Ilk labelda goruldugu gibi on altinci yazmac uzerine asagida 8-13 portlarindan sadece 9-11 portlarinda calismasini ve bu portlarda istege gore anot veya katot davranmasini isteyecegimiz degeri yaziyoruz. Bunu EEPROM icerisinde olan veri yazmaclarina kaydetmek icin EEDR adi altinda Data Register icerisine atiyoruz. Bir sonraki islemimiz ise EEPROM veri yazmacindaki deger veya degerleri yazdirmaktir. EEPROM Control Register adi altinda kontrolcuden ana yazma islemlerini aktif etmesi icin EEMPE’nin aktif olmasi gerekir. Sonra ise yine Control Register tarafindan EEPROM’un yazma islemini aktif etmek icin EEPE’nin aktif olmasi gerekir. sbi denen anahtar kelime , giris cikis yazmaclari icerisinde belirli bitleri ayarlamak icin kullanilir. Yani Control Register’da istenen kontrolun calismasini ayarlamak icin sbi anahtar kelimesini kullanmak gerekir.

Sonrasinda sadece bu label kismini debug olarak Microchip Studio uzerinden calistirip hex dosyasini alip AVRDUDES adi altinda Arduino UNO uzerine kodlarimizi yuklemesi icin seciyoruz. Artik Arduino UNO’nun uzerinde olan EEPROM icerisinde kaydettigimiz deger duruyordur. Simdi ise ilk label kismini yorum satirlari icerisine alip EEPROM uzerinden veriyi almaya calisacagiz. EEPROM kontrol yazmacina EEPROM uzerinden okuma yapmamiz gerektigini soylememiz icin EERE yani EEPROM Okuma Aktif bitini bir yapmamiz gerekir. Okunan veri EEPROM veri yazmacindan almamiz icin IN komutunu kullanarak yirminci yazmaca verimizi yaziyoruz. Oncesinde belki yirminci yazmacta veri olup olmadigini bilemedigimiz icin icerisine sifir degerini yaziyoruz.

EEPROM uzerinde yazdigimiz onceki degeri led label i icerisinde kullanacagiz. Burada bizi DDRB ve PORTB komutlari karsiliyor. PORTB , gelen bitlik sistemdeki bir yazili kisimlarda anot olarak davranirken ; DDRB ise katot olarak baglanir. Sonrasinda LED’ imizin sonsuz bir dongude yanmasi icin tekrar tekrar bu label i sonda cagiriyoruz.