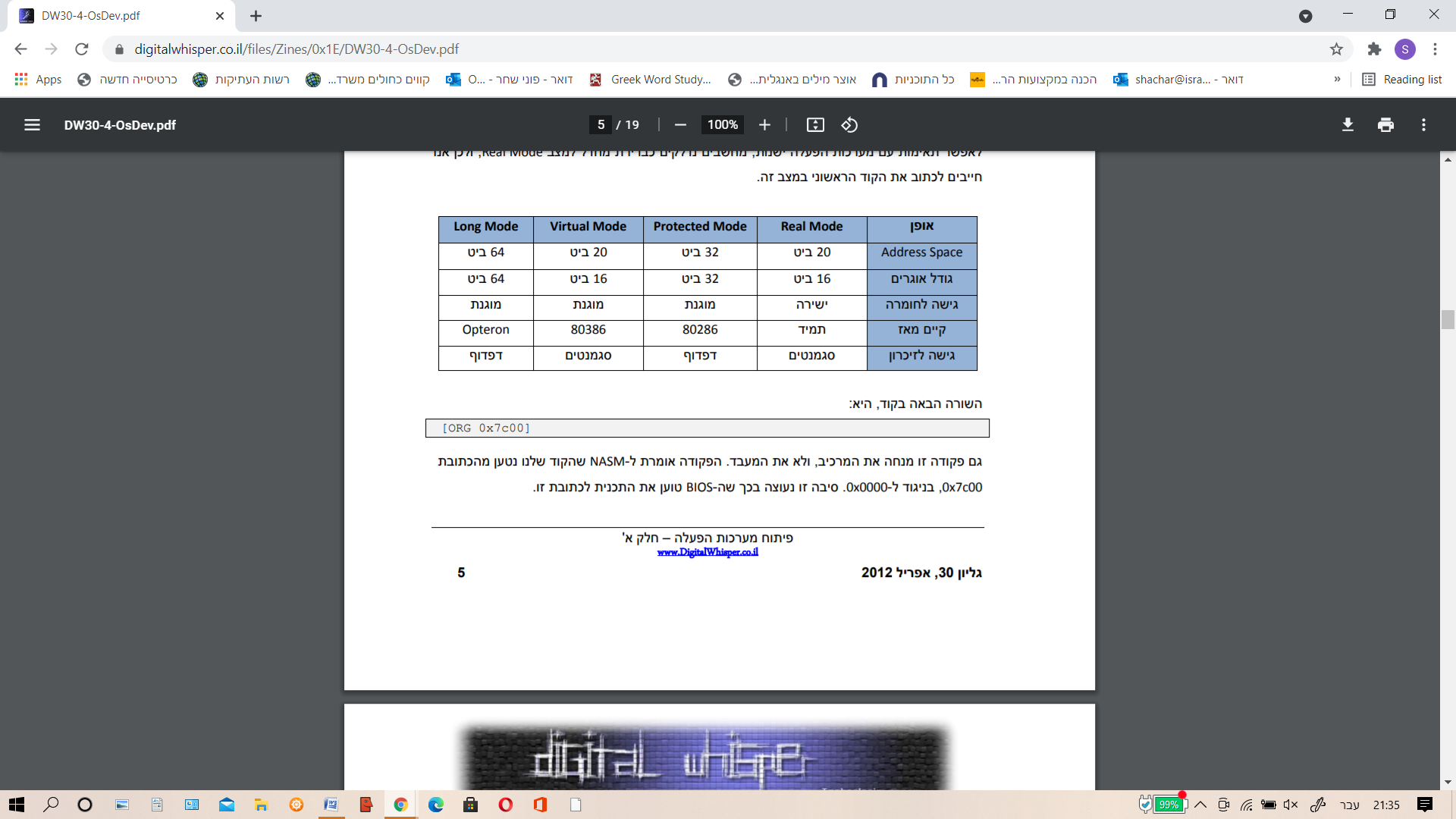
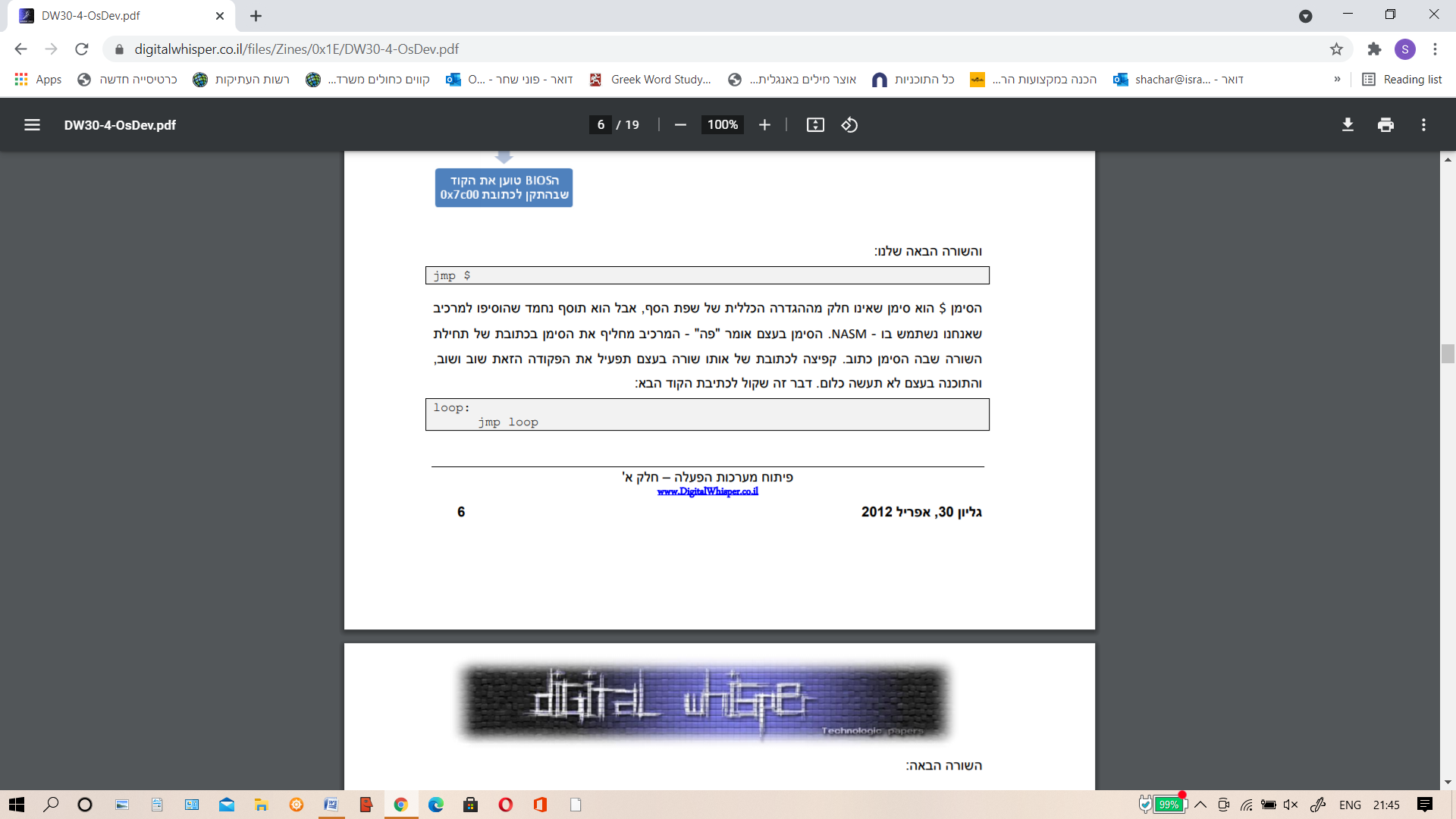
**Boot.asm**



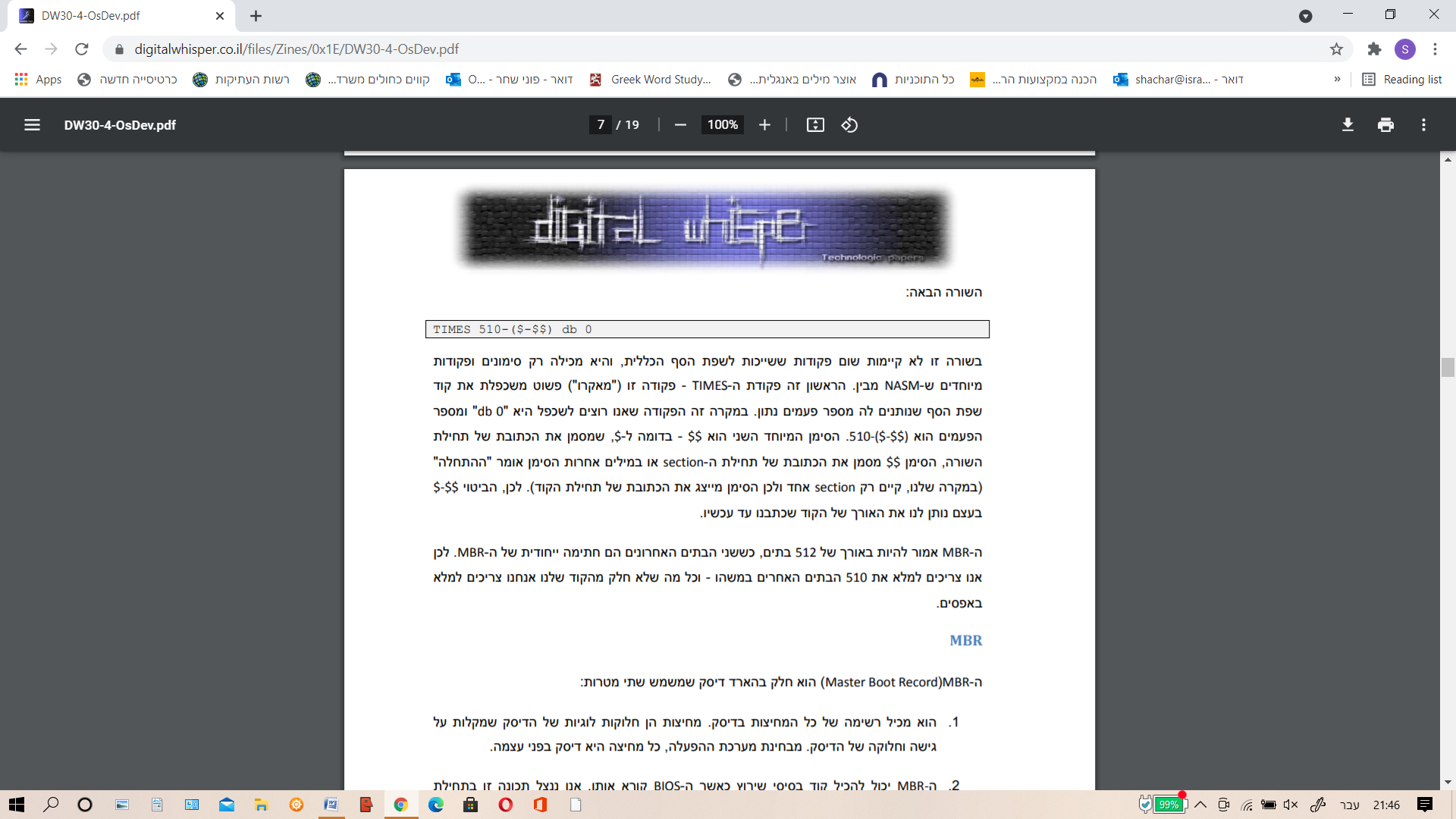
כשאנו מתחילים את הקוד בהוראה [16 BITS [אנו מנחים את המרכיב לתרגם את הקוד לארכיטקטורת 16 ביט ב-real mode



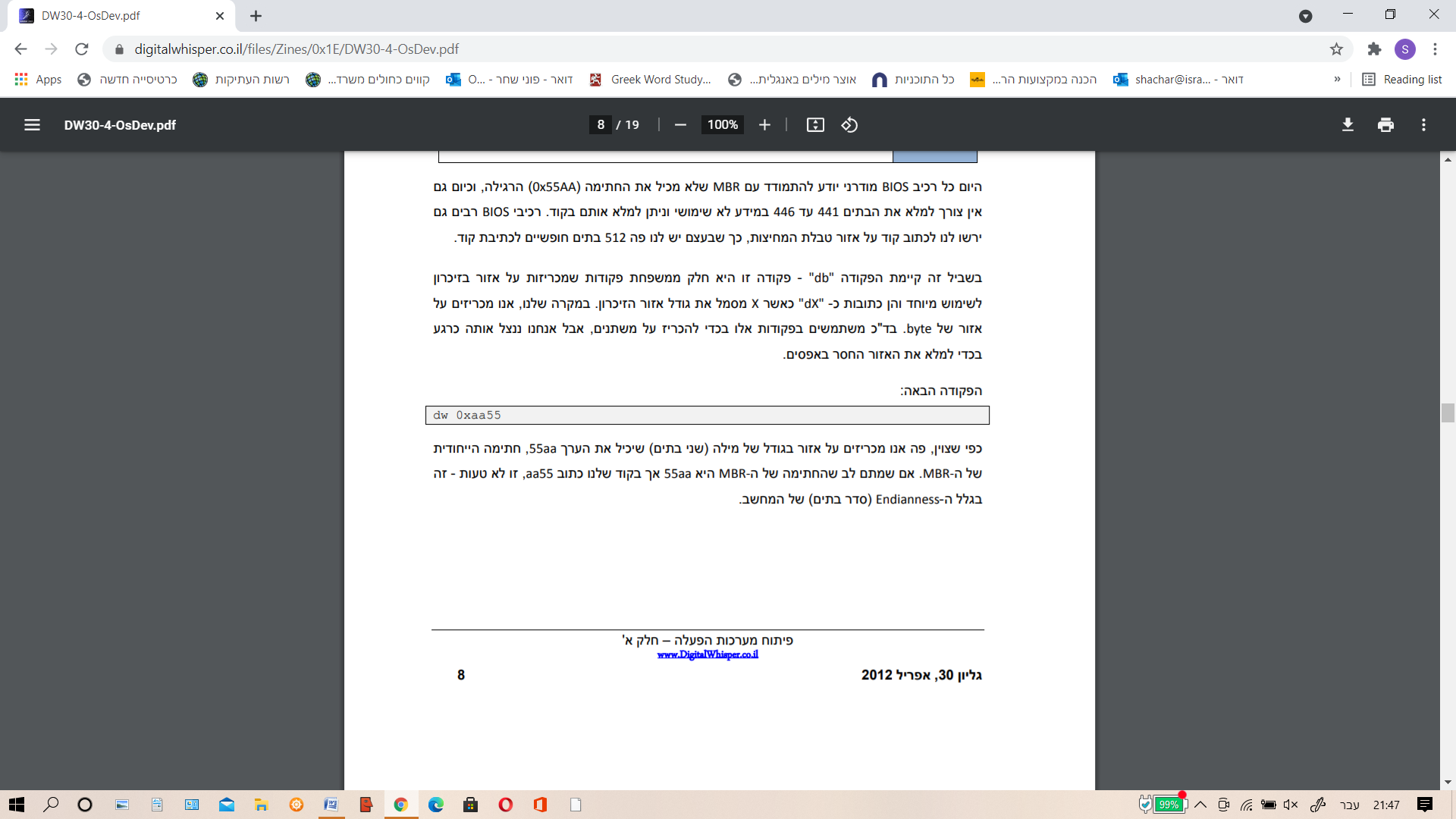
גם פקודה זו מנחה את המרכיב, ולא את המעבד. הפקודה אומרת ל-NASM שהקוד שלנו נטען מהכתובת 0x7c00 ,בניגוד ל-0 x0000 .סיבה זו נעוצה בכך שה-BIOS טוען את התכנית לכתובת זו.



הסימן $ הוא סימן שאינו חלק מההגדרה הכללית של שפת הסף, אבל הוא תוסף נחמד שהוסיפו למרכיב שאנחנו נשתמש בו .הסימן בעצם אומר "פה" - המרכיב מחליף את הסימן בכתובת של תחילת השורה שבה הסימן כתוב. קפיצה לכתובת של אותו שורה בעצם תפעיל את הפקודה הזאת שוב ושוב.



בשורה זו לא קיימות שום פקודות ששייכות לשפת הסף הכללית, והיא מכילה רק סימונים ופקודות מיוחדים ש-NASM מבין. הראשון זה פקודת ה-TIMES - פקודה זו פשוט משכפלת את קוד שפת הסף שנותנים לה מספר פעמים נתון. במקרה זה הפקודה שאנו רוצים לשכפל היא "0 db "ומספר הפעמים הוא ($$-$)-510 .הסימן המיוחד השני הוא $$ - בדומה ל-$, שמסמן את הכתובת של תחילת השורה, הסימן $$ מסמן את הכתובת של תחילת ה-section או במילים אחרות הסימן אומר "ההתחלה") במקרה שלנו, קיים רק section אחד ולכן הסימן מייצג את הכתובת של תחילת הקוד). לכן, הביטוי $$-$ בעצם נותן לנו את האורך של הקוד שכתבנו עד עכשיו. ה-MBR אמור להיות באורך של 512 בתים, כששני הבתים האחרונים הם חתימה ייחודית של ה -MBR .לכן אנו צריכים למלא את 510 הבתים האחרים במשהו - וכל מה שלא חלק מהקוד שלנו אנחנו צריכים למלא באפסים.



כפי שצוין, פה אנו מכריזים על אזור בגודל של מילה שיכיל את הערך 55aa , - החתימה הייחודית של ה mbr.