

Konputagailuen Arkitektura I

Laborategia 2 (ebazpena): inkesta + pantaila

Lantegi batean suteak automatikoki detektatzea nahi dute, horretarako PC bateragarri bat erabiliz (i8086 Intel mikroprozesadorea). Suteak detektatzeko sentsore berezi bat erabiltzen da, KSUA izanik bere kontroladorea. Sentsore honek sua dagoela detektatzean IRQ3 eten-lerrotik eten bat sortzen du, datu erregistroan (0x381 helbidea) suteari buruzko informazioa utziz (sutearen egoera adierazten duen karaktere bat gordetzen du). KSUA kontroladorearekin sinkronizazioa etenen bidez izan daitekeen arren, sinkronizazio hori inkesta bidez burutzea erabaki da (teklatuarekin egin izan dugun moduan). Horretarako beharrezkoa da kontroladore honen etenak galaraztea. Kontroladore honen etenak galarazten dituen errutina C lengoaiari programatzea eskatzen da: `void KSUA_Etenak_Galarazi();`

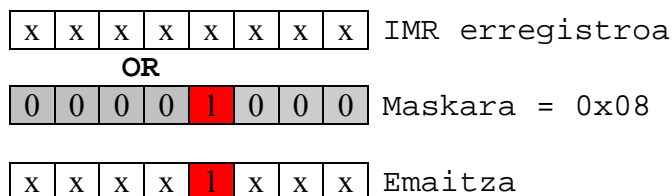
Behin etenak galarazita inkesta bidezko sinkronizazioa erabiliko denez, inkesta hori burutzen duen errutina ere idatzi behar duzu C lengoaiari: `unsigned char Inkesta_KSUA()`. Errutina honek kontroladoreko datu erregistroan gordeta dagoen karakterea itzultzen du. `strobe` sekuentzia bat ere egin behar da kontrol erregistroko (0x382 helbidea) 2 bitean. `strobea` egiten duen errutina kodetzea ere eskatzen da: `void strobe_KSUA()`.

Aurreko errutinan irakurritako suteari buruzko informazioa pantailatik atera behar da. Pantaila memorian mapeatuta dago C000h helbidetik aurrera eta 48 lerro eta 160 zutabe dauzka. Idatzi, C lengoaiari erabiliz, lerro eta zutabe parametroek adierazten duten posizioan karaktere bat idazten duen errutina:

```
void IdatzKar (int lerro, int zutabe, unsigned char kar,
              unsigned char atrib);
```

Ebazpena

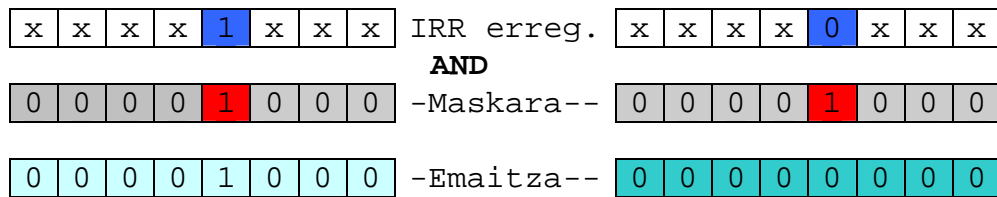
Kodetzea eskatzen den lehenengo errutina KSUA kontroladorearen etenak galarazten dituen da. Horretarako eten-kontroladorean dagoen maskara-erregistroko (IMR erregistroa) 3 bitari (KSUA kontroladoreak 3 zenbakia duen eten-lerroa erabiltzen duelako) 1 balioa eman behar zaio. IMR erregistroa irakurri ondoren, OR eragiketari bat egiten badugu 0x08 maskarekin, emaitzako 3 bitak bat balioa hartuko du (eskeman ikus daitekeen bezala). Emaitza hori IMR erregistroan idatziz bukatuko da eragiketari. Agindu hauek jarraian exekutatzen direla ziurtatzeko hasieran eten guztiak galarazten dira eta bukaeran berriro baimentzen dira.



```
void interrupt KSUA_Etenak_Galarazi()
{
    unsigned char bal;
```

```
    DisableInts(); //Eten guztiak galarazi
    bal = InPort(IMR_ERREG_EK8259); //IMR erregistroa irakurri eta bal aldagaian gorde
    bal = bal | 0x08; //OR eragiketari egin 0x08 maskarekin 3 bitak 1 balioa hartzeko bal
    OutPort(IMR_ERREG_EK8259, bal); //bal aldagaia IMR erregistroan idatzi
    EnableInts(); //Etenak baimendu berriro
}
```

KSUA gailuaren etenak galarazi ditugunez, KSUA gailuak eteten duenean IRQ3 lerroa aktibatuko du eta horren ondorioz IRRko 3 bita aktibatuko da baina prozesadorera ez da etenik iritsiko. Gailu honen inkesta IRR erregistroko 3 bitean burutu behar da, bit hori noiz aktibatzen den aztertuz. Horretarako IRR erregistroaren eta 0x08 maskararen artean AND eragiketa burutzen da. Eragiketa honen emaitza 0 izango da IRRko 3 bitak 0 balio duen bitartean, eta 0ren desberdina IRRko 3 bitak 1 balio duenean (ikus irudia).



Inkesta bukatzen denean KSUA kontroladorearen datu erregistroa irakurtzen da hor dagoen balioa eskuratzeko; strobe bat egin eta irakurritako karakterea itzultzen da.

```

unsigned char Inkesta_KSUA()
{
    unsigned char Nire_Irr;
    unsigned char Kar;

    //IRR erregistroa irakurri
    OutPort(IRR_ERREG_EK8259, 0x0A);
    Nire_Irr = InPort(IRR_ERREG_EK8259);

    //Inkesta egin, IRRko 3 bitak 0 balio duen bitartean
    while ((Nire_Irr & 0x08)==0)
    {
        OutPort(IRR_ERREG_EK8259, 0x0A);
        Nire_Irr = InPort(IRR_ERREG_EK8259);
    }
    Kar = InPort(0x381); //KSUAko datu-erregistroa irakurri
    strobe_KSUA();
    return(Kar);
}

```

Strobe eragiketa KSUA kontroladoreko kontrol-erregistroko 2 bitean egin behar da. Horretarako bit horri lehendabizi 1 balioa eman behar zaio (OR eragiketa) eta ondoren 0 balioa (AND eragiketa).

```

void strobe_KSUA()
{
    unsigned char bal;

    bal = InPort(0x382); //Teklatuaren kontrol-erreg. irakurri
    bal = bal | 0x04 ;//2 bita 1era jarri
    OutPort(0x382,bal); //Kontrol-erreg. idatzi
    bal = bal & 0xFD; //2 bita 0ra jarri
    OutPort(0x382,bal); //Kontrol-erreg. idatzi
}

```

Karaktere bat pantailan idatzi ahal izateko, karakterea eta atributua idatzi behar dira memorian, C000h segmentutik aurrera. Informazio hori memorian idazteko (byte bateko karaktereak dira) `IdatzByteFis` funtzioa daukagu. Helbide txikienean karakterea eta hurrengoan atributua idatzi behar dira. Hasierako helbidearekiko desplazamendua adierazteko kontuan hartu behar da karaktere bakoitzeko bi byte idazten direla (karaktere eta atributua) eta pantailak 160 karaktere dituela lerroko.

```
void IdatzKar(int lerro, int zut, unsigned char kar,
              unsigned char atrib)
{
    IdatzByteFis(C000h, ((lerro*160)+zut)*2, kar);
    IdatzByteFis(C000h, (((lerro*160)+zut)*2)+1, atrib);
}
```