الفصل الثالث 3

طرق التقابل مع الحاسب

3-1 مقدمة

كما نعلم من طبيعة عمل المعالج فإنه يتعامل مع الذاكرة بمجموعة أوامر معينة يستطيع بها القراءة من الذاكرة أو الكتابة فيها ، حيث فى كل مرة من مرات التعامل يستطيع إرسال أو استقبال 1 أو 2 أو 4 أو 8 بايت . كذلك فإن المعالج لديه أمران آخران مهمان يستطيع بهما إرسال أو استقبال 1 أو 2 بايت إلى أو من أى جهاز خارجى . هذان الأمران هما أمر الإدخال IN الذى به يستطيع قراءة بايت أو اثنين من أى جهاز إدخال ، وأمر الإخراج OUT الذى به يستطيع إخراج بايت أو اثنين لأى جهاز إخراج . مجموعة العناوين التى يتعامل معها هذان الأمران تختلف تماما عن مجموعة العناوين الخاصة بالذاكرة ، وليست مقتطعة من خريطة الذاكرة الأساسية للحاسب ولكن لها خريطة خاصة بها وتسمى خريطة الإدخال والإخراج . أقصى عنوان يتم التعامل معه بالأمرين IN و OUT مكون من 16 بت (2 بايت) ، وهذا يعنى أن عدد العناوين أو الأجهزة التى سيتم التعامل معها من خلال هذين الأمرين سيكون 64 كيلو عنوان (65536 عنوان) أو جهاز خارجى . طريقة الإدخال والإخراج للبيانات باستخدام هذين الأمرين تسمى طريقة الإدخال والإخراج المنفصل Isolated input output ، وهذه الطريقة سندرسها بالتفصيل فى هذا المقرر .

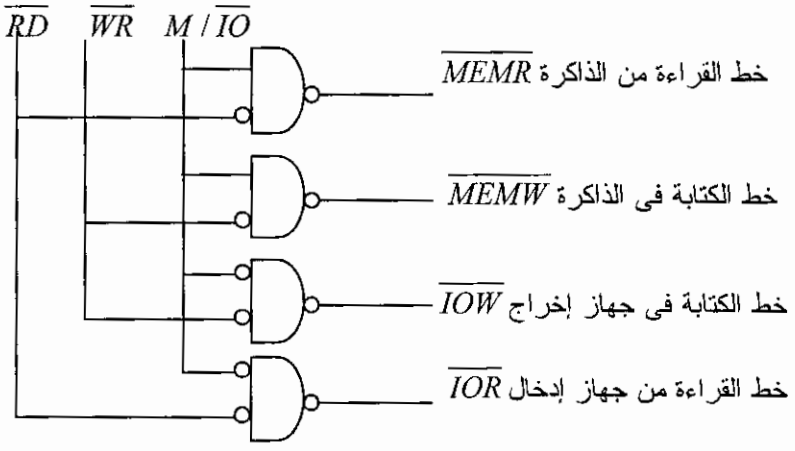
يمكن استخدام عناوين الذاكرة كعناوين إدخال وإخراج لأجهزة خارجية وهذه الطريقة تسمى طريقة خرائط الذاكرة للإدخال والإخراج Memory mapped input output ، وسنلقى الضوء سريعا على هذه الطريقة فيما يلى ولكن قبل أن ندخل فى أى تفاصيل أخرى سنلقى الضوء على بعض الأطراف المهمة المستخدمة فى ذلك . هذه الأطراف تكون متاحة على أطراف المسارات القياسية التى سنتعامل معها ، وهذه الأطراف هى :

1. الطرف $\overline{RD}$ ، وهذا الطرف يكون فعالا (0) عندما يكون المعالج فى حالة قراءة بيانات من أى مصدر خارجى سواء كان هذا المصدر الذاكرة أو أى جهاز إدخال .
2. الطرف $\overline{WR}$ ، وهذا الطرف يكون فعالا (0) عندما يكون المعالج فى حالة كتابة بيانات فى أى هدف خارجى سواء كان هذا الهدف الذاكرة أو أى جهاز إخراج .
3. الطرف $M/\overline{IO}$ ، هذا الطرف يبين عليه المعالج إذا كان تعامله (القراءة أو الكتابة) سيكون مع الذاكرة فيجعل هذا الخط (1) أم مع جهاز خارجى فيجعل هذا الخط (0) ، لاحظ وجود شرطة على الحرفين IO فقط . لذلك فإنه عندما يكون هذا الخط (1) وفى نفس الوقت يكون الخط $\overline{RD}$=0 فإن ذلك يعنى أن

المعالج سيقرأ من الذاكرة . بنفس الطريقة يمكن معرفة إذا كان المعالج سيكتب فى جهاز إخراج أو سيكتب فى الذاكرة . جدول 3-1 يبين كل هذه الحالات والمحددة فى أربع حالات وهى $\overline{MEMR}$ و $\overline{MEMW}$ و $\overline{IOR}$ و $\overline{IOW}$. كل هذه الحالات يمكن الحصول عليها فى صورة خطوط تحكم تستخدم لتنشيط الذاكرة أو أجهزة الإدخال أو الإخراج وهذه الخطوط فى الغالب تكون متاحة على كل المسارات القياسية لاستخدامها فى أى تطبيق . شكل (3-1) يبين كيفية الحصول على هذه الخطوط .

جدول 3-1 حالات القراءة والكتابة المختلفة

الحالة	الطرف $\overline{RD}$	الطرف $\overline{WR}$	الطرف $M / \overline{IO}$
قراءة من الذاكرة $\overline{MEMR}$	0	1	1
كتابة فى الذاكرة $\overline{MEMW}$	1	0	1
قراءة من جهاز إدخال $\overline{IOR}$	0	1	0
كتابة فى جهاز إخراج $\overline{IOW}$	1	0	0



شكل (3-1) الحصول على خطوط التحكم الأربعة $\overline{MEMR}$ و $\overline{MEMW}$ و $\overline{IOW}$ و $\overline{IOR}$

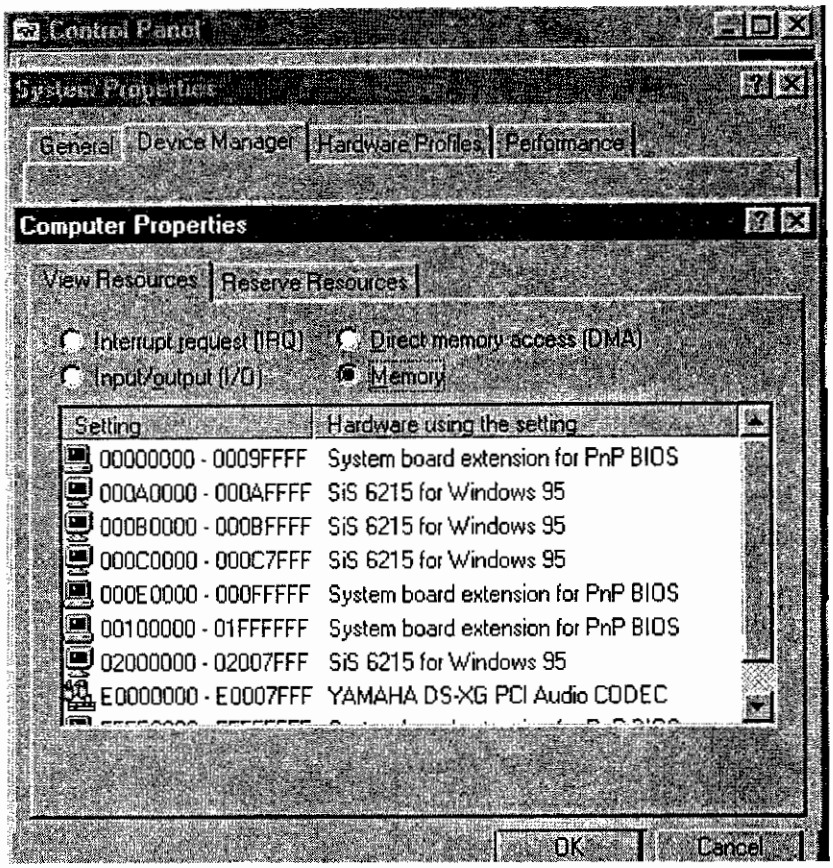
2-3 طريقة خرائط الذاكرة للتقابل مع الحاسب
Memory mapped IO

حينما يتقابل المعالج مع الذاكرة بأحد أوامر القراءة أو الكتابة فإن هذا الأمر لابد أن يحدد العنوان الذى سيتم التعامل معه بدقة ، فإذا كان مسار العناوين لهذا المعالج مكون من 32 بت مثلا ، فلابد أن يكون العنوان الموجود فى هذا الأمر مكون أيضا من 32 بت حتى يتم تحديده بدقة ولا نترك فرصة لإمكانية التعامل مع أكثر من بايت بعنوان واحد ، وهذا ما أسميناه بالعنونة الناقصة partial decoding فى الكتاب السابق (كتاب المعالجات) . إن العنونة الناقصة من الممكن أن تتسبب فى تداخلات ومشاكل كثيرة نتيجة انشغال أماكن كثيرة ومتفرقة من الذاكرة ببرامج التشغيل الخاصة بالحاسب . لذلك فإنه فى حالة استخدام عناوين الذاكرة لعنونة أجهزة خارجية فلابد من فحص خريطة الذاكرة الخاصة بالحاسب المستخدم بدقة لمعرفة أى الأماكن تم حجزها لنظام التشغيل وأيها محجوز للشاشة ، وأيها متروك للاستخدام الحر بواسطة المبرمجين . خريطة الذاكرة هى رسم توضيحى يبين عنوان البداية والنهاية لكل جزء من الأجزاء المستخدمة فى كل المدى العنوانى للحاسب (المدى العنوانى للحاسب = 2 أس عدد خطوط مسار العناوين) . هذه الخريطة يمكن الحصول على محتوياتها من برنامج النوافذ windows98 وما بعده كما يلى :

Start → Setting → Control Panel → System → Device Manager → Properties → Memory

شكل (3-2) يبين مثال لنافذة توضح خريطة الذاكرة لأحد أجهزة الحاسب تم الحصول عليها باتباع سلسلة الأوامر السابقة ، وشكل (3-3) يبين رسما تخطيطيا لهذه الخريطة بناء على هذه العناوين التى تم الحصول عليها فى شكل (3-3) . لاحظ أن العناوين مكتوبة فى النظام الستعشرى وكل عنوان مكتوب فى 8 خانات ، أى أن كل عنوان مكتوب فى 32 بت ثنائية . أى أن مسار العناوين الخاص بالمعالج المستخدم فى هذا الحاسب يتكون من 32 خط .

كما نرى من شكلى (3-2) أو (3-3) فإن هناك بعض العناوين الغير مستخدمة بأنظمة التشغيل والتى يمكن استخدامها كعناوين لأجهزة إدخال وإخراج . من أمثلة هذه العناوين المدى العنوانى من العنوان 000C8000 إلى العنوان 000DFFFF . أى عنوان من هذه العناوين يتم استخدامه لأغراض الإدخال والإخراج لن يستخدم كذاكرة فيما بعد وإلا فإنه من الممكن أن تحدث مشاكل نتيجة هذا الاستخدام المزدوج .

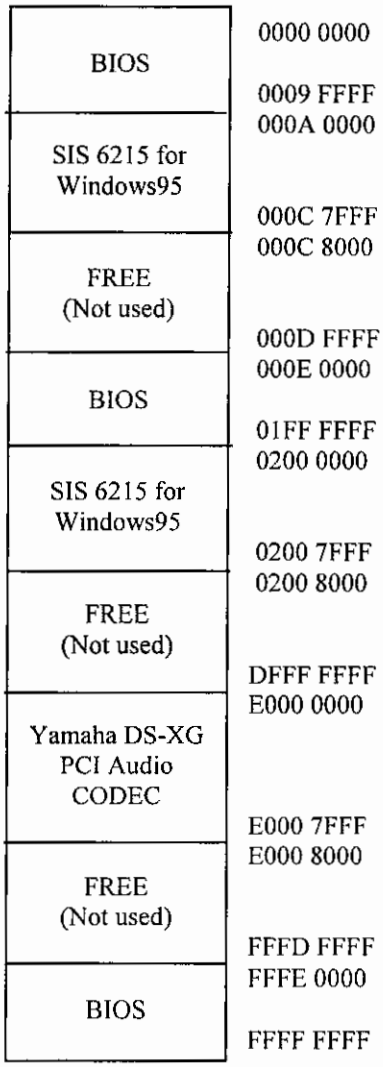


شكل (3-2) محتويات خريطة الذاكرة لأحد الحاسبات الشخصية

شكل (3-4) يبين رسما تخطيطيا لبوابة إدخال باستخدام خرائط الذاكرة . من أهم مكونات هذه البوابة منتقى العناوين أو فاكك الشفرة Address decoder الذى ينشط خرجه فقط عند إدخال العنوان المصمم من أجله . لابد أن تشارك الخطوط $\overline{RD}$ و $M/\overline{IO}$ معا أو الخط $\overline{MEMR}$ بدلا منهم فى عملية الانتقاء . فى حالة بوابات الإخراج يمكن استخدام نفس المنتقى ولكن فى هذه الحالة سنستخدم الخطوط $\overline{WR}$ و $M/\overline{IO}$ معا أو الخط $\overline{MEMW}$ بدلا منهم .

من المكونات الأخرى المهمة فى بوابة الإدخال العازل buffer الذى من الممكن أن يكون 8 أو 16 بت على حسب التصميم المطلوب للبوابة . العازل عبارة عن بوابة ثلاثية المنطق يكون خرجه مقاومة عالية طالما أن خط التحكم الخاص بالبوابة غير فعال . عند تنشيط خط التحكم يصبح الخرج متصلا بالدخل فتنتقل

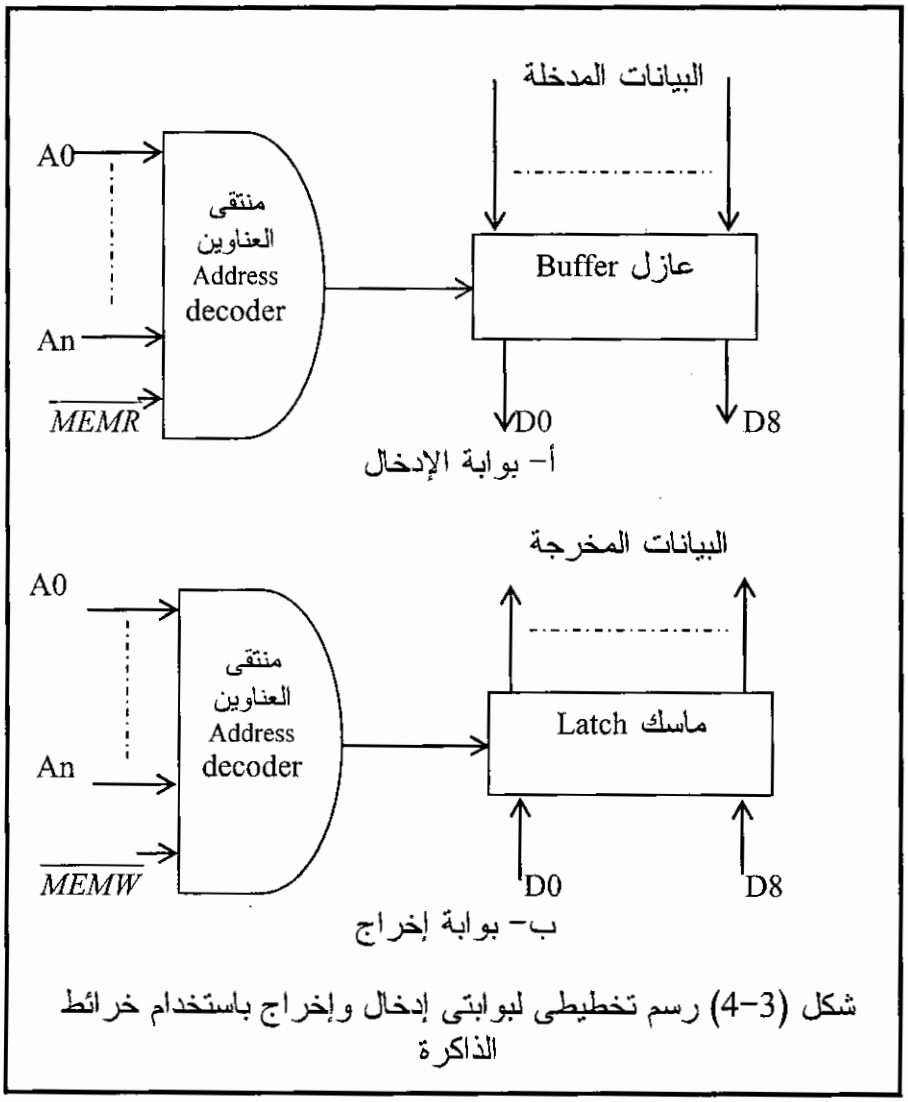
البيانات المطلوب قراءتها إلى مسار البيانات ومنه إلى المعالج حيث تخزن فى المركم Accumulator .



شكل (3-3) خريطة الذاكرة لأحد أجهزة الحاسب الشخصى

لقد تم استخدام ماسك Latch بدلا من العازل فى حالة بوابة الإخراج حيث ستكون مهمة هذا الماسك هى مسك البيانات المخرجة من المعالج من خلال مسار البيانات عند تنشيطه بواسطة الإشارة القادمة من منتقى العناوين ، وبذلك

يمكن الاستفادة من هذه البيانات فى التطبيق المطلوب . مسار البيانات من الممكن أن يكون 8 أو 16 بت . لاحظ أيضا أن دخل منتقى العناوين لابد أن يكون جميع خطوط مسار العناوين من A0 إلى An حيث n هنا ستكون آخر خط من خطوط العناوين .



شكل (3-4) رسم تخطيطى لبوابتى إدخال وإخراج باستخدام خرائط الذاكرة

4-3 طريقة الإدخال والإخراج المنفصل للتقابل مع الحاسب
Isolated Input Output

هناك مجال آخر من العناوين لا يتم التعامل معه من خلال أوامر التعامل مع عناوين الذاكرة العادية ، ولكن يتم التعامل معه من خلال أمرين فقط وهما أمر الإدخال IN وأمر الإخراج OUT . كل من الأمرين يحتوى عنوان مكون من 8 أو 16 بت فى حالة الإدخال أو الإخراج . لذلك فإن المدى العنوانى فى هذه الحالة يساوى 2^{16}=65536=64 كيلو عنوان . هذا المدى العنوانى منفصل تماما عن المدى العنوانى المخصص للذاكرة ، أى أن هذه العناوين ليست مخصومة من حساب المدى العنوانى لخريطة الذاكرة الأساسية الخاصة بالحاسب لأن المعالج ينشط خطوط تحكم أخرى تختلف عن حالة التعامل مع الذاكرة . وهذا يعنى أن هناك 64 كيلو من الأجهزة الخارجية التى يمكن عنونتها لإدخال أو إخراج بيانات باستخدام هذين الأمرين .

بنفس طريقة خريطة الذاكرة تكون هناك خريطة تبين أى هذه العناوين يكون مشغولا بواسطة ملحقات الحاسب الخارجية مثل كارت الفاكس والصوت وغيرها ، وأيها يكون متروكا للاستخدام الحر بواسطة المبرمج ، هذه الخريطة تسمى خريطة الإدخال والإخراج . شكل (3-5) يبين قائمة الملحقات التى تشغل أماكن على خريطة الإدخال والإخراج فى الحاسب الذى نتعامل معه . هذه القائمة يمكن الحصول عليها من خلال قائمة الأوامر التالية الخاصة ببرنامج النوافذ windows98 كما يلى :

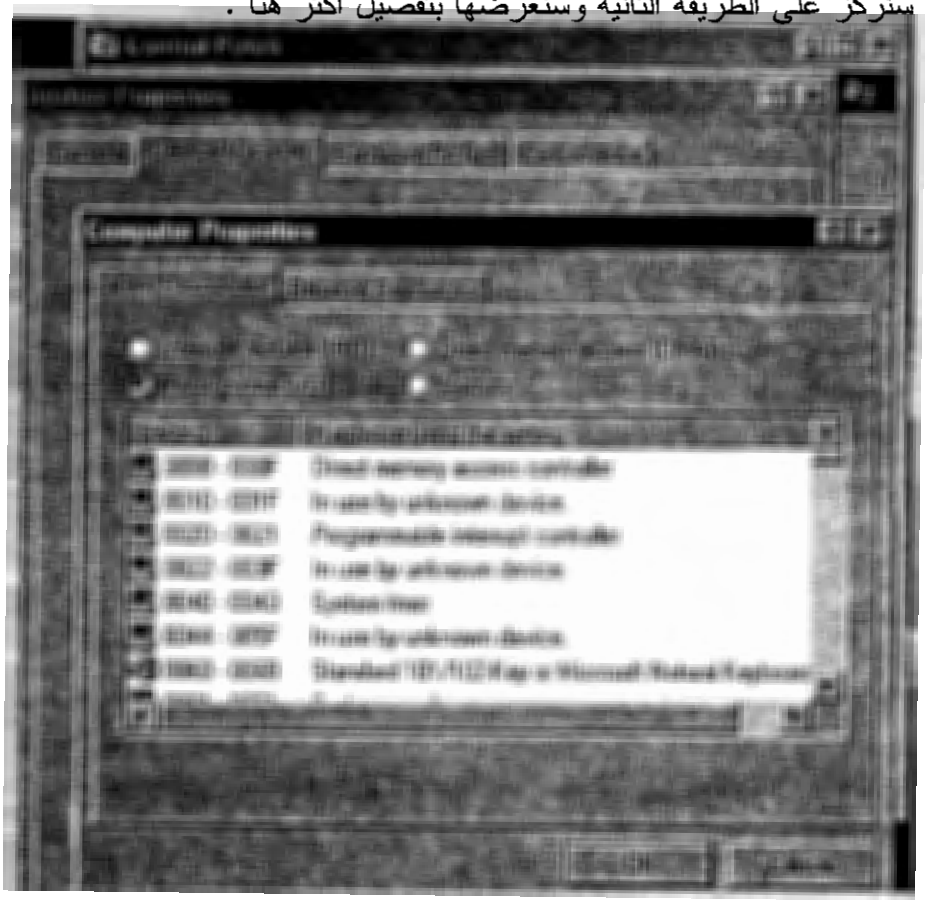
Start→ Setting→ Control Panel→ System→ Device Manager →Properties→ Input output (I/O)

القائمة الموجودة فى شكل (3-5) عبارة عن قائمة كبيرة تبين جميع الأجهزة الملحقة بالحاسب المستخدم حيث يمكن وضعها فى صورة خريطة مثل خريطة الذاكرة تماما . بالطبع فإنه بالنظر للقائمة الموجودة فى شكل (3-5) سنجد الكثير من العناوين الغير مستخدمة بهذه الملحقات والتى يمكننا استخدامها فى التطبيقات الخاصة بنا . من هذه العناوين على سبيل المثال ما يلى :

0100 to 016F
0178 to 01EF
0300 to 032F
3F77 to 3FCF

هذه الطريقة ، طريقة الإدخال والإخراج المنفصل ، هى الأكثر شيوعا فى الاستخدام من الطريقة الأولى ، خرائط الذاكرة ، لعدة أسباب منها :

1. سهولة البرمجة باستخدام هذه الطريقة وبالذات باستخدام اللغات ذات المستوى العالى ، فهناك أمران فقط واحد لإدخال البيانات IN والآخر لإخراجها OUT.
2. خطوات التصميم فى الطريقتين متشابهة لحد كبير .
3. المدى العنوانى فى الطريقة الأولى ، خرائط الذاكرة ، أوسع بكثير من الطريقة الثانية ولابد من استخدام جميع خطوط مسار العناوين فى عملية التشفير مما يجعل عملية التشفير أكثر تعقيدا من الحالة الأولى . لذلك فإننا سنركز على الطريقة الثانية وسنعرضها بتفصيل أكثر هنا .



شكل (3-5) قائمة بملحقات الحاسب الموجودة على خريطة الإدخال والإخراج

شكل (3-6) يبين رسما تخطيطيا لبوابتى إدخال وإخراج باستخدام طريقة الإدخال والإخراج المنفصل Isolated . من هذا الشكل نستطيع تدوين الملاحظات المهمة التالية :

- استخدام خطى التحكم $\overline{IOR}$ و $\overline{IOW}$ فى عملية انتقاء العنوان المطلوب بدلا من الخطين $\overline{MEMW}$ و $\overline{MEMR}$ فى الطريقة السابقة . هذه الخطوط متاحة على المسار القياسى ISA الذى سنطبق عليه كل تجاربنا .
- استخدام خط التحكم AEN فى عملية انتقاء العنوان المطلوب . هذا الخط يكون (1) فى حالة تعامل الحاسب المباشر مع الذاكرة DMA ، ويكون صفر فى غير ذلك . ونحن هنا لا نريد تنشيط أى بوابة نتعامل معها فى حالة تعامل الحاسب المباشر مع الذاكرة ، أى أننا نريد تنشيط كل بوابات الإدخال أو الإخراج فى حالة كون الخط AEN=0 فقط . سنرى ذلك بالتفصيل بعد قليل .
- نلاحظ أيضا استخدام خطوط العناوين A0 إلى A9 فقط فى عملية انتقاء العنوان المطلوب ، وهذا ما يحدث بالفعل فى كل أجهزة الحاسبات IBM حيث أنها تستخدم واحد كيلو فقط (10 خط عناوين) من 64 كيلو (16 خط عناوين) الموجودة فى خريطة الإدخال والإخراج . وهذا ما سنلتزم به فى كل تصميماتنا . لاحظ أن ذلك يعنى أن IBM تستخدم التشفير الناقص لانتقاء العناوين التى تستخدمها فى كل التطبيقات . وهذا يعنى أن كل جهاز يشغل عنوان معين فى مدى الواحد كيلو الأولى سيكون له صور أخرى يشغل بها عناوين أخرى فى المدى العنوانى المتبقى . ولذلك فإننا ننصح بعدم استخدام المدى العنوانى بعد العنوان 03FF (2كيلو) فى تشفير أى بوابة إدخال أو إخراج لأن ذلك سيسبب مشاكل مع الأجهزة الأخرى ، وهذه ملاحظة مهمة نحذر منها .

شكل (3-7) يبين رسما تفصيليا لبوابة إدخال وبوابة إخراج كل منهما لها العنوان 300H . فى هذا الشكل تم استخدام منتقى العناوين 74138 لانتقاء عناوين 8 بوابات ، 4 منها للإدخال ، والأربعة الأخرى للإخراج وكلها لها الأرقام 300 إلى 303 كما فى الشكل .

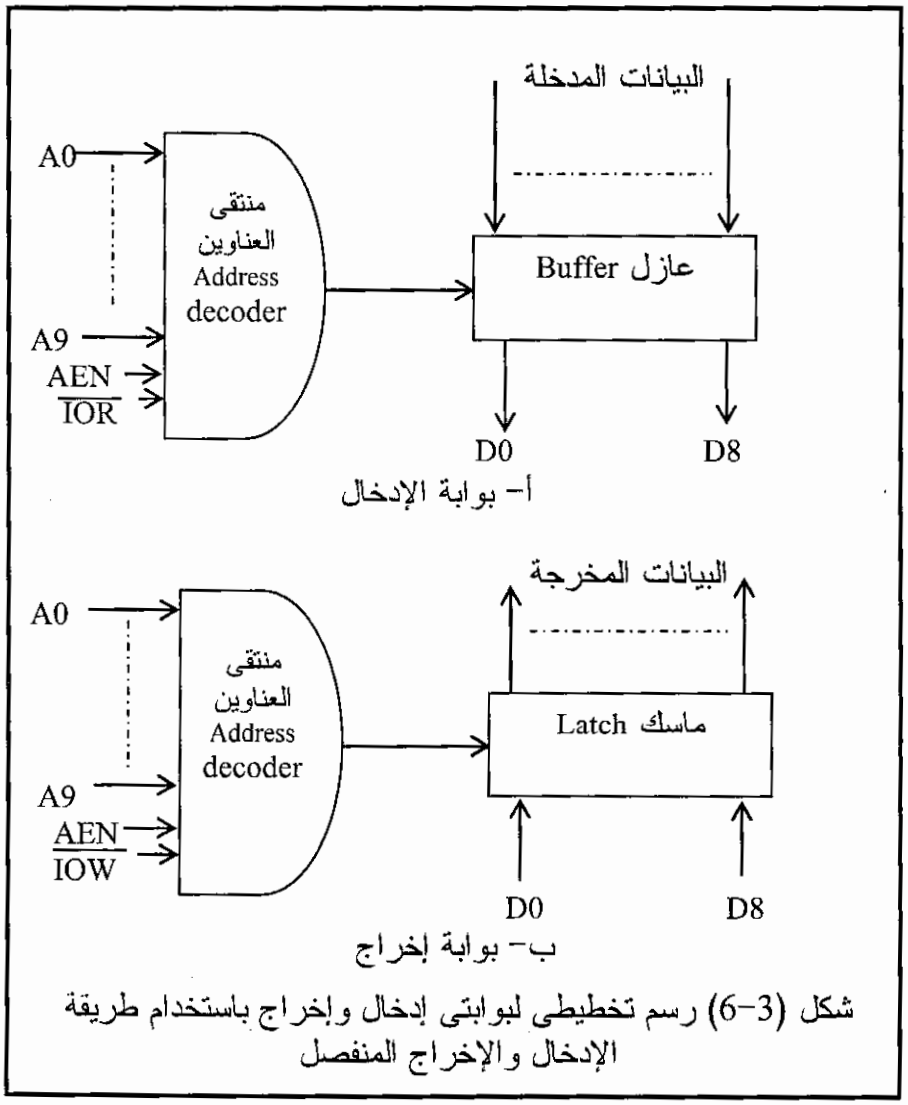
لقد تم استخدام الشريحة 74LS244 وهى عازل ثمانى كبوابة إدخال ، والشريحة 74LS373 وهى ماسك ثمانى كبوابة إخراج . يمكن التعامل مع هذه البوابات من خلال أوامر أكثر من لغة سنعرض أمثلة لها فى لغة التجميع ولغة C . فى لغة التجميع يمكن قراءة محتويات بوابة إدخال كما فى الأمرين التاليين :

```
MOV DX,0300H
IN AL,DX
```

حيث الأمر الأول يضع عنوان البوابة المراد التعامل معها فى المسجل DX ، والأمر الثانى يضع محتويات البوابة التى عنوانها فى المسجل DX فى المسجل AL . لاحظ أن بوابة الإدخال فى هذه الحالة مكونة من 8 بت .

فى لغة C++ يمكن استخدام الأمر التالى لقراءة محتويات بوابة إدخال :

value = inportb (Port address);



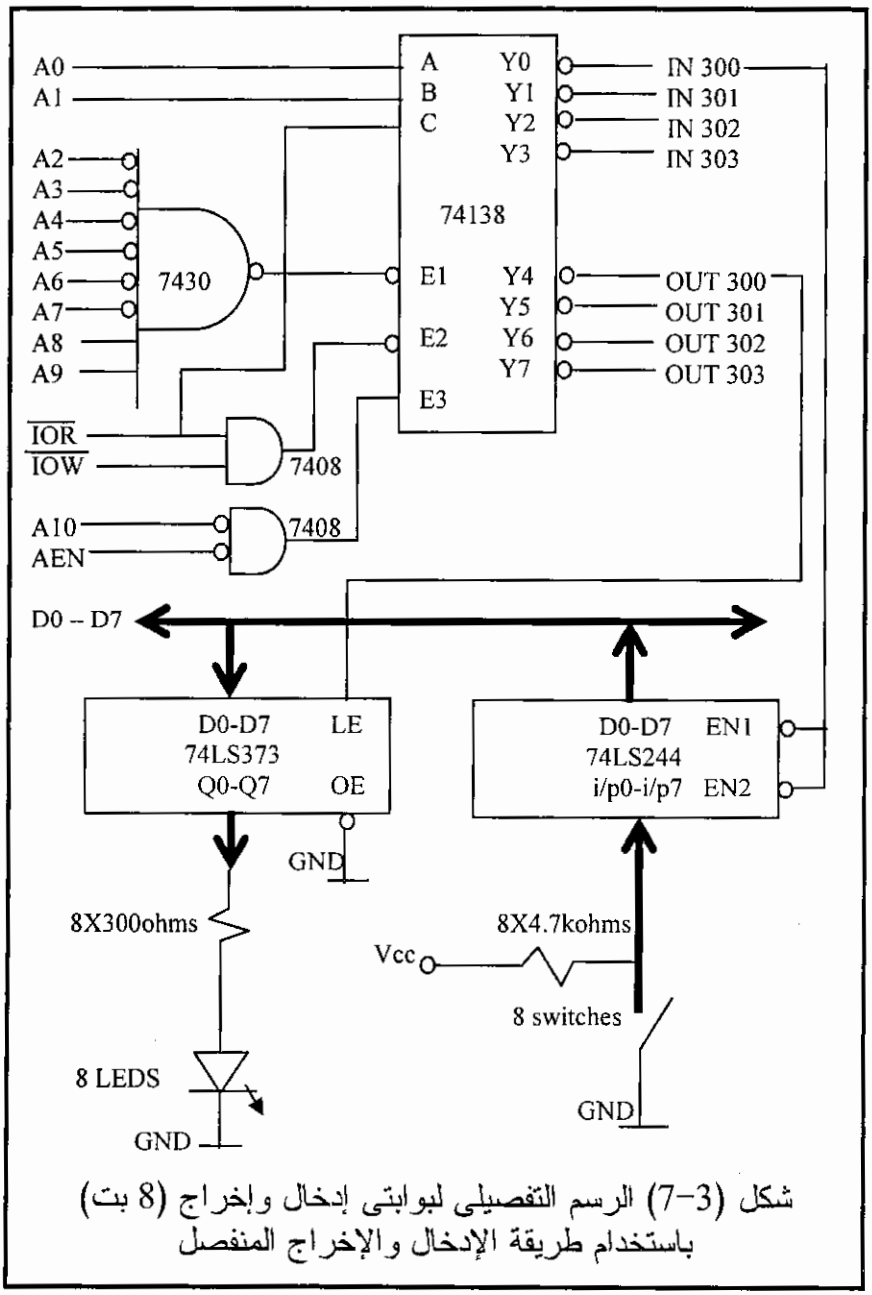
شكل (3-6) رسم تخطيطى لبوابتى إدخال وإخراج باستخدام طريقة الإدخال والإخراج المنفصل

حيث Port address هو عنوان البوابة المراد قراءة محتوياتها ، فمثلا لقراءة محتويات البوابة رقم 300H نكتب الأمر التالى :

Value = inportb (0x300);

يمكن إخراج محتويات المركم AL على البوابة رقم 300 من خلال الأمرين التاليين :

```
MOV DX,0300H
OUT DX,AL
```



شكل (3–7) الرسم التفصيلى لبوابتى إدخال وإخراج (8 بت) باستخدام طريقة الإدخال والإخراج المنفصل

حيث الأمر الثانى يخرج محتويات المركم AL على البوابة التى عنوانها فى المسجل DX . يمكن عمل نفس الشىء فى لغة C كما يلى :

```
outportb (port address, value);
```

حيث هذا الأمر سيخرج القيمة value على البوابة التى عنوانها هو port address. كمثال على ذلك الأمر التالى :

```
outportb (0x300,value);
```

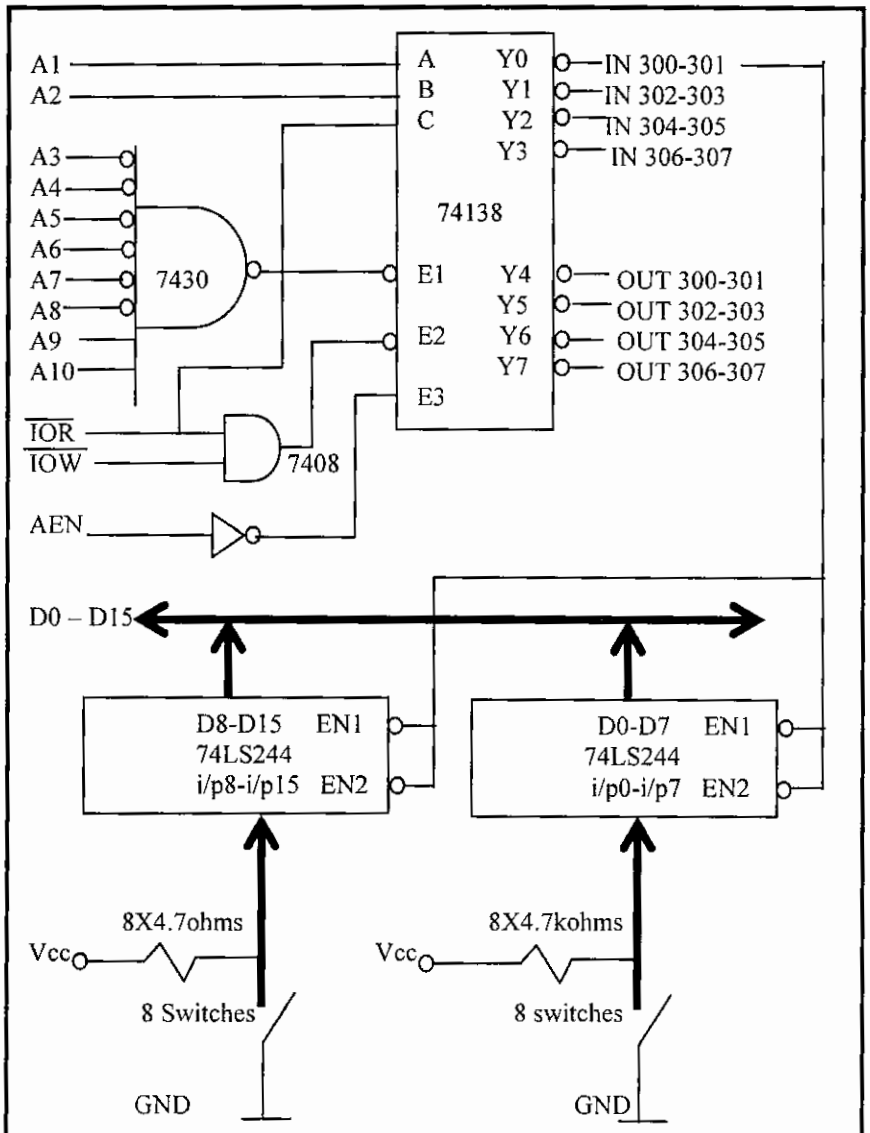
البرنامج التالى يقرأ محتويات البوابة رقم 300H ويخرجها على البوابة رقم 300H وذلك باستمرار فى حلقة مغلقة :

```
#define portAddress 0x300
#include <dos.h>
main()
{ int value;
  for(;;)
   { value = inportb(portAddress);
     outportb (portAddress, value);
   }
}
```

بتنفيذ هذا البرنامج فى كارت المواجهة الذى تم تجهيزه كما فى شكل (3-7) فإن أى بيانات يتم إدخالها من خلال المفاتيح المتصلة على بوابة الإدخال (74244) سنجدها فورا على خرج بوابة الإخراج (74373) . حاول تنفيذ الدائرة الموجودة فى شكل (3-7) واختبارها بالبرنامج السابق .

فى الكثير من التطبيقات يتطلب الأمر التعامل مع 2 بايت (16 بت) من البيانات كما فى المحولات الرقمية الانسيابية (D/A) أو المحولات الانسيابية الرقمية (A/D) والتى تكون فى كثير من الأحيان أكثر من بايت واحدة . فى هذه الحالة يمكن استخدام بوابتين منفصلتين كل منهما بعنوان خاص كما فى شكل (3-7) وإجراء عملية الإدخال أو الإخراج بأمرين متعاقبين لكل بوابة . فمثلا يمكن تصميم بوابتى الإدخال رقم 300H و 301H واستخدام الأمرين التاليين لقراءة محتوياتهما :

```
MOV DX, 0300H
IN AL, DX
MOV BL,AL
MOV DX,0301H
IN AL,DX
MOV BH, AL
```



شكل (3-8) الرسم التفصيلى لبوابة إدخال (16 بت) باستخدام طريقة الإدخال والإخراج المنفصل

بذلك تصبح محتويات البوابتين 300H و 301H فى المسجل BX بحيث أن البوابة 300 تكون موجودة فى البايت ذات القيمة الصغرى BL والبوابة 301 موجودة فى البايت ذات القيمة العظمى BH .

شكل (3-8) يبين تصميما لبوابة إدخال من 16 بت يمكن قراءتها بأمر واحد فقط . فى شكل (3-8) تم استخدام خط تنشيط واحد لكل من الرقمين 300 و 301 وهذا الخط سيقوم بتنشيط كل من الشريحتين 74244 لإدخال 16 بت على مسار البيانات D0 إلى D15 وتخزين هذه المحتويات فى المسجل AX مباشرة ، وتم ذلك عن طريق عدم أخذ الخط A0 فى عملية التشفير . حاول وضع عناوين مختلفة على مسار العناوين ولاحظ خطوط التنشيط التى تخرج من المنتقى 74138 . فى هذه الحالة يمكن قراءة البوابة كما يلى :

```
MOV DX, 300H
IN AX, DX
```

فى هذه الحالة سيتم قراءة كل من البوابتين 300H و 301H ووضعهم فى المركم AX مرة واحدة .

بذلك نكون قد رأينا كيفية توصيل بوابة إخراج أو بوابة إدخال على كارت مواجهة يوضع على أى مسار من مسارات المعالج القياسية ، وقد اقترحنا هنا استخدام كارت ISA .