الوحدة الرابعة: الدوائر المنطقية الترابطية Combinational Logic Circuits

محتويات الوحدة

تمهيد

أهداف الوحدة

1. الدوائر المنطقية الترابطية
2. بعض استخدامات بوابة XOR
   - 2-1 العاكس المنطقي المحكوم
   - 2-2 التحويل من الشفرة الثنائية إلى الشفرة الرمادية
   - 2-3 دوائر التحقق (Parity Checking)
3. دوائر الجمع (Adders)
   - 3-1 نصف الجامع (Half Adder)
   - 3-2 الجامع الكامل (Full Adder)
   - 3-3 الجامع متعدد الخانات (Multi-bit Adder)
   - 3-4 عملية الطرح (Subtraction)
   - 3-5 وحدة الحساب (Arithmetic Unit)
4. فاك الشفرة (Decoder)
5. المشفر (Encoder)
6. الدامج (Multiplexer)
7. المفرق (Demultiplexer)
8. طرق بديلة لتصميم الدوائر المنطقية
   - 8.1 التصميم باستخدام فاك شفرة و مشفر (Decoder & Encoder)
   - 8.2 التصميم باستخدام الدامج (Multiplexer)

تمهيد

مرحباً بك عزيزي الدارس في الوحدة الرابعة من مقرر "أساسيات التصميم المنطقي". نقوم في هذه الوحدة بعرض بعض الدوائر المنطقية الترابطية (Combinational Logic Circuits) التي تقوم بأداء وظائف مفيدة، و التي يتوفر أغلبها تجارياً بصورة جاهزة في شكل دوائر متكاملة (Integrated Circuits أو IC's)، بحيث يمكن شراؤها و استخدامها مباشرة في بناء الأنظمة الرقمية. أي أنه من الناحية العملية ليس من الضروري أن نستخدم مهارات تصميم الدوائر المنطقية التي تعلمناها في الوحدة السابقة لتصميم جميع الدوائر المنطقية التي نحتاج إليها في نظام رقمي معين، بل يمكن استخدام الدوائر الجاهزة في بناء جزء كبير من النظام الرقمي، و تصميم عدد قليل من الدوائر التي قد لا تكون

PDF created with pdfFactory Pro trial version www.pdffactory.com

متوفرة تجارياً. و سنتعرف في هذه الوحدة على عدد من تلك الدوائر الجاهزة و ندرس خصائص كل دائرة منها و استخداماتها و طرق ربطها مع بعضها البعض، كما سنتعرف على طرق بديلة لتصميم الدوائر المنطقية باستخدام تلك الدوائر الجاهزة.

أهداف الوحدة

عزيزي الدارس، بعد دراسة هذه الوحدة ينبغي أن تكون قادراً على:

- توضيح المقصود بالدوائر المنطقية الترابطية.
- استخدام بوابة XOR في تصميم دوائر منطقية مفيدة.
- تصميم دوائر منطقية تقوم بإجراء عمليتي الجمع و الطرح.
- تصميم وحدة حساب (Arithmetic Unit) و استخدامها في الأنظمة الرقمية.
- فهم وظيفة كل من فاك الشفرة و المشفر و الدامج و المفرق و استخدامها في الأنظمة الرقمية.
- ربط الدوائر المنطقية الترابطية مع بعضها البعض لبناء دوائر أكبر.
- تصميم الدوائر المنطقية بالطرق البديلة التي تستخدم فيها الدوائر الجاهزة.

1- الدوائر المنطقية الترابطية (Combinational Logic Circuits)

جميع الدوائر المنطقية التي تعاملنا معها حتى الآن هي دوائر منطقية ترابطية (Combinational)، و سميت بالترابطية لأن وظيفة الدائرة تقتصر على ربط متغيرات الدخل بعمليات منطقية لتوليد متغيرات الخرج. و من الواضح أن الخرج في الدوائر الترابطية يعتمد فقط على القيم الحالية للدخل، فمتى ما تغيرت قيم الدخل تغيرت معها قيم الخرج. و النوع الآخر من الدوائر المنطقية هو الدوائر التتابعية (Sequential) التي سنقوم بدراستها في الوحدة الخامسة من هذا المقرر.
أما في هذه الوحدة فسنقوم بدراسة بعض الدوائر الترابطية الشائعة الاستخدام في الأنظمة الرقمية نظراً لقيامها بأداء وظائف مفيدة يتكرر ظهورها في تلك الأنظمة، و التي عادة ما تكون متوفرة تجارياً في صورة جاهزة في شكل دوائر متكاملة (IC's) بحيث يمكن شراؤها و استخدامها مباشرة في بناء الأنظمة الرقمية.

2- بعض استخدامات بوابة XOR

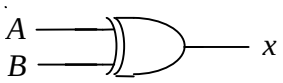
لبوابة XOR بعض الاستخدامات المفيدة في الدوائر المنطقية، و سنتعرض في هذا الجزء من المقررلثلاثة من أبرز تلك الاستخدامات.

في البداية نذكرك بأن عملية XOR تسمى عملية الاختلاف، حيث أن الخرج يساوي 1 إذا كان الدخلان مختلفين، و يساوي 0 إذا كانا متشابهين. و يرمز لها بالطريقة التالية

$$x = A \oplus B$$

PDF created with pdfFactory Pro trial version www.pdffactory.com

و في ما يلي شكل بوابة XOR و جدول الصواب لها



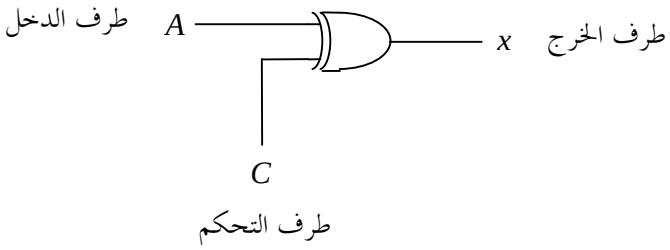
A	B	x
0	0	0
0	1	1
1	0	1
1	1	0

و كما نعلم فإنه يمكن التعبير عن عملية XOR باستخدام العمليات الأساسية الثلاث كالتالي

$$A \oplus B = \overline{A}B + A\overline{B}$$

2-1 العاكس المنطقي المحكوم (Controlled Logic Inverter)

يمكن استخدام بوابة XOR كعاكس منطقي محكوم (Controlled)، أي عاكس منطقي يمكننا أن نتحكم به بحيث يقوم بإجراء عملية العكس المنطقي للمتغير الداخل إليه أو لا يقوم بإجرائها، و ذلك حسب القيمة المنطقية التي نقوم بوضعها في طرف التحكم الخاص به. و يتم ذلك كالتالي



حيث قمنا باستخدام طرف الدخل الثاني لبوابة XOR كطرف تحكم (Control Line).

PDF created with pdfFactory Pro trial version www.pdffactory.com

و يمكن فهم طريقة عمل العاكس المنطقي المحكوم من جدول الصواب التالي

C	A	x
0	0	0
0	1	1
1	0	1
1	1	0

نلاحظ أنه في النصف الأعلى من جدول الصواب، عندما يكون طرف التحكم C مساوياً 0، يكون الخرج x مساوياً للدخل A ، أي أن القيمة الموضوعة في طرف الدخل تمر إلى الخرج كما هي. و في النصف الأسفل من جدول الصواب، عندما يكون طرف التحكم C مساوياً 1، يكون الخرج x مساوياً لمعكوس الدخل A ، أي أن القيمة الموضوعة في طرف الدخل يتم عكسها.

و باستخدام الجبر البولياني

$$x = C \oplus A$$
$$x = \overline{C}\, A + C\, \overline{A}$$

فعندما يكون $C = 0$ فإن

$$x = 1 \cdot A + 0 \cdot \overline{A} = A$$

و عندما يكون $C = 1$ فإن

$$x = 0 \cdot A + 1 \cdot \overline{A} = \overline{A}$$

أي أن العاكس المنطقي المحكوم لا يقوم بإجراء عملية العكس و يمرر القيمة الموضوعة في طرف الدخل كما هي إلى الخرج عندما نضع القيمة 0 في طرف التحكم الخاص به، و يقوم بعكس القيمة الموضوعة في طرف الدخل عندما نضع القيمة 1 في طرف التحكم الخاص به. و يمكن تلخيص ذلك في جدول الصواب التالي

C	x
0	A
1	$\overline{A}$

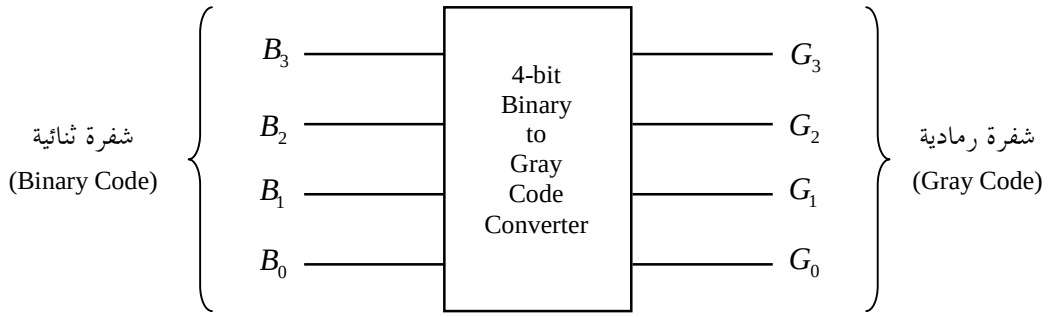
أي أن

$$x = \begin{cases} A & , C = 0 \\ \overline{A} & , C = 1 \end{cases}$$

PDF created with pdfFactory Pro trial version www.pdffactory.com

2-2 التحويل من الشفرة الثنائية إلى الشفرة الرمادية (Binary-to-Gray Code Conversion)

سبق و أن قمنا في الوحدة السابقة بتصميم دائرة تقوم بتحويل شفرة ثنائية مكونة من أربعة خانات إلى الشفرة الرمادية (4-bit Binary-to-Gray Code Converter)، و الموضح المخطط المنطقي لها أدناه



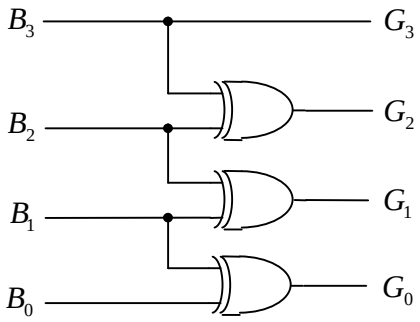
و حصلنا على التعبيرات المنطقية المختصرة التالية:

$$G_3 = B_3$$
$$G_2 = \overline{B_3}B_2 + B_3\overline{B_2}$$
$$G_1 = \overline{B_2}B_1 + B_2\overline{B_1}$$
$$G_0 = \overline{B_1}B_0 + B_1\overline{B_0}$$

من الواضح أنه يمكن كتابة التعبيرات المنطقية أعلاه بدلالة عملية XOR كالتالي:

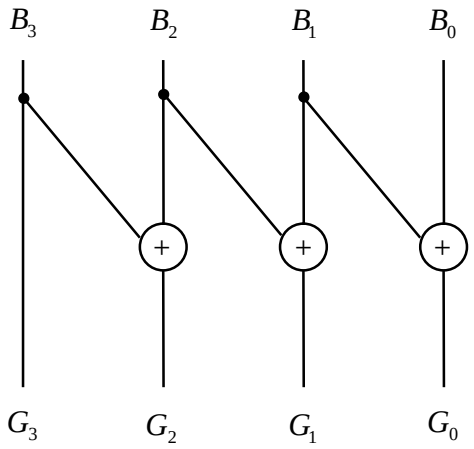
$$G_3 = B_3$$
$$G_2 = B_3 \oplus B_2$$
$$G_1 = B_2 \oplus B_1$$
$$G_0 = B_1 \oplus B_0$$

و عليه يمكن بناء الدائرة بالكامل باستخدام بوابات XOR كالتالي:



PDF created with pdfFactory Pro trial version www.pdffactory.com

الدائرة أعلاه توحي لنا بطريقة سهلة و مباشرة للتحويل من الشفرة الثنائية إلى الشفرة الرمادية، كم هو مبين أدناه

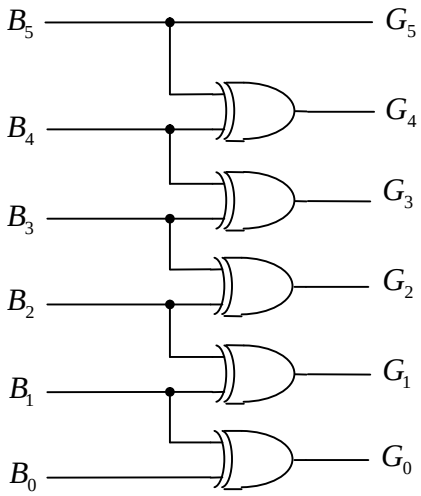


حيث نبدأ بالخانة العليا في الشفرة الرمادية G_3 و هي نفسها الخانة العليا في الشفرة الثنائية B_3 ، ثم ننتقل للخانة التالية في الشفرة الرمادية G_2 و نحصل عليها عن طريق جمع الخانة المقابلة لها في الشفرة الثنائية B_2 جمعاً ثنائياً مع الخانة الأعلى منها B_3 ، و الخانة التالية في الشفرة الرمادية G_1 نحصل عليها عن طريق جمع الخانة المقابلة لها في الشفرة الثنائية B_1 جمعاً ثنائياً مع الخانة الأعلى منها B_2 ، و الخانة G_0 نحصل عليها بجمع B_0 مع B_1 . أي أن كل خانة من خانات الشفرة الرمادية نحصل عليها عن طريق جمع الخانة الثنائية المقابلة لها جمعاً ثنائياً مع الخانة الثنائية الأعلى منها، ما عدا الخانة العليا في الشفرة الرمادية و التي نأخذها مباشرة كالخانة العليا في الشفرة الثنائية.

لاحظ أن عملية الجمع الثنائي التي استخدمناها أعلاه في التحويل من الشفرة الثنائية إلى الشفرة الرمادية هي نفسها عملية XOR.

و بناء على ما سبق يمكننا تعديل الدائرة المبنية من بوابات XOR أعلاه، و التي تقوم بتحويل شفرة ثنائية مكونة من أربعة خانات إلى الشفرة الرمادية، بحيث تقوم بتحويل شفرة ثنائية مكونة من أي عدد من الخانات إلى الشفرة الرمادية، مثلاً

PDF created with pdfFactory Pro trial version www.pdffactory.com



الدائرة أعلاه تقوم بتحويل شفرة ثنائية مكونة من ستة خانات إلى الشفرة الرمادية.

3-2 دوائر التحقق (Parity Checking)

عند نقل البيانات رقمياً عبر وسائل الاتصال المختلفة في شكل سلسلة من الرموز (Characters) الممثلة في الصورة الثنائية (Binary) قد تتعرض تلك البيانات لحدوث أخطاء (Errors). و يتمثل الخطأ هنا في تغير قيمة رقم ثنائي (bit) أو أكثر في أحد الرموز المُرسلة من 0 إلى 1 أو من 1 إلى 0. و عملية التحقق (Parity Checking) هي عملية تستخدم لاكتشاف حدوث خطأ في البيانات المنقولة، حيث تضاف لكل رمز خانة تسمى خانة التحقق (Parity bit)، و يتفق كل من الطرف المُرسِل للبيانات و الطرف المُستقبِل لها على أن يكون العدد الكلى للـــ 1's في أي رمز مُرسَل فردياً مثلاً، و هذا ما يسمى بالتحقق الفردي (Odd Parity Checking). و بناء على ذلك يقوم الطرف المُرسِل قبل إرسال أي رمز بحساب عدد الـــ 1's الموجودة فيه، فإذا وجد أن عددها فردي يقوم بوضع 0 في خانة التحقق، وذلك للحفاظ على العدد الكلي للـــ 1's في الرمز فردياً. أما إذا وجد أن عدد الـــ 1's في الرمز المُرسَل زوجي فإنه يقوم بوضع 1 في خانة التحقق، بحيث يصبح العدد الكلي للـــ 1's في الرمز فردياً. أي أن مهمة الطرف المُرسِل هي التأكد من أن عدد الـــ 1's فردي في كل رمز يقوم بإرساله، و ذلك بوضع القيمة المناسبة في خانة التحقق. أما بالنسبة للطرف المُستقبِل فإنه يقوم بحساب عدد الـــ 1's فى أي رمز يصل إليه، فإذا وجد أن عددها فردي كان معنى ذلك عدم حدوث خطأ أثناء عملية النقل، أما إذا وجد أن عددها زوجي فمعنى ذلك حدوث خطأ. و الطريقة الوحيدة الممكنة لتصحيح الخطأ الذي حدث هنا هي أن يطلب الطرف المُستقبِل من الطرف المُرسِل إعادة إرسال الرمز الذي وصله خاطئاً، وهذا يتطلب بالطبع وجود إمكانية الاتصال في الاتجاهين، و هو أمر غير متاح في كثير من الأحيان. لاحظ أن هذا الأسلوب فى اكتشاف حدوث الأخطاء يعجز عن اكتشاف حدوث خطأ فى خانتين فى وقت واحد، و لا توجد مشكلة هنا حيث أنه في أي نظام رقمي مصمم بصورة جيدة يكون احتمال حدوث خطأ في خانتين في وقت واحد أمراً

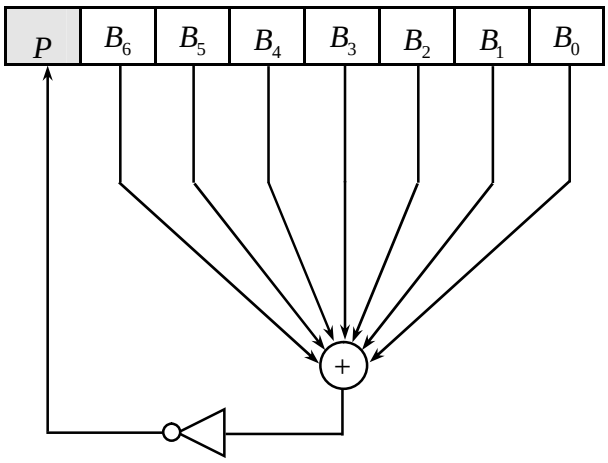
PDF created with pdfFactory Pro trial version www.pdffactory.com

نادر الحدوث بحيث يمكن تجاهله. يمكن أيضاً أن يتفق الطرفان المُرسِل والمُستقبِل على أن يكون العدد الكلى للـــ 1's في أي رمز مُرسَل زوجياً، و يسمى هذا بالتحقق الزوجي (Even Parity Checking).

سنقوم الآن بتصميم الدوائر المنطقية المستخدمة في حالة التحقق الفردي (Odd Parity Checking). حيث سنصمم الدائرة المنطقية التي يستخدمها الطرف المُرسِل في توليد القيمة التي توضع في خانة التحقق، و سنطلق عليها تسمية دائرة توليد خانة التحقق الفردي (Odd Parity bit Generator)، و الدائرة التي يستخدمها الطرف المُستقبِل لتحديد ما إذا كان هناك خطأ في الرمز الواصل إليه أم لا، و سنطلق عليها تسمية دائرة التحقق الفردي (Odd Parity Checker). و سنفترض أن الرموز ممثلة باستخدام شفرة ASCII، أي أن الرمز مكون من سبعة خانات و خانة التحقق هي الخانة الثامنة.

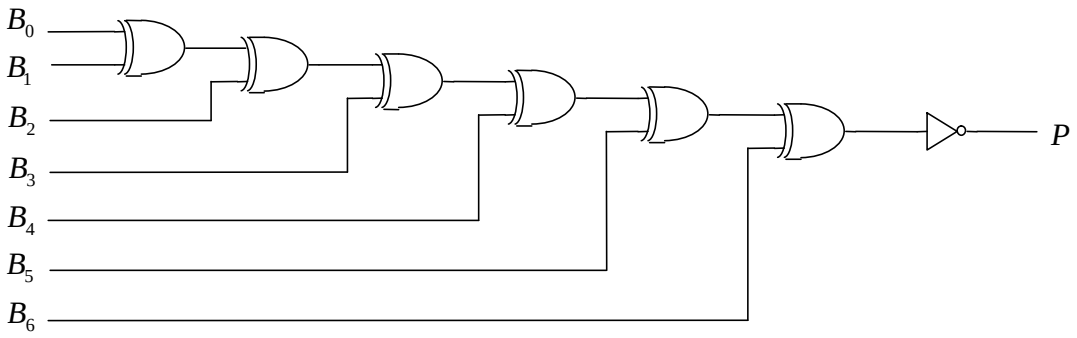
دائرة توليد خانة التحقق الفردي (Odd Parity bit Generator)

مهمة هذه الدائرة حساب عدد الـــ 1's في الخانات السبعة للرمز المُرسَل لتحديد ما إذا كان عددها فردياً أم لا، و تحديد القيمة المناسبة التي يجب وضعها في خانة التحقق بناء على ذلك. يتم ذلك بجمع الخانات السبعة جمعاً ثنائياً، أي إجراء عملية XOR في ما بينها، فإذا كان المجموع مساوياً 0 فمعنى ذلك أن عدد الـــ 1's في الخانات السبعة زوجي و نحتاج لوضع 1 في خانة التحقق، و إذا كان المجموع مساوياً 1 فمعنى ذلك أن عدد الـــ 1's في الخانات السبعة فردي و نحتاج لوضع 0 في خانة التحقق. أي أن القيمة التي توضع في خانة التحقق هي معكوس حاصل الجمع، كما هو موضح بالمخطط المنطقي التالي

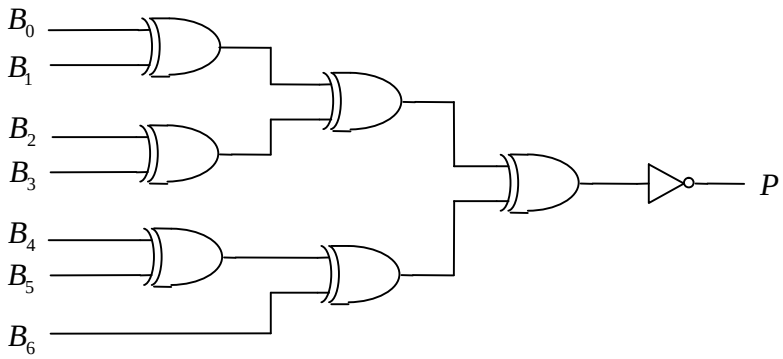


PDF created with pdfFactory Pro trial version www.pdffactory.com

نظراً لعدم توفر بوابة XOR بسبعة مداخل نستخدم في بناء الدائرة عدداً من بوابات XOR ذات المدخلين، كما هو موضح أدناه



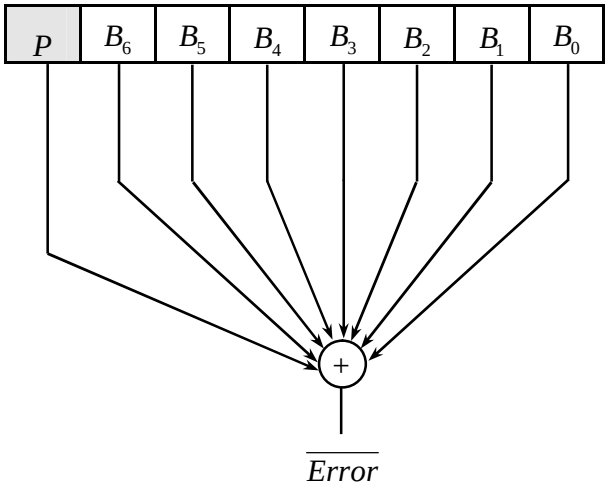
أو



دائرة التحقق الفردي (Odd Parity Checker)

مهمة هذه الدائرة حساب عدد الـــ 1's في خانات الرمز الذي تم استقباله، بما فيها خانة التحقق، لتحديد ما إذا كان عددها فردياً أو زوجياً، و تحديد ما إذا كان هنالك خطأ في الرمز أم لا بناء على ذلك. يتم ذلك بجمع الخانات الثمانية جمعاً ثنائياً، أي إجراء عملية XOR في ما بينها، فإذا كان المجموع مساوياً 0 فمعنى ذلك أن عدد الـــ 1's زوجي و هنالك خطأ في الرمز، أما إذا كان المجموع مساوياً 1 فمعنى ذلك أن عدد الـــ 1's فردي و لا يوجد خطأ في الرمز. و يمكن توضيح ذلك بالمخطط المنطقي التالي

PDF created with pdfFactory Pro trial version www.pdffactory.com



لاحظ أننا استخدمنا الرمز $\overline{Error}$ لخرج الدائرة، فإذا كان $\overline{Error}=0$ فإن $Error=1$ و هنالك خطأ في الرمز، أما إذا كان $\overline{Error}=1$ فإن $Error=0$ و لا يوجد خطأ.

و سنترك لك، عزيزي الدارس، بناء الدائرة باستخدام بوابات XOR كتدريب.

تدريب 1:

قم بتصميم كل من دائرة توليد خانة التحقق الزوجي (Even Parity bit Generator) و دائرة التحقق الزوجي (Even Parity Checker)، ثم قم ببنائهما باستخدام بوابات XOR.

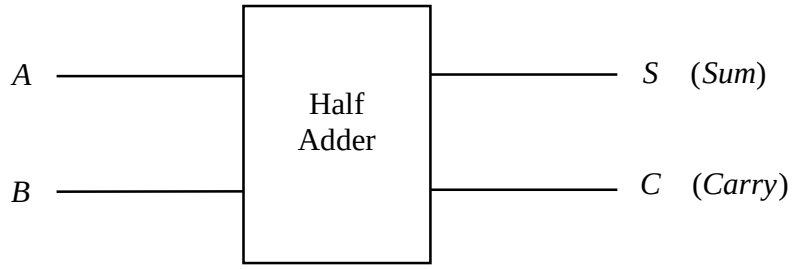
PDF created with pdfFactory Pro trial version www.pdffactory.com

3- دوائر الجمع (Adders)

دوائر الجمع أو الجوامع (Adders) هي دوائر منطقية تقوم بإجراء عملية جمع الأعداد الممثلة في الصورة الثنائية.

3-1 نصف الجامع (Half Adder)

نصف الجامع هو أبسط أنواع الجوامع، و هو عبارة عن دائرة منطقية تقوم بجمع خانتين ثنائيتين إلى بعضهما البعض و إيجاد حاصل الجمع (Sum) و الحمل (Carry)، كما هو موضح بالمخطط المنطقي و جدول الصواب التاليين



A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

و يمكن تصميم دائرة نصف الجامع بسهولة باستخدام طريقة التصميم التي درسناها في الوحدة السابقة.

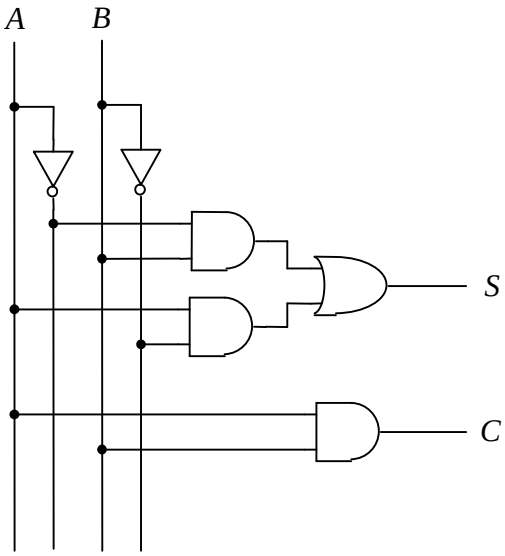
التعبيرات المنطقية:

$$S = \overline{A}B + A\overline{B} = A \oplus B$$
$$C = AB$$

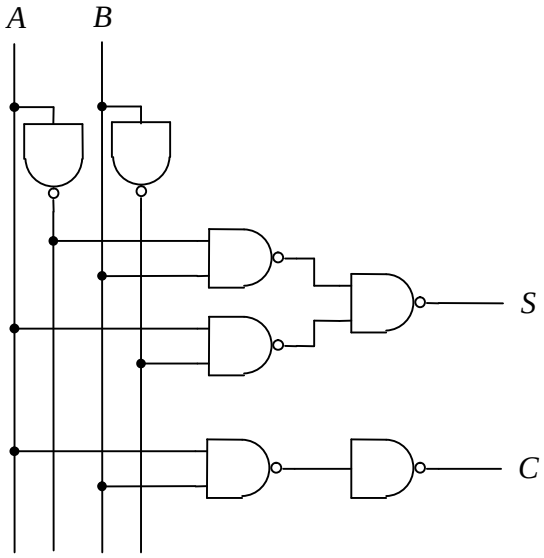
و التعبيرات هنا في أبسط صورة و لا تحتاج إلى تبسيط

PDF created with pdfFactory Pro trial version www.pdffactory.com

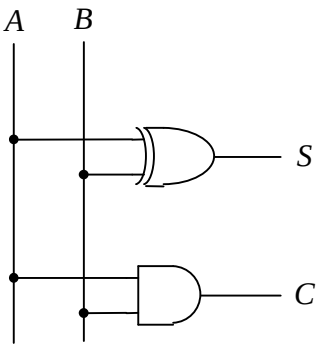
الدائرة المنطقية:



أو



أو



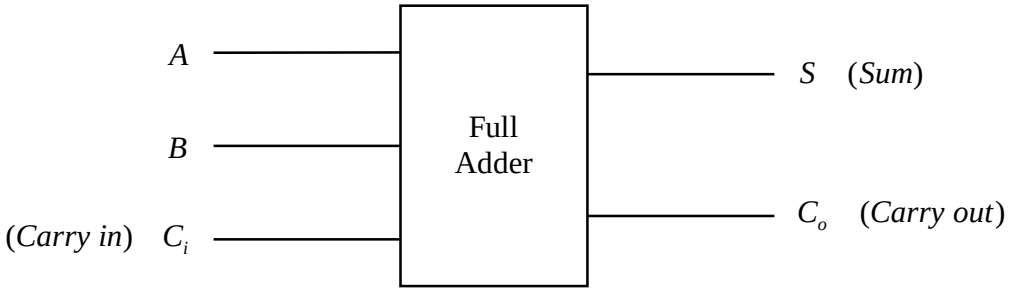
PDF created with pdfFactory Pro trial version www.pdffactory.com

تدريب 2:

استخدم بوابات NOR في بناء دائرة نصف الجامع (Half Adder).

2-3 الجامع الكامل (Full Adder)

تتشابه دائرة الجامع الكامل مع دائرة نصف الجامع في أنها تقوم بإجراء عملية الجمع و إيجاد كل من المجموع (Sum) و الحمل الخارج (Carry out)، إلا أن لها دخلاً ثالثاً هو عبارة عن حمل داخل (Carry in)، كما هو موضح بالمخطط المنطقي و جدول الصواب التاليين



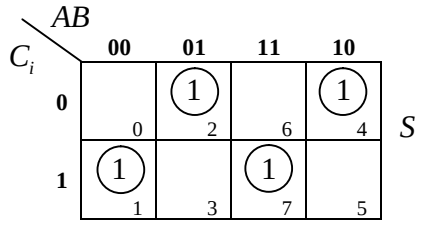
#	A	B	C_i	S	C_o
0	0	0	0	0	0
1	0	0	1	1	0
2	0	1	0	1	0
3	0	1	1	0	1
4	1	0	0	1	0
5	1	0	1	0	1
6	1	1	0	0	1
7	1	1	1	1	1

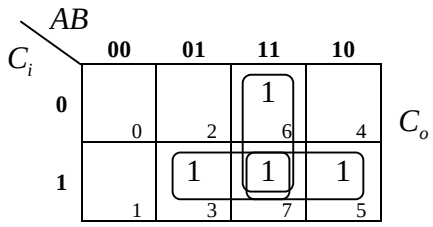
التعبيرات المنطقية:

$$S = \sum m\,(1,2,4,7)$$
$$C_o = \sum m\,(3,5,6,7)$$

PDF created with pdfFactory Pro trial version www.pdffactory.com

التعبيرات المنطقية المختصرة:





$$S = \overline{A}\overline{B}C_i + \overline{A}B\overline{C_i} + ABC_i + A\overline{B}\overline{C_i}$$
$$C_o = AB + BC_i + AC_i$$

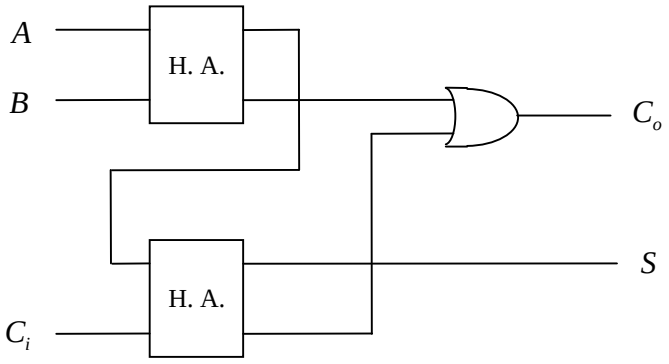
تدريب 3:

قم ببناء دائرة الجامع الكامل باستخدام:

(أ) البوابات الأساسية الثلاث (AND، OR، NOT).

(ب) بوابات NAND.

بناء الجامع الكامل باستخدام دائرتي نصف جامع



PDF created with pdfFactory Pro trial version www.pdffactory.com

لاحظ أننا قد استخدمنا هنا نصف الجامع الأول لجمع الخانتين A و B، ثم أدخلنا حاصل الجمع الناتج إلى نصف الجامع الثاني مع الخانة الثالثة C_i، فحصلنا على مجموع الخانات الثلاثة. أما الحمل الخارج C_o فإنه إما أن ينتج عن عملية الجمع الأولى أو عن عملية الجمع الثانية، لذلك ربطنا الحمل الخارج من دائرتي نصف الجامع بعملية OR.

3-3 الجامع المتعدد الخانات (Multi-bit Adder)

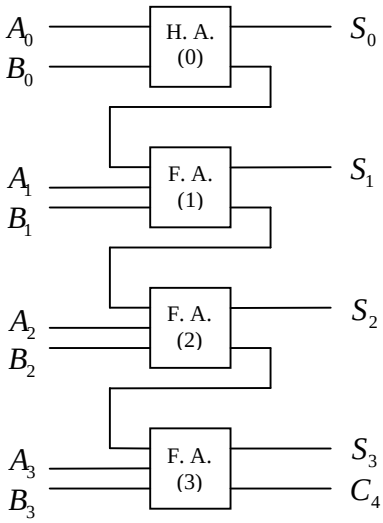
المطلوب الآن تصميم دائرة منطقية تقوم بجمع عددين ثنائيين يتكون كل منهما من أربعة خانات (4-bit Adder).

إذا أردنا إجراء عملية الجمع هذه يدوياً فإننا نبدأ بوضع العددين الثنائيين فوق بعضهما البعض، ثم نقوم بجمع كل خانة من العدد الأول مع الخانة المقابلة لها من العدد الثاني، مبتدئين بالخانة الدنيا (LSB)، مع ترحيل الحمل الخارج الناتج من خانة معينة إلى الخانة التي تليها كحمل داخل، كما هو مبين أدناه

$$\begin{array}{ccccc} & C_3 & C_2 & C_1 & \\ & A_3 & A_2 & A_1 & A_0 \\ & B_3 & B_2 & B_1 & B_0 \\ \hline C_4 & S_3 & S_2 & S_1 & S_0 \end{array}$$

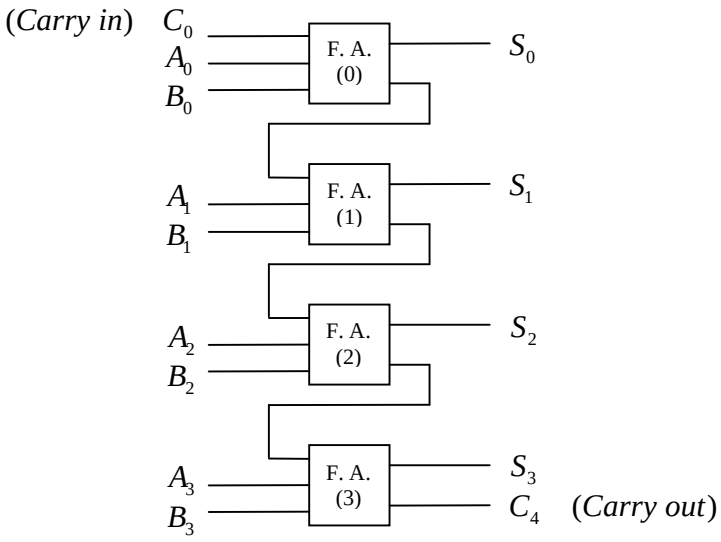
حيث جمعنا A_0 مع B_0 فحصلنا على المجموع S_0 و الحمل الخارج C_1 الذي قمنا بترحيله إلى الخانة التالية، ثم جمعنا C_1 مع A_1 و B_1 فحصلنا على المجموع S_1 و الحمل الخارج C_2 الذي قمنا بترحيله إلى الخانة التالية، ... و هكذا، حتى الخانة الأخيرة حيث جمعنا C_3 مع A_3 و B_3 فحصلنا على المجموع S_3 و الحمل الخارج C_4.

يمكننا استخدام نصف جامع لإجراء عملية الجمع في الخانة الدنيا (LSB)، و جامع كامل لإجراء عملية الجمع في كل خانة من الخانات التالية، مع مراعاة ترحيل الحمل من خانة إلى الخانة التي تليها، كما هو مبين أدناه

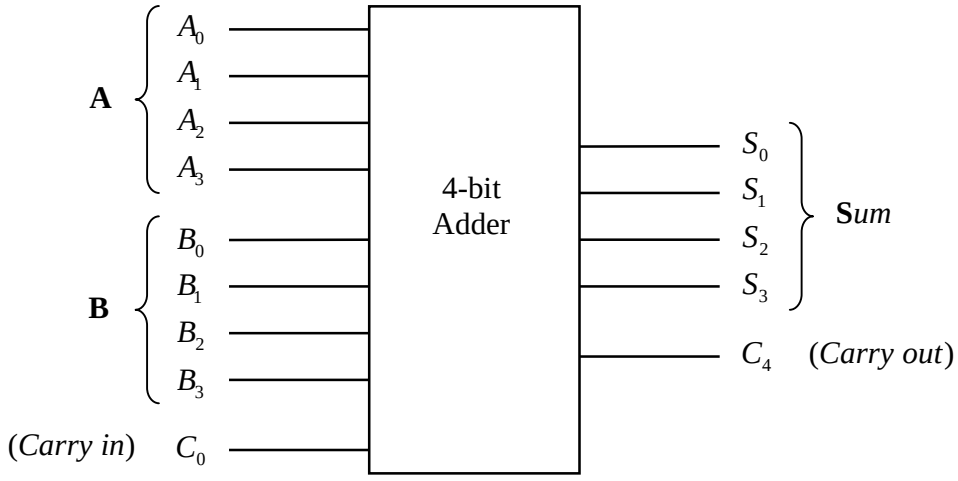


PDF created with pdfFactory Pro trial version www.pdffactory.com

عادة ما يستخدم جامع كامل بدلاً عن نصف الجامع في الخانة الدنيا (LSB)، الأمر الذي يسمح بوجود حمل داخل (Carry in) للجامع المتعدد الخانات. و يستخدم هذا الحمل الداخل في عمليات ربط الدوائر مع بعضها البعض و في إجراء عملية الطرح، كما سيتضح لاحقاً. و بذلك يكون الشكل النهائي لدائرة الجامع ذو الأربعة خانات هو



الشكل التالي يمثل المخطط المنطقي للجامع ذو الأربعة خانات



PDF created with pdfFactory Pro trial version www.pdffactory.com

لاحظ أن دائرة الجامع ذو الأربعة خانات (4-bit Adder) لها تسعة أطراف دخل، مما يجعل من تصميم هذه الدائرة باستخدام أسلوب التصميم الذي درسناه في الوحدة السابقة أمراً غاية في الصعوبة (لاحظ أن عدد أسطر جدول الصواب وحده هو $2^9 = 512$). تم حل هذه الإشكالية هنا بتقسيم الدائرة الكبيرة إلى عدد من الوحدات الصغيرة (جوامع كاملة) كل وحدة منها بعدد محدود من أطراف الدخل، و بعد تصميم الوحدة الصغيرة تم ربط الوحدات مع بعضها البعض بحيث تؤدي وظيفة الدائرة الكبيرة. و هذا الأسلوب في التصميم شائع الاستخدام في الأنظمة الرقمية حيث يتم تقسيم أي نظام رقمي معقد إلى عدد من الوحدات الأصغر، ثم تقسيم كل وحدة من هذه الوحدات بدورها إلى عدد من الوحدات الأصغر، ... و هكذا.

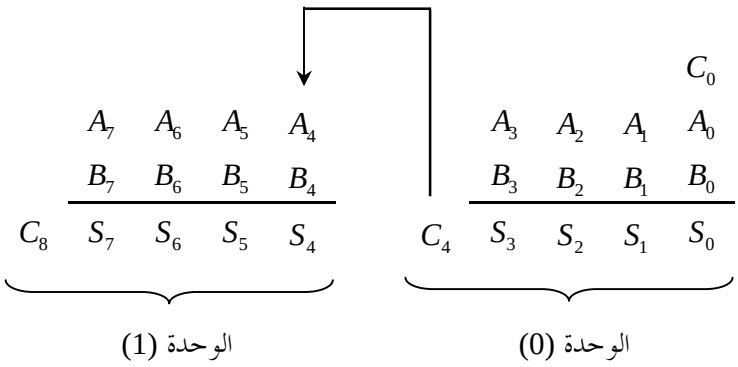
لاحظ أيضاً أنه يمكن بسهولة زيادة عدد خانات الجامع متعدد الخانات بزيادة عدد الجوامع الكاملة، بحيث نستطيع تصميم جامع بأي عدد من الخانات.

تدريب 4:

صمم جامعاً ذو ثمانية خانات (8-bit Adder).

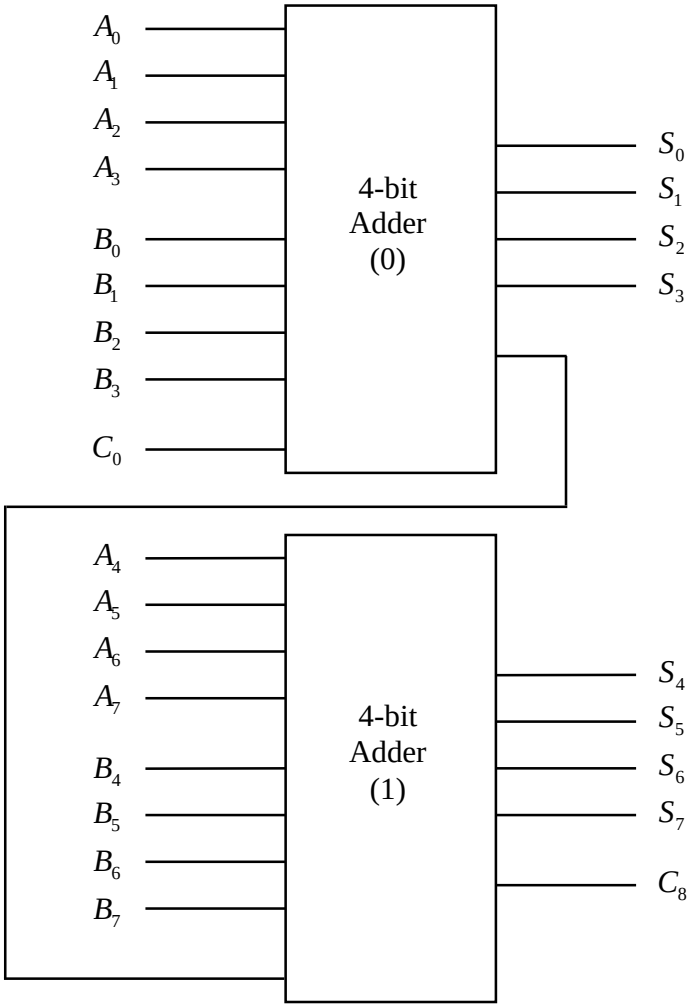
ربط الجوامع

يمكن ربطات وحدات جامع صغيرة لبناء جامع أكبر. مثلاً إذا قمنا بربط وحدتي جامع ذو أربعة خانات نحصل على جامع ذو ثمانية خانات، كما هو موضح أدناه



أي أننا يجب أن نقوم بترحيل الحمل الخارج (Carry out) من الوحدة الأولى و إدخاله كحمل داخل (Carry in) إلى الوحدة الثانية.

PDF created with pdfFactory Pro trial version www.pdffactory.com



تدريب 5:

وضح طريقة ربط 4 وحدات جامع ذو أربعة خانات (4-bit Adders) لبناء جامع ذو 16 خانة (16-bit Adder).

PDF created with pdfFactory Pro trial version www.pdffactory.com

عملية الطرح (Subtraction)

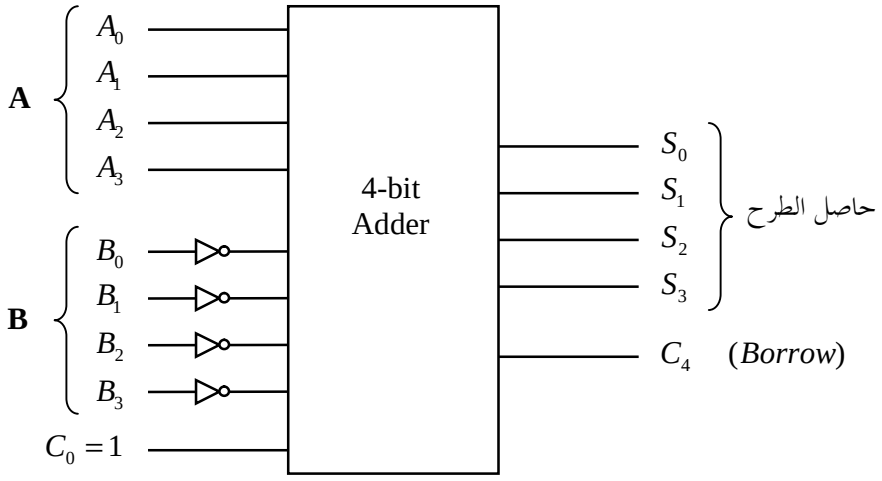
يتم تحويل عملية الطرح إلى عملية جمع مع سالب العدد المطروح كالتالي

$$A - B = A + (-B)$$

و سالب العدد B هو المكمل الثاني (2's Complement) له، و نحصل عليه بعكس جميع خانات العدد B ثم إضافة 1 إلى الخانة الدنيا (LSB). فإذا اعتبرنا أن كل من A و B عبارة عن عدد ثنائي ذو أربعة خانات فإن عملية الطرح تتم كالتالي

$$\begin{array}{cccccc} & & & & & 1 \\ & A_3 & A_2 & A_1 & A_0 & \\ & \overline{B_3} & \overline{B_2} & \overline{B_1} & \overline{B_0} & \\ \hline C_4 & S_3 & S_2 & S_1 & S_0 & \end{array}$$

و يتم إجراء العملية باستخدام الجامع ذو الأربعة خانات كالتالي

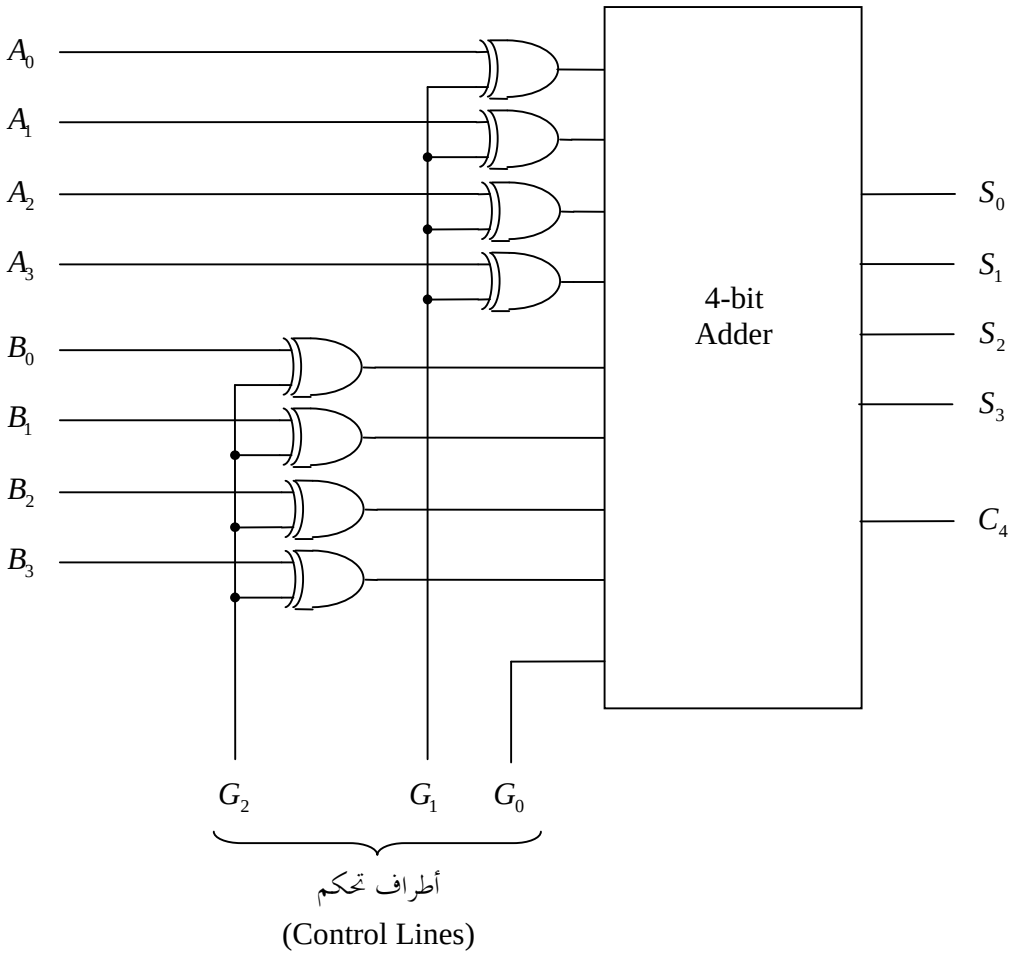


لاحظ أنه إذا كان الخرج C_4 مساوياً 1 فإن ذلك يدل على حدوث إستلاف (Borrow) من الخانة التي تلي الخانة العليا (MSB)، و يحدث هذا الإستلاف إذا كان العدد المطروح B أكبر من العدد المطروح منه A. أي أنه إذا كان C_4 مساوياً 1 فإن ذلك يدل على أن حاصل الطرح عبارة عن عدد سالب، أي أن C_4 يمثل إشارة حاصل الطرح.

PDF created with pdfFactory Pro trial version www.pdffactory.com

و لكن الدائرة بشكلها هذا لا تستطيع إجراء أي عملية بخلاف العملية $A-B$ ، حيث لا يمكن مثلاً إجراء عمليات مثل $B-A$ أو $A+B$.

لزيادة مرونة الدائرة يمكن أن نستخدم عواكس منطقية محكومة (Controlled Logic Inverters) بدلاً عن العواكس المنطقية العادية، بحيث نستطيع أن نقوم بإجراء عملية العكس المنطقي أو عدم إجرائها، حسب الحاجة. و ذلك كالتالي



تذكر أن العاكس المنطقي المحكوم يقوم بإجراء عملية العكس إذا وضعنا القيمة 1 في طرف التحكم الخاص به، و يمرر القيمة دون عكس إذا وضعنا القيمة 0 في طرف التحكم الخاص به.

و عليه يمكن إجراء عدد من العمليات الحسابية المختلفة بوضع القيم المناسبة في أطراف التحكم G_0 G_1 G_2. مثلاً

PDF created with pdfFactory Pro trial version www.pdffactory.com

- لإجراء العملية $A-B$ نحتاج لعكس خانات العدد B لإيجاد المكمل الأول له، لذلك نضع 1 في طرف التحكم G_2، كما نحتاج لوضع 1 في طرف التحكم G_0، الذي يمثل الحمل الداخل (Carry in) للجامع ذو الأربع خانات، و ذلك لإيجاد المكمل الثاني للـــ B. أما العدد A فلا نحتاج لعكس خاناته بل نريدها أن تمر كما هي لذلك نضع 0 في طرف التحكم G_1.
- لإجراء العملية $A+B$ لا نحتاج لعكس خانات أي من العددين A و B لذلك نضع 0 في طرفي التحكم G_1 و G_2. كما لا نحتاج لوضع 1 في الطرف G_0 لذلك نضع فيه 0 أيضاً.
- لإجراء العملية $B-A$ نحتاج لعكس خانات العدد A بوضع 1 في طرف التحكم G_1، كما نحتاج لوضع 1 في طرف التحكم G_0 لإيجاد المكمل الثاني للـــ A. أما العدد B فنريد أن تمر خاناته كما هي لذلك نضع 0 في طرف التحكم G_2.

و يمكن تلخيص ذلك في الجدول التالي

العملية (Operation)	إشارات التحكم		
	G_2	G_1	G_0
$A-B$	1	0	1
$A+B$	0	0	0
$B-A$	0	1	1

لاحظ أن عدد أطراف التحكم هو 3 و معنى ذلك أنه يوجد $2^3=8$ احتمالات مختلفة للقيم التي يمكن وضعها على هذه الأطراف. و قد أخذنا في الاعتبار ثلاثة فقط من هذه الإحتمالات في الجدول أعلاه. المطلوب الآن إنشاء جدول صواب يحتوي على جميع احتمالات الدخل لأطراف التحكم و العملية التي تقابل كل احتمال منها.

G_2	G_1	G_0	Operation
0	0	0	$A+B$
0	0	1	$A+B+1$
0	1	0	$B-A-1$
0	1	1	$B-A$
1	0	0	$A-B-1$
1	0	1	$A-B$
1	1	0	$-A-B-2$
1	1	1	$-A-B-1$

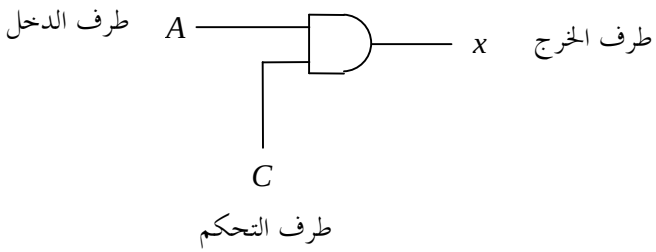
يمكننا الآن القول أن الدائرة السابقة عبارة عن دائرة جامع/طارح ذو أربعة خانات (4-bit Adder/Subtracter).

PDF created with pdfFactory Pro trial version www.pdffactory.com

وحدة الحساب (Arithmetic Unit)

يمكننا إجراء المزيد من العمليات الحسابية المفيدة باستخدام دائرة الجامع/الطارح ذو الأربعة خانات إذا أضفنا إليها إمكانية تصفير أحد العددين A أو B، أي التعويض عنه بصفر. مثلاً في العملية $A+B+1$ إذا عوضنا عن العدد B بصفر نحصل على العملية $A+1$، أي عملية Increment A، التي يرمز لها في بعض لغات البرمجة بالرمز $A++$. و في العملية $B-A-1$ إذا عوضنا عن العدد A بصفر نحصل على العملية $B-1$، أي عملية Decrement B، التي يرمز لها في بعض لغات البرمجة بالرمز $B--$. و في العملية $B-A$ إذا عوضنا عن العدد B بصفر نحصل على العملية $-A$، أي عملية Negate A.

يمكن إضافة إمكانية تصفير أحد العددين العددين A أو B إلى الدائرة باستخدام مجموعة من بوابات AND كبوابات تحكم، كالتالي



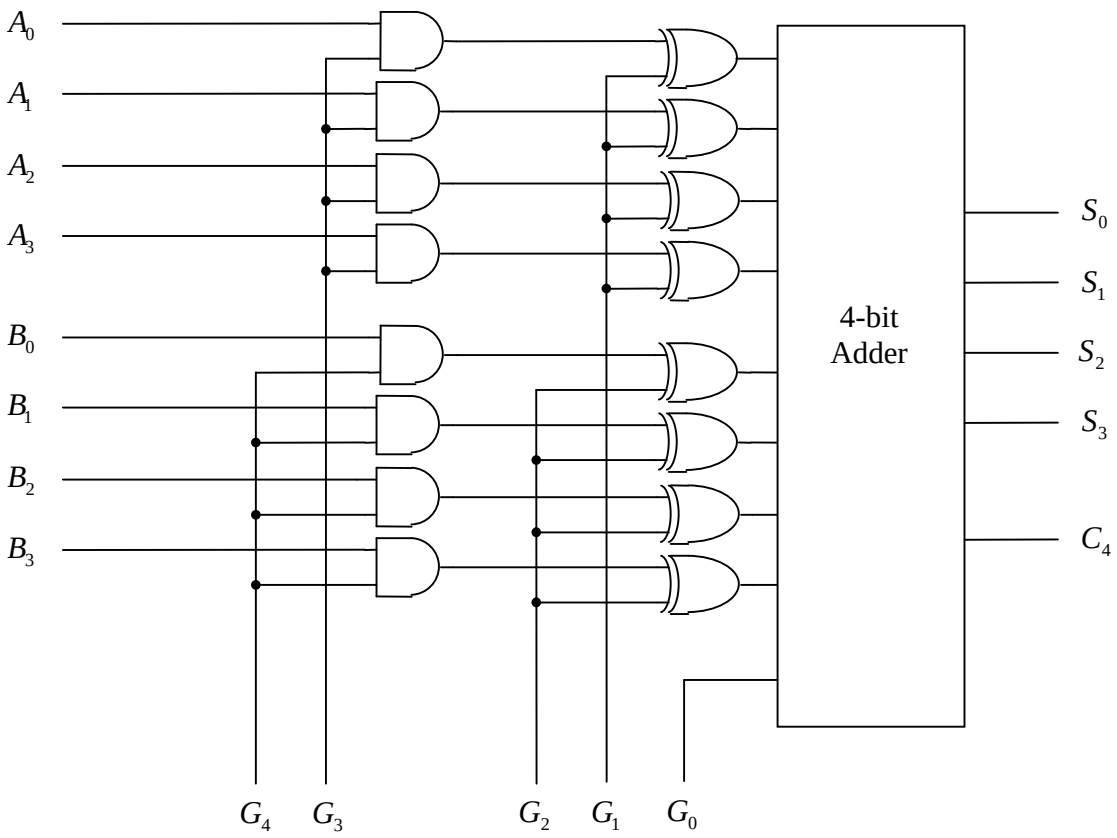
	C	A	x	
$C=0$	0	0	0	تصفير
	0	1	0	
$C=1$	1	0	0	تمرير
	1	1	1	

C	x
0	0
1	A

أي أنه عند وضع القيمة 1 في طرف التحكم لبوابة AND فإنها تمرر القيمة الموضوعة في طرف الدخل لها كما هي، و عند وضع القيمة 0 في طرف التحكم لبوابة AND فإنها تقوم بتصفير خرجها.

PDF created with pdfFactory Pro trial version www.pdffactory.com

يتم إضافة بوابات AND لدائرة الجامع/الطارح ذو الأربعة خانات كالتالي:



و الجدول التالي يوضح بعض العمليات التي يمكن إجراؤها باستخدام الدائرة المعدلة و إشارات التحكم اللازمة للقيام بكل عملية

العملية (Operation)	إشارات التحكم				
	تصفير		عكس		
	G_4	G_3	G_2	G_1	G_0
$A++$	0	1	0	0	1
$B--$	1	0	0	0	1
$-A$	0	1	0	1	1
$-B$	1	0	1	0	1
$\overline{A} = -A-1$	0	1	0	1	0
$\overline{B} = -B-1$	1	0	1	1	1

PDF created with pdfFactory Pro trial version www.pdffactory.com

الدائرة المعدلة تمثل وحدة حساب (Arithmetic Unit) ذات أربعة خانات، و إذا أضيف لها الجزء الخاص بإجراء العمليات المنطقية تصبح وحدة حساب و منطق (Arithmetic Logic Unit) أو ALU.

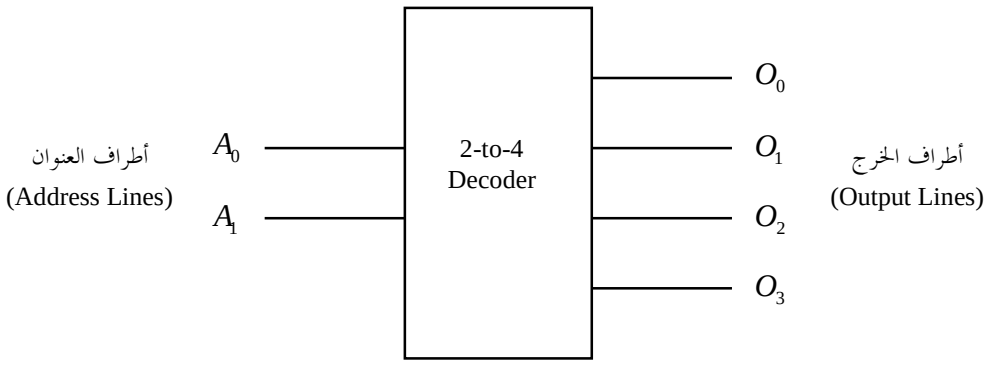
تدريب 6:

وضح في شكل جدول جميع العمليات الحسابية التي يمكن إجراؤها باستخدام وحدة الحساب أعلاه، و إشارات التحكم اللازمة للقيام بكل عملية.

4- فاك الشفرة (Decoder)

فاك الشفرة عبارة عن دائرة منطقية لها عدة أطراف خرج (Output Lines). واحد فقط من أطراف الخرج هذه يكون نشطاً (Active) أما بقية أطراف الخرج تكون غير نشطة. طرف الخرج النشط تظهر فيه القيمة المنطقية 1، أما بقية أطراف الخرج (غير النشطة) فتظهر في كل منها القيمة المنطقية 0. يتم اختيار طرف الخرج النشط بواسطة أطراف الدخل للدائرة و التي تسمى أطراف العنوان (Address Lines)، فلكل طرف من أطراف الخرج عنوان (Address) فريد يميزه، و هذا العنوان عبارة عن شفرة ثنائية (Binary Code) عندما توضع على أطراف العنوان ينشط طرف الخرج المقابل لذلك العنوان.

و في ما يلي المخطط المنطقي و جدول الصواب لفاك شفرة من نوع 2 إلى 4 (2-to-4 Decoder)



#	A_1	A_0	O_3	O_2	O_1	O_0
0	0	0	0	0	0	1
1	0	1	0	0	1	0
2	1	0	0	1	0	0
3	1	1	1	0	0	0

PDF created with pdfFactory Pro trial version www.pdffactory.com

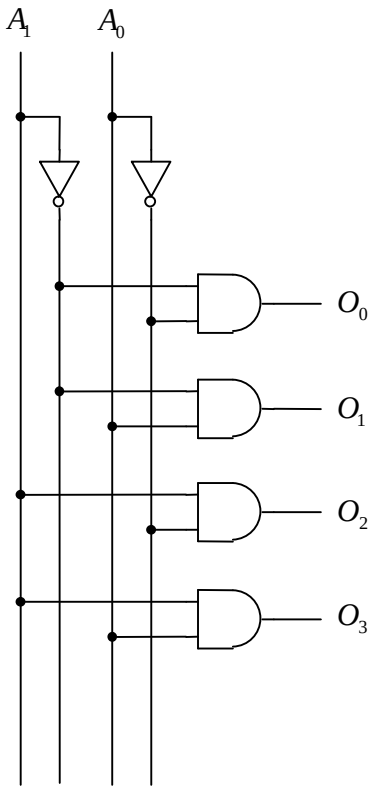
إذا أردنا تصميم دائرة فاك الشفرة من نوع 2 إلى 4 الموضح المخطط المنطقي و جدول الصواب لها أعلاه فما علينا إلا إتباع خطوات التصميم التي تدربنا عليها في الوحدة السابقة.

التعبيرات المنطقية:

$$O_0 = \overline{A_1}\,\overline{A_0} = m_0$$
$$O_1 = \overline{A_1} A_0 = m_1$$
$$O_2 = A_1 \overline{A_0} = m_2$$
$$O_3 = A_1 A_0 = m_3$$

نلاحظ أن فاك الشفرة يقوم بتوليد الحدود الصغرى (minterms) لمتغيرات الدخل في أطراف الخرج له.

الدائرة المنطقية:



نلاحظ أن الدائرة المنطقية لفاك الشفرة تتكون أساساً من مجموعة من بوابات AND بعدد أطراف الخرج.

PDF created with pdfFactory Pro trial version www.pdffactory.com

لاحظ وجود علاقة ما بين عدد أطراف الخرج (Output Lines) و عدد أطراف العنوان (Address Lines) لفاك الشفرة، فإذا كان عدد أطراف العنوان هو N فمعنى ذلك أن عدد العناوين الممكنة هو 2^N ، و بالتالي فإن عدد أطراف الخرج يجب أن يكون أقل من أو مساوياً 2^N ، بحيث يكون لكل طرف من أطراف الخرج عنوان فريد يميزه.

لاحظ أيضاً الأسلوب المتبع في تسمية فاك الشفرة، حيث نذكر في الإسم عدد أطراف الدخل ثم عدد أطراف الخرج و بينهما كلمة إلى (to).

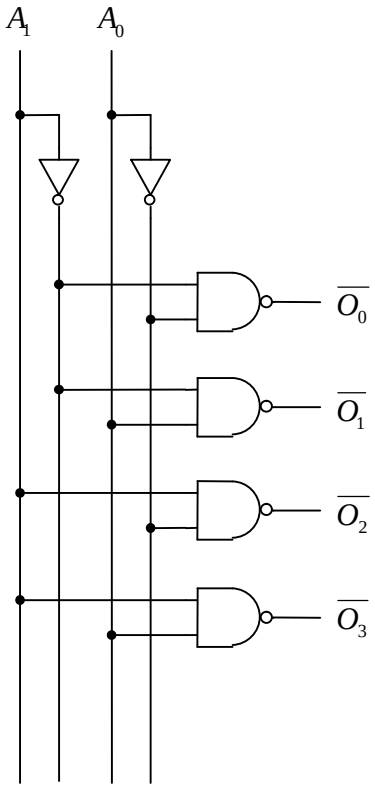
تدريب 7:

وضح المخطط المنطقي و جدول الصواب، ثم اكتب التعبيرات المنطقية و ارسم الدائرة المنطقية لـــ:

(أ) فاك شفرة من نوع 1 إلى 2 (1-to-2 Decoder).

(ب) فاك شفرة من نوع 3 إلى 8 (3-to-8 Decoder).

أحياناً يتم استبدال بوابات AND في دائرة فاك الشفرة ببوابات NAND، كما هو موضح في الشكل التالي لفاك شفرة من نوع 2 إلى 4 (2-to-4 Decoder)



PDF created with pdfFactory Pro trial version www.pdffactory.com

في هذه الحالة يكون الخرج معكوساً، و بالتالي فإن طرف الخرج النشط تظهر فيه القيمة المنطقية 0، و أطراف الخرج الأخرى (غير النشطة) تظهر في كل منها القيمة المنطقية 1. و نقول في مثل هذه الحالة أن فاك الشفرة ذو خرج نشط منخفض (Active Low Outputs). و كثيراً ما يستخدم مصطلحي منخفض (Low) و مرتفع (High) للإشارة إلى حالة أطراف الخرج (أو أطراف الدخل) في الدوائر المنطقية، لأنه عادة ما يتم تمثيل القيمة المنطقية 0 في تلك الدوائر بجهد كهربائي منخفض (مثلاً 0 V)، و القيمة المنطقية 1 بجهد كهربائي مرتفع (مثلاً 5+ V).

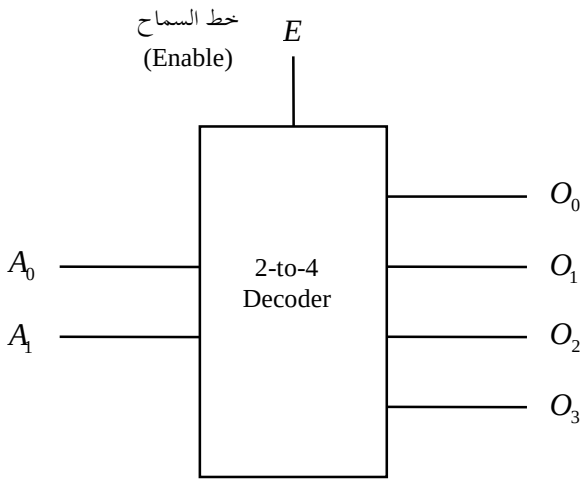
تدريب 8:

وضح المخطط المنطقي و جدول الصواب، ثم اكتب التعبيرات المنطقية لفاك الشفرة من نوع 2 إلى 4 بخرج نشط منخفض (2-to-4 Decoder with Active Low Outputs) الموضح الدائرة المنطقية له أعلاه.

خط السماح (Enable)

عادة ما يكون فاك الشفرة مزوداً بخط سماح (Enable). و خط السماح، في الدوائر المنطقية بصورة عامة، هو عبارة عن طرف تحكم يمكن بواسطته أن نبطل عمل الدائرة، أو نسمح لها بالعمل كالمعتاد.

و في ما يلي المخطط المنطقي لفاك شفرة من نوع 2 إلى 4 مزود بخط سماح (2-to-4 Decoder with Enable)



عند وضع القيمة المنطقية 0 في خط السماح E يبطل عمل فاك الشفرة فلا يستجيب للقيم الموضوعة في أطراف العنوان و تكون جميع أطراف الخرج له غير نشطة، أما عند وضع القيمة المنطقية 1 في خط السماح E فإن فاك الشفرة يعمل كالمعتاد. و يمكن توضيح ذلك بجدول الصواب التالي

PDF created with pdfFactory Pro trial version www.pdffactory.com

E	A_1	A_0	O_3	O_2	O_1	O_0
0	0	0	0	0	0	0
0	0	1	0	0	0	0
0	1	0	0	0	0	0
0	1	1	0	0	0	0
1	0	0	0	0	0	1
1	0	1	0	0	1	0
1	1	0	0	1	0	0
1	1	1	1	0	0	0

و يمكن كتابة جدول الصواب بصورة مختصرة كالتالي

E	A_1	A_0	O_3	O_2	O_1	O_0
0	×	×	0	0	0	0
1	0	0	0	0	0	1
1	0	1	0	0	1	0
1	1	0	0	1	0	0
1	1	1	1	0	0	0

السطر الأول من جدول الصواب هنا يعني أنه طالما كان خط السماح $E = 0$ فإنه بغض النظر عن قيم طرفي العنوان A_1 A_0 تكون جميع أطراف الخرج لفاك الشفرة غير نشطة. فاستخدامنا لرمز القيم غير المحددة × هنا مكننا من دمج أربعة أسطر من جدول الصواب في سطر واحد نظراً لتشابه قيم الخرج في هذه الأسطر الأربعة.

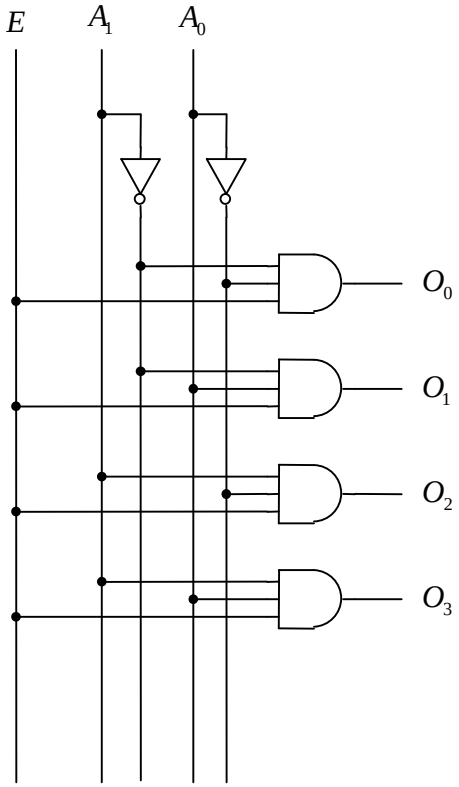
و يمكن بسهولة تصميم دائرة فاك الشفرة من نوع 2 إلى 4 بخط سماح (2-to-4 Decoder with Enable) كالتالي

التعبيرات المنطقية:

$$O_0 = E\overline{A_1}\,\overline{A_0}$$
$$O_1 = E\overline{A_1}A_0$$
$$O_2 = EA_1\overline{A_0}$$
$$O_3 = EA_1A_0$$

PDF created with pdfFactory Pro trial version www.pdffactory.com

الدائرة المنطقية:



خط السماح (Enable) يمكن أن يكون نشطاً منخفضاً (Active Low) أيضاً، و يرمز له في هذه الحالة بالرمز $\overline{E}$ ، و يسمح للدائرة بالعمل عندما توضع فيه القيمة المنطقية 0، و يبطل عملها عندما توضع فيه القيمة المنطقية 1.

تدريب 9:

وضح المخطط المنطقي و جدول الصواب، ثم اكتب التعبيرات المنطقية و ارسم الدائرة المنطقية لفاك شفرة من نوع:

(أ) 3 إلى 8 بخط سماح (3-to-8 Decoder with Enable).

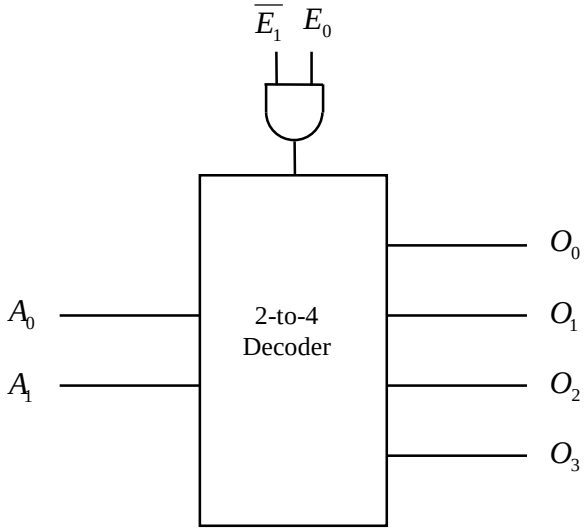
(ب) 3 إلى 8 بخط سماح و خرج نشط منخفض (3-to-8 Decoder with Enable and Active Low Outputs).

(ج) 3 إلى 8 بخط سماح نشط منخفض (3-to-8 Decoder with Active Low Enable).

PDF created with pdfFactory Pro trial version www.pdffactory.com

خطوط السماح المتعددة (Multiple Enables)

في بعض الأحيان قد يكون لدائرة ما أكثر من خط سماح واحد، ترتبط مع بعضها البعض بعمليات منطقية، و تعمل معاً على إبطال عمل الدائرة أو السماح لها بالعمل. مثلاً



خطا السماح E_0 و E_1 هنا مرتبطان بعملية AND، و شرط عمل الدائرة هنا هو أن يكون $\overline{E_1}E_0 = 1$، أي أن يكون $E_0 = 1$ و $E_1 = 0$.

تدريب 10:

وضح جدول الصواب، ثم اكتب التعبيرات المنطقية و ارسم الدائرة المنطقية لفاك الشفرة من نوع 2 إلى 4 بخطي سماح الموضح المخطط المنطقي له أعلاه.

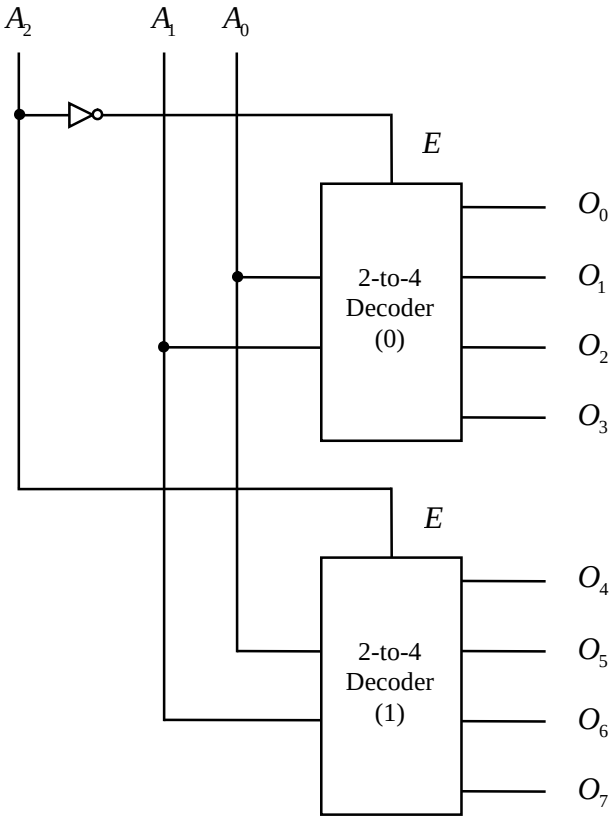
الاستخدام الأساسي لفاك الشفرة

لفاك الشفرة استخدامات عديدة، إلا أن أهم تلك الاستخدامات هو استخدامه في دوائر الذاكرة (Memory)، بأنواعها المختلفة، للوصول إلى موقع معين من مواقع الذاكرة عن طريق عنوانه. فلكل موقع من مواقع الذاكرة عنوان (Address) خاص به، و للوصول إلى ذلك الموقع يتم وضع عنوانه على أطراف العنوان لفاك الشفرة، فينشط طرف الخرج في فاك الشفرة المتصل بذلك الموقع و يقوم بفتح الموقع لعمليات القراءة (Read) أو الكتابة (Write). أي أن مهمة فاك الشفرة هي الربط ما بين مواقع الذاكرة و عناوينها.

PDF created with pdfFactory Pro trial version www.pdffactory.com

ربط دوائر فاك الشفرة

يمكن أن يتم ربط عدد من الوحدات الصغيرة من دوائر فاك الشفرة لبناء وحدة كبيرة. مثلاً، يمكن ربط وحدتي فاك شفرة من نوع 2 إلى 4 لبناء فاك شفرة من نوع 3 إلى 8، كما هو موضح أدناه



نلاحظ أن وحدات فاك الشفرة المطلوب ربطها يجب أن تكون مزودة بخط سماح (Enable).

من جدول الصواب لفاك الشفرة من نوع 3 إلى 8 الموضح أدناه يمكن توضيح طريقة الربط

A_2	A_1	A_0	O	
0	0	0	0	Decoder (0)
0	0	1	1	
0	1	0	2	
0	1	1	3	
1	0	0	4	Decoder (1)
1	0	1	5	
1	1	0	6	
1	1	1	7	

PDF created with pdfFactory Pro trial version www.pdffactory.com

قمنا هنا بتقسيم جدول الصواب إلى نصفين، النصف الأعلى يقابل الوحدة الأولى (0)، و النصف الأسفل يقابل الوحدة الثانية (1). و من الجدول يمكن أن نلاحظ الآتي

1. أطراف الخرج للوحدة الكبيرة، و عددها هنا هو 8، موزعة بالتساوي ما بين الوحدات الصغيرة. مع ضرورة ترقيم الوحدات و مراعاة الترتيب.
2. أطراف العنوان الدنيا، و هي أطراف العنوان التي تظهر في كل وحدة من الوحدات الصغيرة المطلوب ربطها، و هي هنا عبارة عن الطرفين A_0 و A_1، تكون مشتركة. و السبب في ذلك إن قيم هذه الأطراف تكون متشابهة في نصفي جدول الصواب الأعلى و الأسفل.
3. طرف العنوان الأعلى A_2 يستخدم في اختيار الوحدة النشطة (Active Unit) من بين الوحدات المربوطة مع بعضها البعض، و ذلك عن طريق خطوط السماح (Enable) لتلك الوحدات.

لاحظ أنه عند إدخال أي عنوان من عناوين النصف الأعلى من جدول الصواب، و فيها جميعاً طرف العنوان الأعلى $A_2 = 0$، تنشط الوحدة الأولى (0) لأن القيمة التي تظهر في خط السماح لها هي $\overline{A_2} = 1$، في حين تكون الوحدة الثانية (1) غير نشطة لأن القيمة التي تظهر في خط السماح لها هي $A_2 = 0$. و بما أن الوحدة الأولى (0) نشطة فإنها تستجيب للقيم الموضوعة في أطراف العنوان الخاصة بها، و هي أطراف العنوان الدنيا A_0 و A_1، و ينشط أحد أطراف الخرج لها (و هي أطراف الخرج الأربعة الأولى) بناء على ذلك. أما الوحدة الثانية (1)، غير النشطة، فلا تستجيب لأطراف العنوان و تكون جميع أطراف الخرج لها (و هي أطراف الخرج الأربعة الأخيرة) غير نشطة. و عند إدخال أي عنوان من عناوين النصف الأسفل من جدول الصواب، و فيها جميعاً طرف العنوان الأعلى $A_2 = 1$، يحدث العكس، حيث تنشط الوحدة الثانية (1) و تستجيب لأطراف العنوان الدنيا A_0 و A_1، و ينشط أحد أطراف الخرج لها بناء على ذلك، في حين تكون الوحدة الأولى (0) غير نشطة و لا تستجيب لأطراف العنوان و تكون جميع أطراف الخرج لها غير نشطة.

PDF created with pdfFactory Pro trial version www.pdffactory.com

مثال:

وضح طريقة ربط وحدات فاك شفرة من نوع 2 إلى 4 لبناء فاك شفرة من نوع 4 إلى 16.

الحل:

- الخطوة الأولى هنا هي تحديد عدد الوحدات الصغيرة التي نحتاج إليها في البناء، و يتم ذلك بملاحظة عدد أطراف الخرج للوحدة الصغيرة و عدد أطراف الخرج للوحدة الكبيرة المطلوب بناءها. عدد أطراف الخرج للوحدة الصغيرة هنا هو 4، و عدد أطراف الخرج للوحدة الكبيرة هو 16، أي أننا نحتاج إلى أربعة من الوحدات الصغيرة، أي أربعة وحدات فاك شفرة من نوع 2 إلى 4.

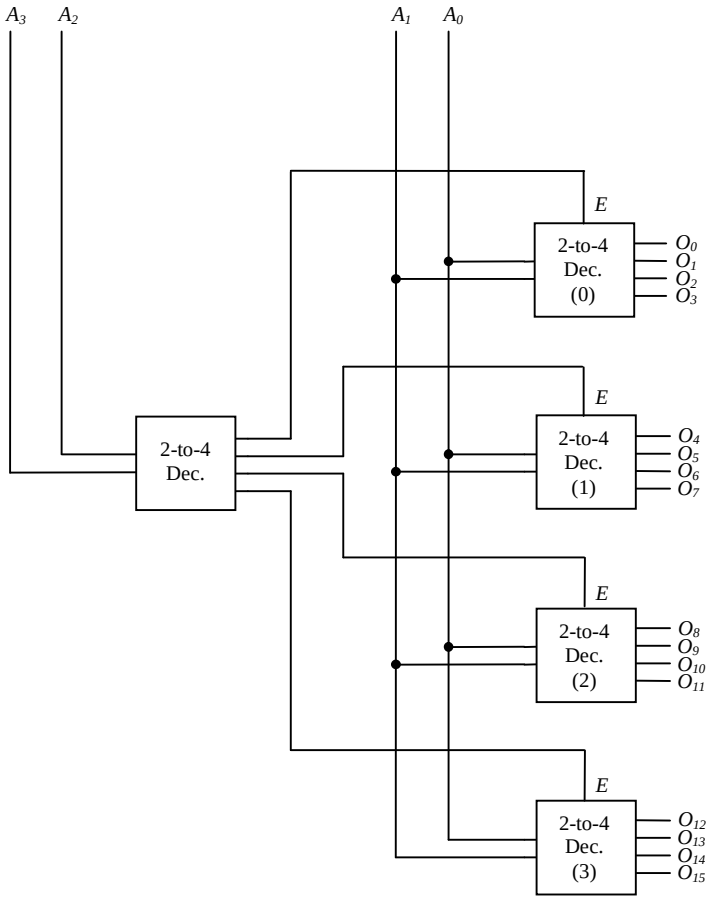
- الخطوة الثانية هي جدول الصواب للوحدة الكبيرة، أي جدول الصواب لفاك شفرة من نوع 4 إلى 16

A_3	A_2	A_1	A_0	O	
0	0	0	0	0	Decoder (0)
0	0	0	1	1	
0	0	1	0	2	
0	0	1	1	3	
0	1	0	0	4	Decoder (1)
0	1	0	1	5	
0	1	1	0	6	
0	1	1	1	7	
1	0	0	0	8	Decoder (2)
1	0	0	1	9	
1	0	1	0	10	
1	0	1	1	11	
1	1	0	0	12	Decoder (3)
1	1	0	1	13	
1	1	1	0	14	
1	1	1	1	15	

- الخطوة الثالثة هي عملية الربط:

1. أطراف الخرج للوحدة الكبيرة موزعة بالتساوي ما بين الوحدات الصغيرة.
2. أطراف العنوان الدنيا A_0 و A_1 مشتركة.
3. أطراف العنوان العليا A_2 و A_3 تستخدم في اختيار الوحدة النشطة. و نستعين في اختيار الوحدة النشطة هنا بفاك شفرة من نوع 2 إلى 4.

PDF created with pdfFactory Pro trial version www.pdffactory.com



تدريب تقويم ذاتي:

تحقق من صحة عمل الدائرة أعلاه، و ذلك بوضع عدد من العناوين المختلفة على أطراف العنوان للدائرة و إيجاد الطرف الذي ينشط، و ذلك للتأكد من أنه فعلاً الطرف صاحب العنوان الموضوع.

ملاحظة:

عندما قمنا بربط دائرتي فاك شفرة من نوع 2 إلى 4 لبناء فاك شفرة من نوع 3 إلى 8 استخدمنا في اختيار الوحدة النشطة عاكساً منطقياً، و بالطبع فإن في إمكاننا أن نستخدم في اختيار الوحدة النشطة فاك شفرة من نوع 1 إلى 2. و في واقع الأمر فإن فاك الشفرة من نوع 1 إلى 2 (بدون خط سماح (Enable)) تتكون دائرته المنطقية من عاكس منطقي واحد فقط. (إرجع إلى تدريب 7 (أ))

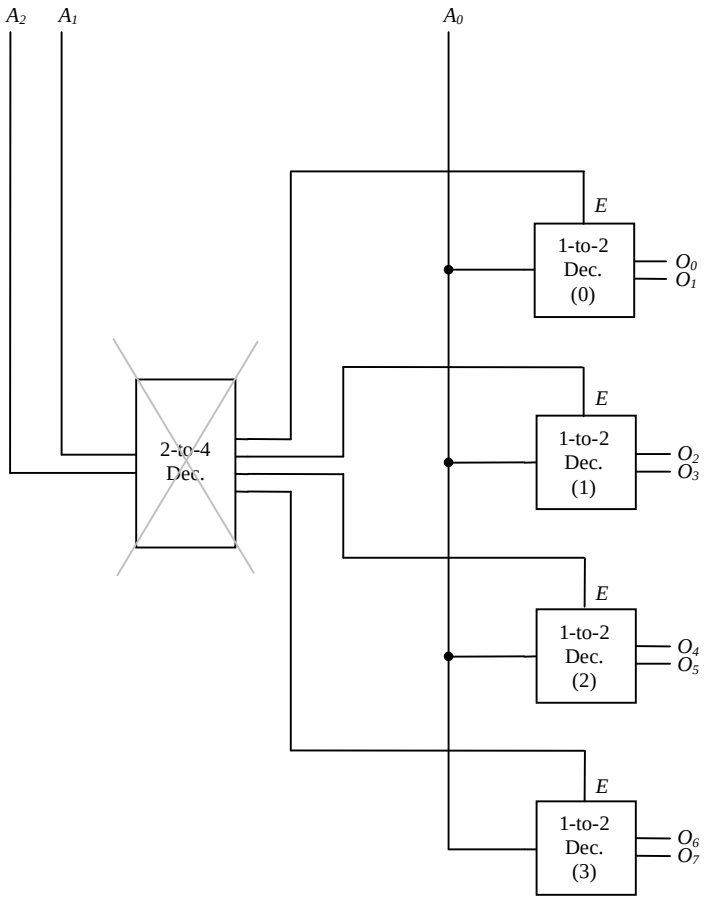
PDF created with pdfFactory Pro trial version www.pdffactory.com

مثال:

وضح طريقة بناء فاك شفرة من نوع 3 إلى 8 باستخدام وحدات فاك شفرة من نوع 1 إلى 2.

الحل:

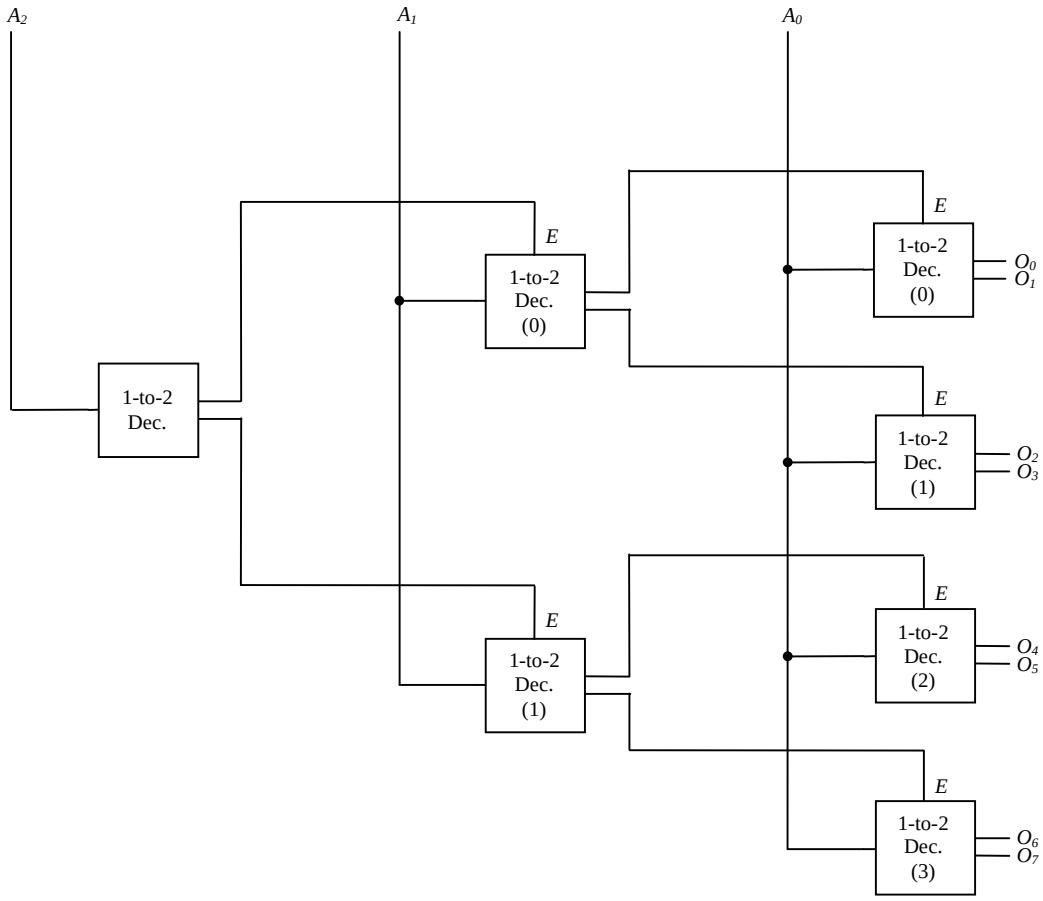
الحل التالي (الذي يشبه إلى حد كبير حل المثال السابق) هو أول ما يتبادر إلى الذهن



و لكن هذا الحل ليس هو بالحل الصحيح، لأنه إذا قرأنا نص المسألة بدقة نجد أن المطلوب استخدام دوائر فاك شفرة من نوع 1 إلى 2 فقط في البناء، و عليه فمن غير المسموح لنا استخدام فاك الشفرة من نوع 2 إلى 4 الذي نحتاج إليه في عملية الربط. و حل هذه الإشكالية بسيط، حيث نقوم ببناء فاك الشفرة من نوع 2 إلى 4 نفسه باستخدام دوائر فاك شفرة من نوع 1 إلى 2.

PDF created with pdfFactory Pro trial version www.pdffactory.com

و عليه فإن الحل الصحيح للمسألة هو



تدريب 11:

وضح طريقة بناء فاك شفرة من نوع 4 إلى 16 (4-to-16 Decoder) و ذلك باستخدام دوائر فاك شفرة من نوع:

(أ) 3 إلى 8 (3-to-8 Decoders).

(ب) 2 إلى 4 (2-to-4 Decoders).

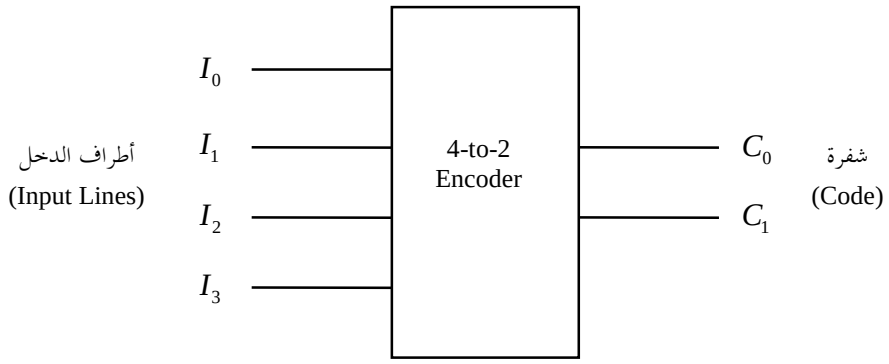
(ج) 1 إلى 2 (1-to-2 Decoders).

PDF created with pdfFactory Pro trial version www.pdffactory.com

5- المشفر (Encoder)

كما هو واضح من التسمية فإن المشفر (Encoder) يؤدي عكس الوظيفة التي يؤديها فاك الشفرة (Decoder).
حيث أن المشفر عبارة عن دائرة منطقية لها عدة أطراف دخل (Input Lines)، و يكون واحد فقط من أطراف الدخل هذه نشطاً (Active)، أي مساوياً 1، أما بقية أطراف الدخل تكون غير نشطة، أي مساوية 0. خرج الدائرة عبارة عن شفرة (Code) تمثل طرف الدخل النشط.

و في ما يلي المخطط المنطقي و جدول الصواب لمشفر من نوع 4 إلى 2 (4-to-2 Encoder)



I_3	I_2	I_1	I_0	C_1	C_0
0	0	0	1	0	0
0	0	1	0	0	1
0	1	0	0	1	0
1	0	0	0	1	1

لاحظ أن جدول الصواب الموضح أعلاه هو جدول صواب مختصر، تظهر فيه احتمالات الدخل الواردة فقط، و عددها أربعة، حيث أن طريقة عمل المشفر تشترط أن يكون طرف واحد فقط من أطراف الدخل نشطاً. أما جدول الصواب الكامل فيحتوي على 16 احتمال دخل، الخرج المقابل للـ 12 احتمال دخل غير الواردة منها عبارة عن قيم غير محددة (Don't Cares).

التعبيرات المنطقية:

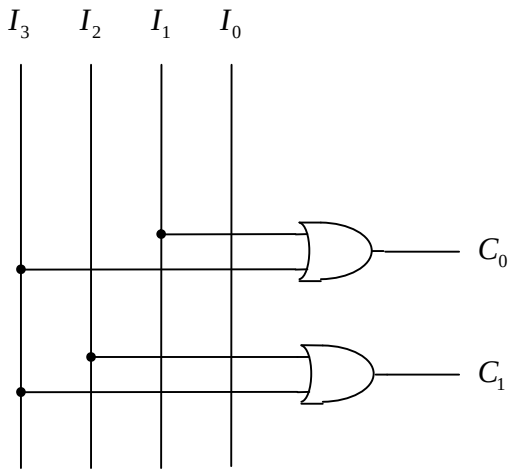
سنقوم هنا بكتابة التعبيرات المنطقية المختصرة مباشرة من جدول الصواب المختصر بأسلوب غير تقليدي. فلكتابة التعبير المختصر لمتغير معين من متغيرات الخرج نبحث أسفل ذلك المتغير في جدول الصواب عن الـ 1's، و لكل 1 نجده أسفل متغير الخرج نقوم بأخذ متغير الدخل الذي يساوي 1 في نفس السطر من جدول الصواب، ثم نربط متغيرات الدخل هذه مع بعضها البعض بعمليات OR.

PDF created with pdfFactory Pro trial version www.pdffactory.com

$$C_0 = I_1 + I_3$$
$$C_1 = I_2 + I_3$$

أي أن متغير الخرج C_0 يساوي 1 إذا كان طرف الدخل I_1 مساوياً 1 أو إذا كان طرف الدخل I_3 مساوياً 1. و متغير الخرج C_1 يساوي 1 إذا كان طرف الدخل I_2 مساوياً 1 أو إذا كان طرف الدخل I_3 مساوياً 1.

الدائرة المنطقية:



لاحظ أن الدائرة المنطقية للمشفر تتكون أساساً من مجموعة من بوابات OR بعدد أطراف الخرج.

لاحظ أنه يجب أن يكون لكل طرف من أطراف الدخل (Input Lines) في المشفر شفرة فريدة تميزه. فإذا كان عدد أطراف الخرج هو N فإن عدد الشفرات المتاحة 2^N، و بالتالي فإن عدد أطراف الدخل يجب أن يكون أقل من أو مساوياً 2^N.

تدريب تقويم ذاتي:

قمنا بكتابة التعبيرات المنطقية المختصرة لدائرة المشفر من نوع 4 إلى 2 الموضحة أعلاه مباشرة من جدول الصواب المختصر، و ذلك باستخدام أسلوب غير تقليدي. المطلوب الآن إتباع الأسلوب التقليدي للوصول إلى نفس التعبيرات المختصرة، و ذلك كالتالي:

- إنشاء جدول الصواب الكامل.
- كتابة التعبيرات المنطقية في صورة: - مجموع الحدود الصغرى (Sum of minterms).
  - مضروب الحدود الكبرى (Product of Maxterms).
- تبسيط التعبيرات المنطقية في كلا الصورتين باستخدام مخططات كارنو.

PDF created with pdfFactory Pro trial version www.pdffactory.com

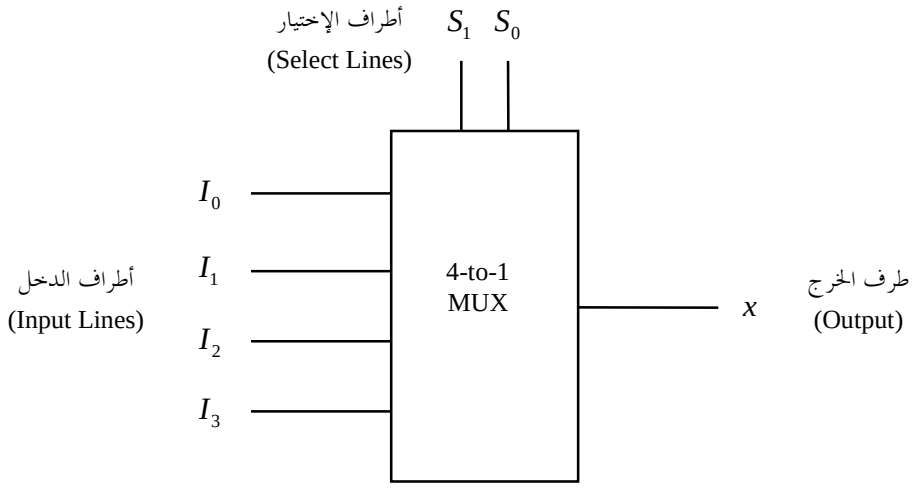
تدريب 12:

وضح المخطط المنطقي و جدول الصواب، ثم اكتب التعبيرات المنطقية و ارسم الدائرة المنطقية لمشفر من نوع 8 إلى 3 (8-to-3 Encoder).

6- الدامج (Multiplexer)

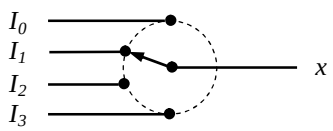
الدامج عبارة عن دائرة منطقية لها عدة أطراف دخل، و طرف خرج واحد. يتم توصيل واحد من أطراف الدخل مع طرف الخرج، و يتم اختيار طرف الدخل الذي يتم توصيله بالخرج بواسطة أطراف الإختيار (Select Lines).

و في ما يلي المخطط المنطقي و جدول الصواب لدامج من نوع 4 إلى 1 (4-to-1 Multiplexer)



S_1	S_0	x
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

هذا و يمكن تشبيه عمل الدامج بعمل المفتاح الدائري (Rotary Switch) الموضح أدناه



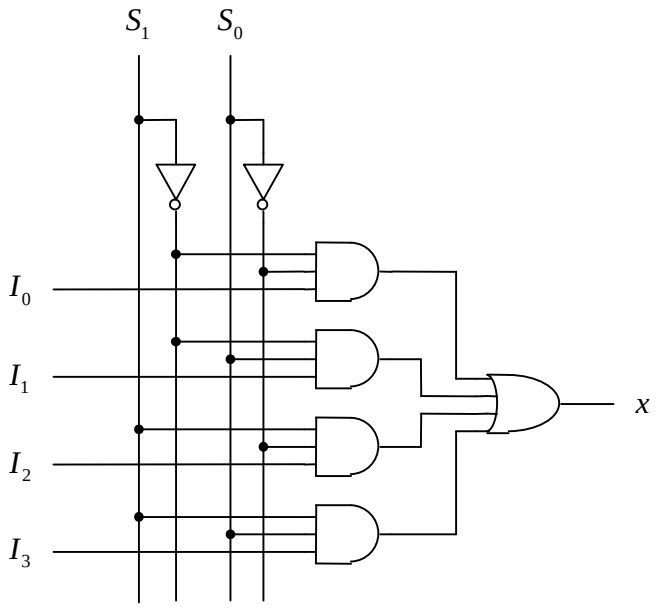
PDF created with pdfFactory Pro trial version www.pdffactory.com

التعبير المنطقي:

$$x = \overline{S_1}\,\overline{S_0} I_0 + \overline{S_1} S_0 I_1 + S_1 \overline{S_0} I_2 + S_1 S_0 I_3$$
$$x = m_0 I_0 + m_1 I_1 + m_2 I_2 + m_3 I_3$$

حيث m_0 ، m_1 ، m_2 ، m_3 هي الحدود الصغرى (minterms) لمتغيري الإختيار S_0 و S_1.

الدائرة المنطقية:



تدريب 13:

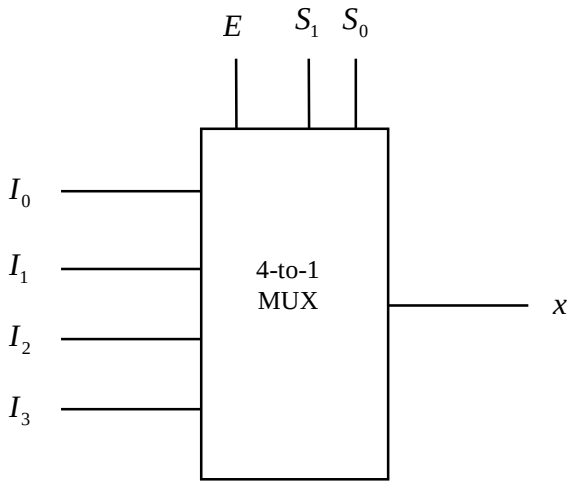
وضح المخطط المنطقي و جدول الصواب، ثم اكتب التعبير المنطقي و ارسم الدائرة المنطقية لدامج من نوع 8 إلى 1 (8-to-1 Multiplexer).

PDF created with pdfFactory Pro trial version www.pdffactory.com

خط السماح (Enable)

في بعض الأحيان قد يكون الدامج مزوداً بخط سماح (Enable). و وظيفة خط السماح، كما نعلم، هي إبطال عمل الدائرة أو السماح لها بأن تؤدي وظيفتها كالمعتاد.

و في ما يلي طريقة ظهور خط السماح في دامج من نوع 4 إلى 1



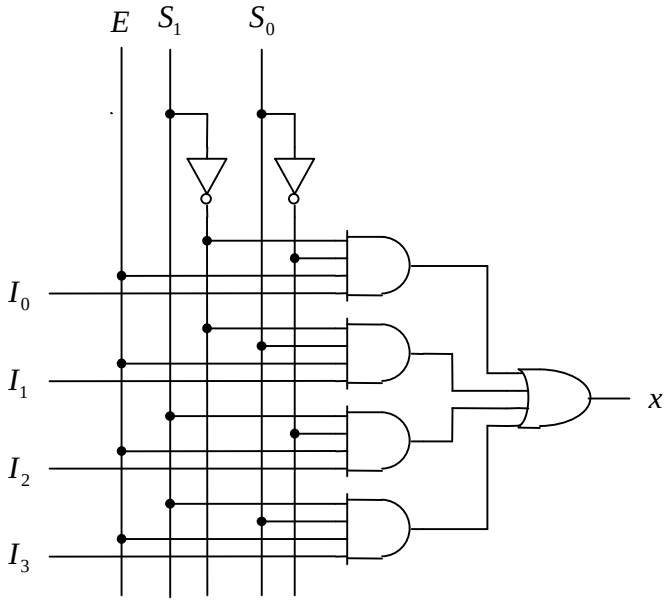
E	S_1	S_0	x
0	×	×	0
1	0	0	I_0
1	0	1	I_1
1	1	0	I_2
1	1	1	I_3

التعبير المنطقي:

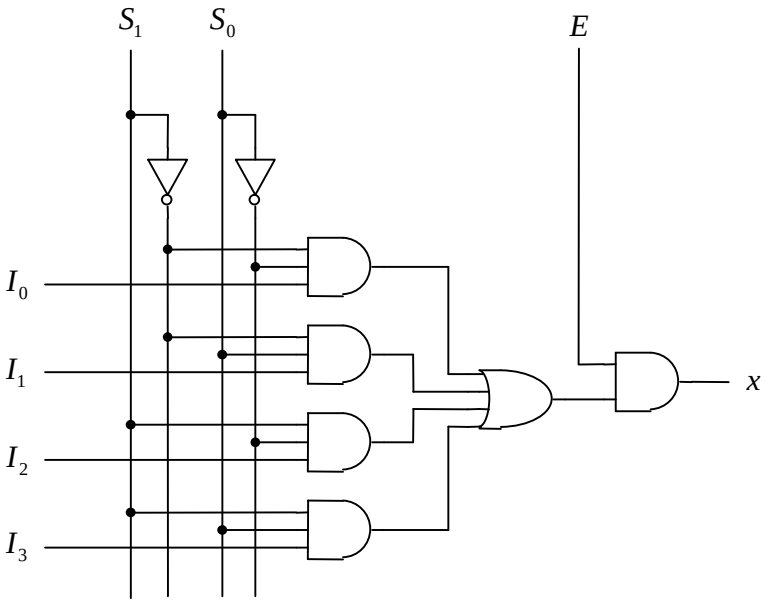
$$x = E\overline{S_1}\,\overline{S_0}I_0 + E\overline{S_1}S_0I_1 + ES_1\overline{S_0}I_2 + ES_1S_0I_3$$
$$x = E(\overline{S_1}\,\overline{S_0}I_0 + \overline{S_1}S_0I_1 + S_1\overline{S_0}I_2 + S_1S_0I_3)$$

PDF created with pdfFactory Pro trial version www.pdffactory.com

الدائرة المنطقية:



أو

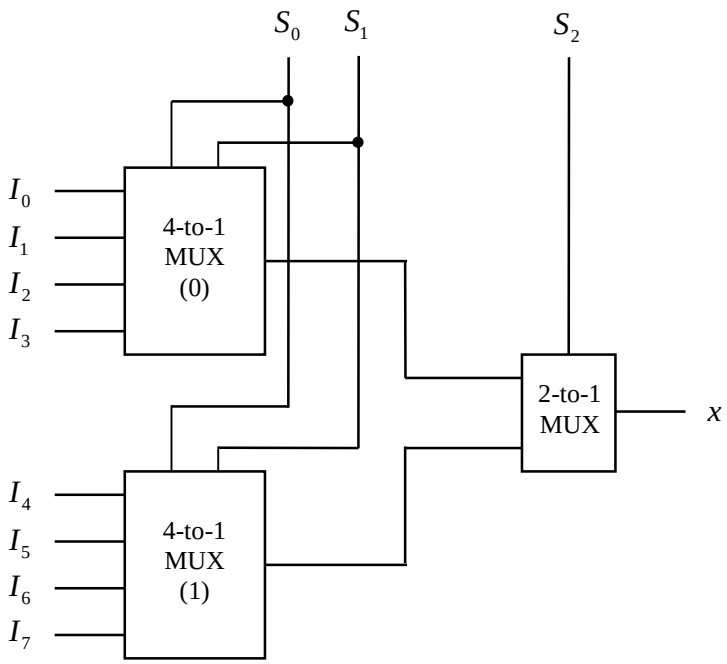


من الواضح أن الدائرة الثانية أفضل من الأولى لأنها أبسط.

PDF created with pdfFactory Pro trial version www.pdffactory.com

ربط الدوامج:

يمكن ربط عدد من وحدات الدامج الصغيرة لبناء وحدة دامج أكبر. مثلاً، يمكن ربط وحدتي دامج من نوع 4 إلى 1 (4-to-1 MUX's) لبناء دامج من نوع 8 إلى 1 (8-to-1 MUX)، كما هو موضح أدناه



من الشكل أعلاه يمكن ملاحظة أن خطوات الربط هي:

1. أطراف الدخل للوحدة الكبيرة موزعة بالتساوي ما بين الوحدات الصغيرة.
2. أطراف الإختيار الدنيا، و هي أطراف الإختيار التي تظهر في كل وحدة من الوحدات الصغيرة المطلوب ربطها، تكون مشتركة.
3. طرف الإختيار الأعلى يستخدم في اختيار الوحدة الصغيرة التي يتم تمرير خرجها إلى خرج الوحدة الكبيرة من بين الوحدات المربوطة مع بعضها البعض، و ذلك باستخدام دامج عدد أطراف الدخل له يساوي عدد الوحدات المربوطة.

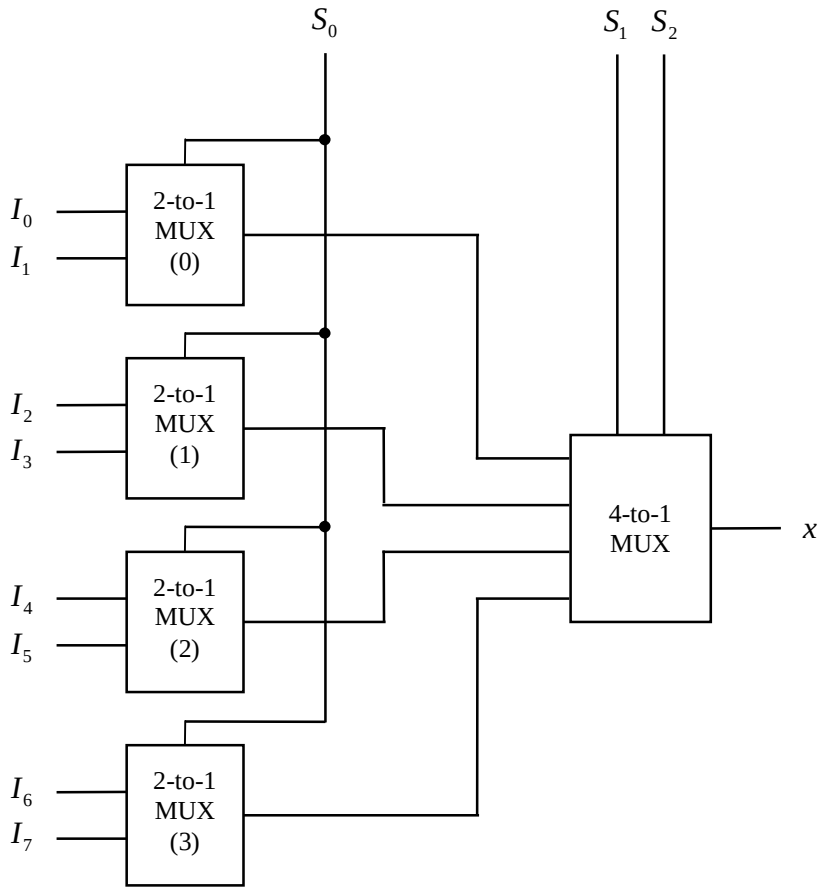
PDF created with pdfFactory Pro trial version www.pdffactory.com

مثال:

وضح طريقة ربط وحدات دامج من نوع 2 إلى 1 لبناء دامج من نوع 8 إلى 1.

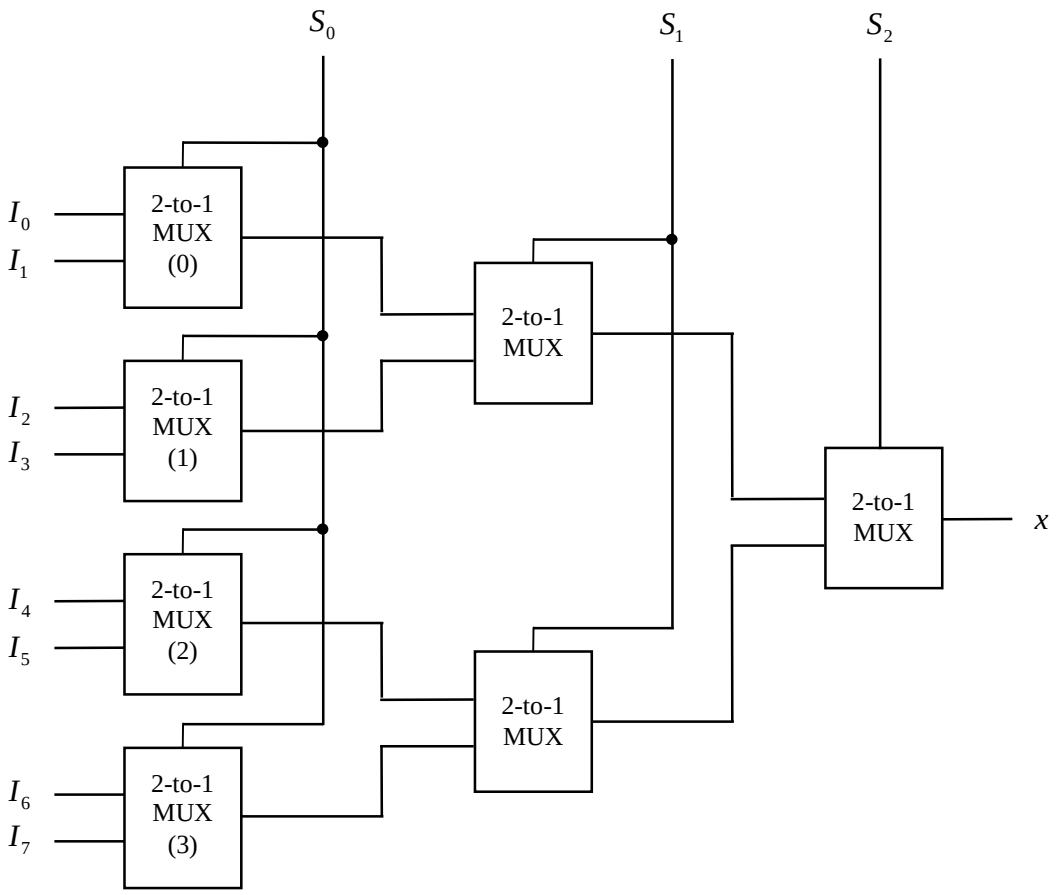
الحل:

بمقارنة عدد أطراف الدخل للوحدة الصغيرة المستخدمة في البناء و عدد أطراف الدخل للوحدة الكبيرة المطلوب بناؤها يتضح لنا أن عدد الوحدات الصغيرة التي نحتاج إليها هو 4. أما الربط فيتم كالتالي



في المثال أعلاه إذا كان مطلوباً بناء الدامج من نوع 8 إلى 1 باستخدام وحدات دامج من نوع 2 إلى 1 فقط فما علينا إلا أن نقوم ببناء الدامج من نوع 4 إلى 1 المستخدم في الربط باستخدام وحدات دامج من نوع 2 إلى 1، كما هو موضح أدناه

PDF created with pdfFactory Pro trial version www.pdffactory.com



تدريب 14:

وضح طريقة بناء دامج من نوع 16 إلى 1 (16-to-1 Multiplexer) و ذلك باستخدام وحدات دامج من نوع:

(أ) 4 إلى 1 (4-to-1 Multiplexers).

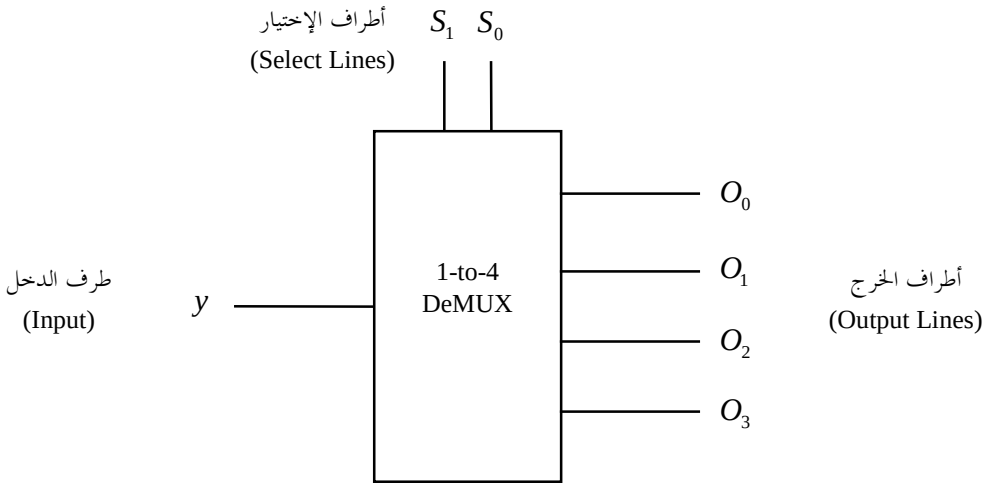
(ب) 2 إلى 1 (2-to-1 Multiplexers).

PDF created with pdfFactory Pro trial version www.pdffactory.com

7- المفرق (Demultiplexer)

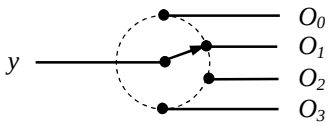
واضح من التسمية أن المفرق (Demultiplexer) يؤدي عكس الوظيفة التي يؤديها الدامج (Multiplexer)، فالمفرق عبارة عن دائرة منطقية لها عدة أطراف خرج، و طرف دخل واحد. يتم توصيل طرف الدخل مع أحد أطراف الخرج، و يتم اختيار طرف الخرج الذي يتم توصيله بالدخل بواسطة أطراف الإختيار (Select Lines).

و في ما يلي المخطط المنطقي و جدول الصواب لمفرق من نوع 1 إلى 4 (1-to-4 Demultiplexer)



S_1	S_0	O_3	O_2	O_1	O_0
0	0	0	0	0	y
0	1	0	0	y	0
1	0	0	y	0	0
1	1	y	0	0	0

هذا و يمكن تشبيه عمل المفرق بعمل المفتاح الدائري (Rotary Switch) الموضح أدناه



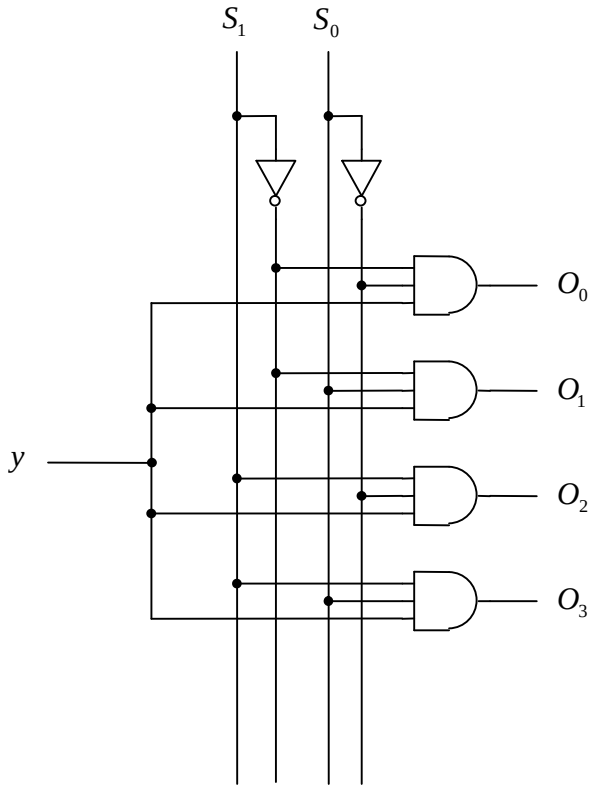
PDF created with pdfFactory Pro trial version www.pdffactory.com

التعبيرات المنطقية:

$$O_0 = \overline{S_1}\,\overline{S_0}y = m_0 y$$
$$O_1 = \overline{S_1}S_0 y = m_1 y$$
$$O_2 = S_1\overline{S_0}y = m_2 y$$
$$O_3 = S_1 S_0 y = m_3 y$$

حيث m_0 ، m_1 ، m_2 ، m_3 هي الحدود الصغرى (minterms) لمتغيري الإختيار S_0 و S_1.

الدائرة المنطقية:



لاحظ أن هذه الدائرة المنطقية تتطابق تماماً مع الدائرة المنطقية لفاك شفرة من نوع 2 إلى 4 مزود بخط سماح، مما يعني أن المفرق من نوع 1 إلى 4 يمكن استخدامه كفاك شفرة من نوع 2 إلى 4 مزود بخط سماح، و ذلك باستبدال طرفي الإختيار S_0 و S_1 بطرفي العنوان A_0 و A_1، و استبدال طرف الدخل y بخط السماح E.

PDF created with pdfFactory Pro trial version www.pdffactory.com

تدريب 15:

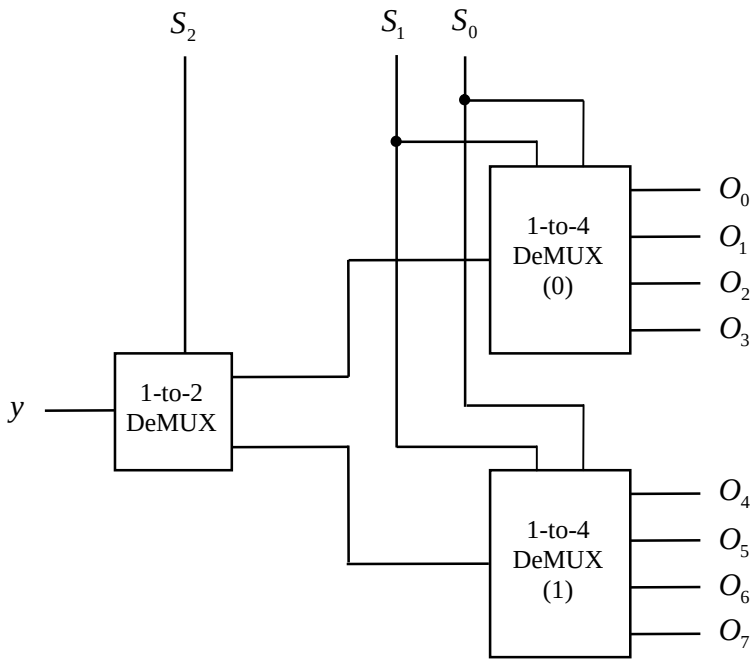
وضح المخطط المنطقي و جدول الصواب، ثم اكتب التعبيرات المنطقية و ارسم الدائرة المنطقية لمفرق من نوع:

(أ) 1 إلى 2 (1-to-2 Demultiplexer).

(ب) 1 إلى 8 (1-to-8 Demultiplexer).

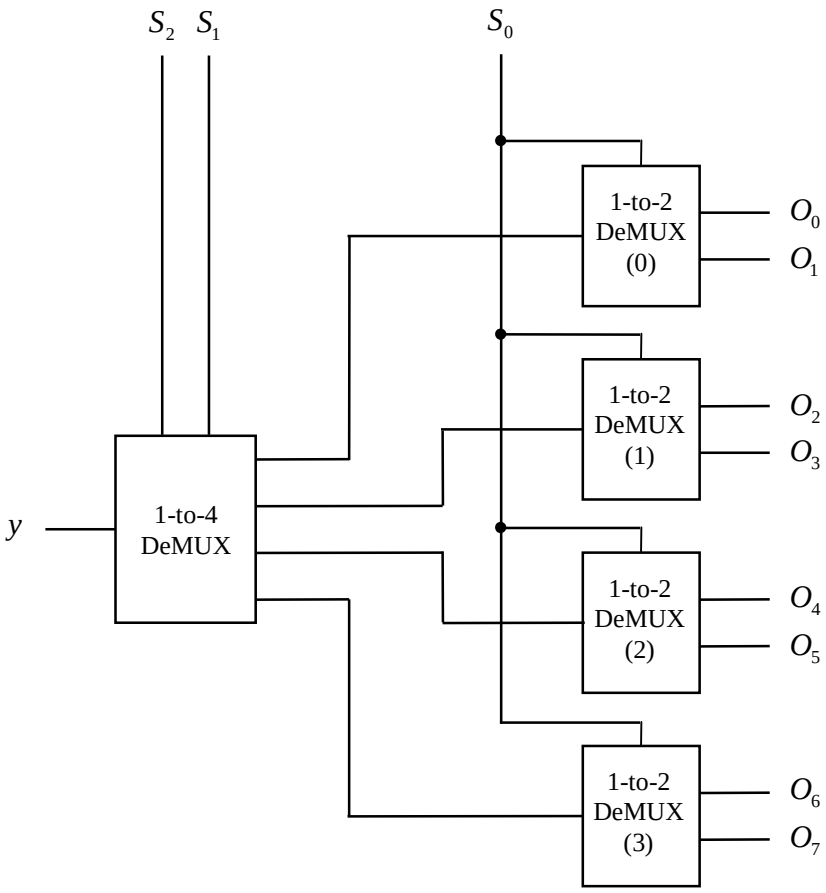
ربط المفرقات

يتم ربط المفرقات بنفس الأسلوب الذي أستخدم في ربط الدوامج. على سبيل المثال يمكن ربط وحدتي مفرق من نوع 1 إلى 4 لبناء مفرق من نوع 1 إلى 8 كما هو موضح بالشكل التالي



PDF created with pdfFactory Pro trial version www.pdffactory.com

كما يمكن ربط 4 وحدات مفرق من نوع 1 إلى 2 لبناء مفرق من نوع 1إلى 8 كما هو موضح أدناه



تدريب 16:

وضح طريقة بناء مفرق من نوع 1 إلى 8 باستخدام وحدات مفرق من نوع 1 إلى 2.

PDF created with pdfFactory Pro trial version www.pdffactory.com

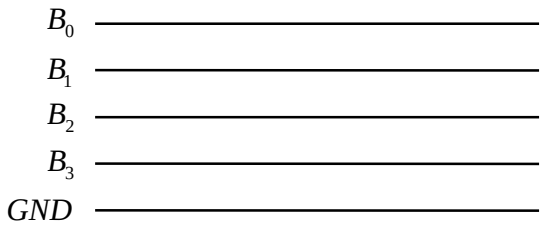
استخدام عملي للدامج و المفرق

يوجد أسلوبان لنقل البيانات (Data Transmission) في الأنظمة الرقمية:

1. النقل على التوازي (Parallel Transmission).
2. النقل على التوالي (Serial Transmission).

النقل على التوازي (Parallel Transmission):

في هذا الأسلوب يتم نقل مجموعة من الـــ bits دفعة واحدة على التوازي. و يتطلب هذا وجود موصل (سلك) لكل bit من الـــ bits المنقولة، إضافة إلى موصل أرضي (Ground) يستخدم كمرجع لقياس الجهد الكهربائي على بقية الموصلات، كما هو موضح أدناه

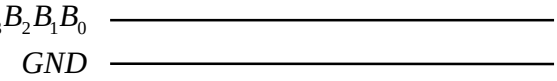


يمتاز هذا الأسلوب في نقل البيانات بالسرعة، و لكن يعيبه إرتفاع تكلفة الكيبل المستخدم و عدم إمكانية نقل البيانات لمسافات طويلة.

يستخدم هذا الأسلوب في نقل البيانات داخل جهاز الحاسوب بين أجزائه المختلفة مثل المعالج (Processor) و الذاكرة (Memory) عبر الناقل (Bus)، كما يستخدم في نقل البيانات من جهاز الحاسوب إلى الطابعة (Printer).

النقل على التوالي (Serial Transmission):

في هذا الأسلوب يتم استخدام موصلين فقط لنقل مجموعة من الـــ bits واحداً تلو الآخر، كما هو موضح أدناه

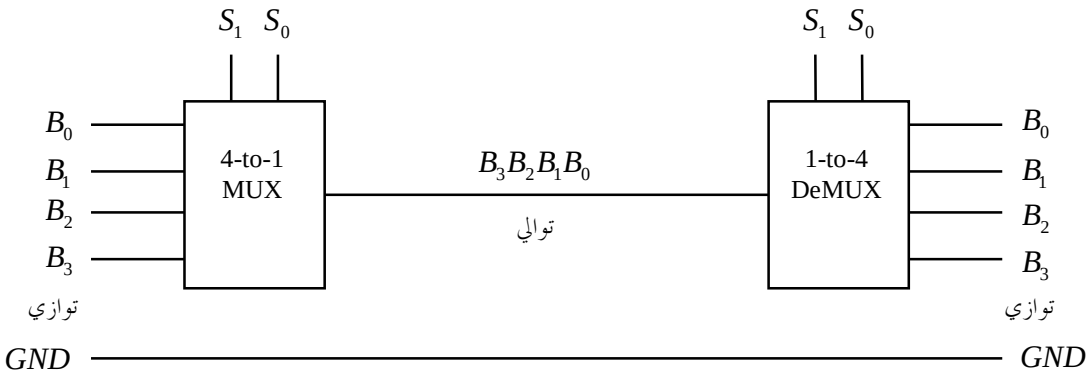


هذا الأسلوب في نقل البانات يعيبه البطء، إلا أنه يمتاز بانخفاض تكلفة الكيبل المستخدم و إمكانية نقل البيانات لمسافات طويلة.

يستخدم هذا الأسلوب في نقل البيانات في شبكات الحاسوب (Networks)، كما يستخدم عندما تكون السرعة في نقل البيانات غير مطلوبة مثل نقل البيانات من لوحة المفاتيح (Keyboard) و الفأرة (Mouse) إلى جهاز الحاسوب.

PDF created with pdfFactory Pro trial version www.pdffactory.com

في بعض الأحيان قد يكون مطلوباً تحويل البيانات المنقولة داخل النظام الرقمي من توازي إلى توالي أو العكس، كما يحدث داخل كرت الشبكة (Network Interface Card)، حيث يستقبل الكرت البيانات من جهاز الحاسوب على التوازي و يرسلها عبر كيبل الشبكة على التوالي. فلابد هنا من إجراء عملية تحويل للبيانات المنقولة من توازي إلى توالي. كما يجب إجراء العملية العكسية، أي التحويل من توالي إلى توازي، عند استقبال البيانات من كيبل الشبكة ونقلها إلى جهاز لحاسوب. هنا يأتي دور كل من الدامج و المفرق، حيث يمكن استخدام الدامج في التحويل من توازي إلى توالي، و استخدام المفرق في التحويل من توالي إلى توازي، كما هو موضح أدناه



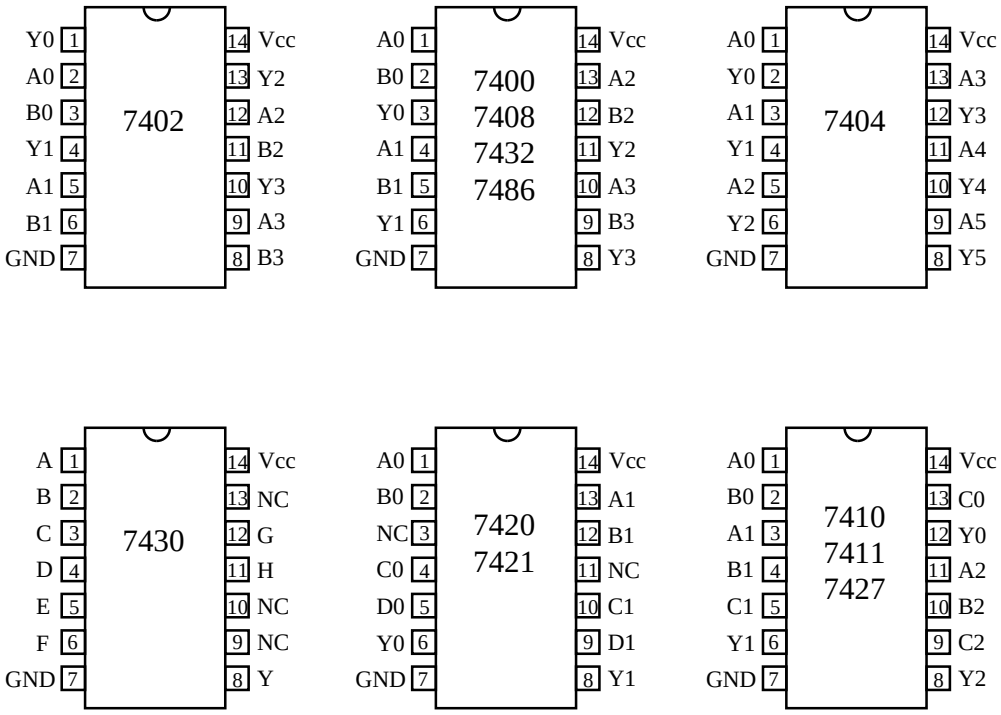
حيث يقوم كل من المرسل والمستقبل بوضع القيمة 00 في طرفي الإختيار S_0 و S_1 مما يعني إرسال القيمة الأولى B_0 ، بعدها يتم وضع القيمة 01 في طرفي الإختيار ممايعني إرسال القيمة B_1 ..... وهكذا

8- طرق بديلة لتصميم الدوائر المنطقية

السؤال الذي قد يتبادر إلى الذهن هنا هو: لماذا نحتاج إلى طرق تصميم بديلة؟

الطريقة التي درسناها لتصميم الدوائر المنطقية تهدف بالأساس إلى تقليل عدد البوابات المنطقية المستخدمة في بناء الدائرة لتقليل تكلفة الدائرة. و يصلح هذا الأسلوب إذا كان بناء الدائرة سيتم في مصنع متخصص في تصنيع الدوائر المنطقية، حيث يتم في هذه الحالة تصنيع البوابات اللازمة لبناء الدائرة من مكوناتها الأساسية (بلورات شبه موصلة)، و ذلك على سطح شريحة صغيرة من السيليكون، ثم يتم ربط تلك البوابات مع بعضها البعض لبناء الدائرة، و أخيراً يتم تغليف شريحة السيليكون في شكل دائرة متكاملة (Integrated Circuit). و يتم بناء الدائرة المنطقية بهذه الطريقة عندما يكون مطلوباً تصنيع عدد كبير من الوحدات من تلك الدائرة، أما إذا كان المطلوب هو عدد محدود من الوحدات من الدائرة فإن تصنيع تلك الدائرة في مصنع متخصص يكون غير عملي، حيث أن تكلفة إنتاج الوحدة الواحدة هنا ستكون مرتفعة للغاية. لذلك يتم بناء الدائرة المنطقية في مثل هذه الحالات باستخدام البوابات المنطقية الجاهزة المتوفرة تجارياً في شكل دوائر متكاملة (IC's). و الشكل التالي يوضح بعضاً من تلك الدوائر المتكاملة المتوفرة تجارياً

PDF created with pdfFactory Pro trial version www.pdffactory.com



الدائرة المتكاملة	محتوياتها
7404	Hex (6) Inverter
7400	Quad (4) 2-Input NAND Gate
7408	Quad (4) 2-Input AND Gate
7432	Quad (4) 2-Input OR Gate
7486	Quad (4) 2-Input XOR Gate
7402	Quad (4) 2-Input NOR Gate
7410	Triple (3) 3-Input NAND Gate
7411	Triple (3) 3-Input AND Gate
7427	Triple (3) 3-Input NOR Gate
7420	Dual (2) 4-Input NAND Gate
7421	Dual (2) 4-Input AND Gate
7430	8-Input NAND Gate

Vcc و GND هما طرفا تزويد الدائرة المتكاملة بالقدرة الكهربائية (Power)

NC تعني (No Connection) أي أن الطرف غير مستخدم

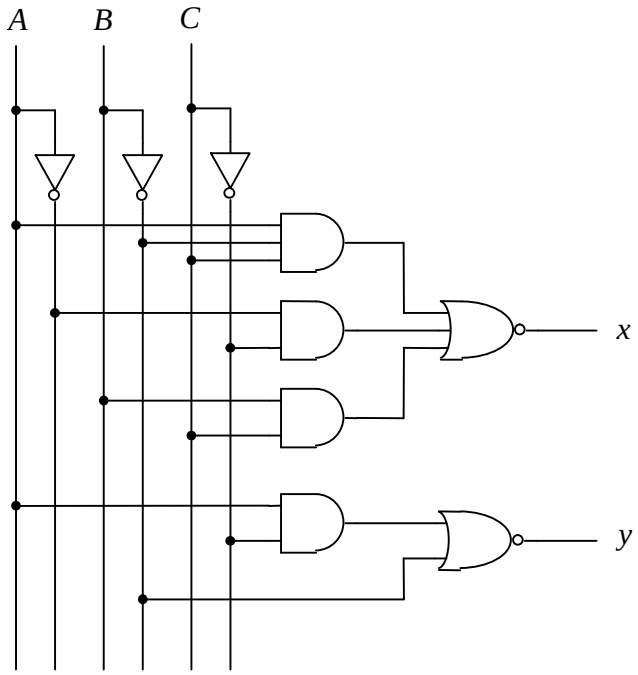
A0، B0، C0، .... هي أطراف الدخل للبوابة رقم 0

Y0 هو طرف الخرج للبوابة رقم 0

PDF created with pdfFactory Pro trial version www.pdffactory.com

من المذكور أعلاه نلاحظ أن البوابات المنطقية الجاهزة لا تُباع منفردة و إنما تأتي مجموعة منها من نوع واحد في شكل دائرة متكاملة (I.C.). فمثلاً العواكس المنطقية تأتي ستة منها في الدائرة المتكاملة 7404، و بوابات AND بمدخلين تأتي أربعة منها في الدائرة المتكاملة 7408، و بوابات NOR بثلاثة مداخل تأتي ثلاثة منها في الدائرة المتكاملة 7427، و بوابات NAND بأربعة مداخل تأتي إثنتان منها في الدائرة المتكاملة 7420، ... و هكذا.

فإذا أردنا بناء الدائرة المنطقية التالية، مثلاً، باستخدام البوابات الجاهزة



فإننا نحتاج إلى:

- 3 عواكس منطقية $\left(\frac{3}{6}\times 7404\right)$
- 3 بوابات AND بمدخلين $\left(\frac{3}{4}\times 7408\right)$
- بوابة AND بثلاثة مداخل $\left(\frac{1}{3}\times 7411\right)$
- بوابة NOR بمدخلين $\left(\frac{1}{4}\times 7402\right)$
- بوابة NOR بثلاثة مداخل $\left(\frac{1}{3}\times 7427\right)$

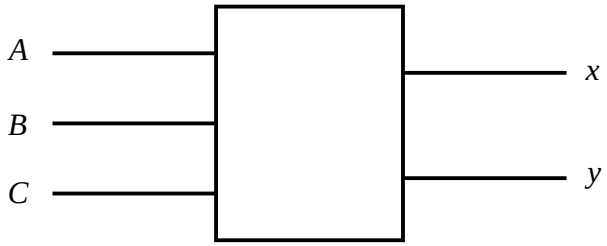
PDF created with pdfFactory Pro trial version www.pdffactory.com

أي أننا نحتاج إلى 5 دوائر متكاملة، وهذا العدد من الدوائر المتكاملة يشغل مساحة كبيرة من سطح لوحة التوصيل. كما أن عدداً مقدراً من البوابات بتلك الدوائر المتكاملة غير مستخدم، مع العلم بأن تلك البوابات غير المستخدمة تستهلك القدرة الكهربائية، مما يعني هدراً للأموال و للطاقة الكهربائية و إنبعاث كمية من الحرارة قد لا يكون من السهل التخلص منها.

إذن المطلوب الآن هو طرق تصميم بديلة للدوائر المنطقية تهدف بالأساس إلى تقليل عدد الدوائر المتكاملة (IC's) المستخدمة في بناء الدائرة المنطقية، بدلاً عن تقليل عدد البوابات المنطقية المستخدمة في بناء الدائرة المنطقية. في طرق التصميم البديلة هذه نستعين بالدوائر المنطقية الترابطية الجاهزة مثل فاك الشفرة (Decoder) و الدامج (Multiplexer).

التصميم باستخدام فاك شفرة و مشفر (Decoder & Encoder)

المطلوب تصميم الدائرة المنطقية الموضح المخطط المنطقي و جدول الصواب لها أدناه باستخدام فاك شفرة و مشفر



#	*A*	*B*	*C*	*x*	*y*
0	0	0	0	0	1
1	0	0	1	1	0
2	0	1	0	0	1
3	0	1	1	1	0
4	1	0	0	0	1
5	1	0	1	0	1
6	1	1	0	0	0
7	1	1	1	1	0

PDF created with pdfFactory Pro trial version www.pdffactory.com

خطوات التصميم:

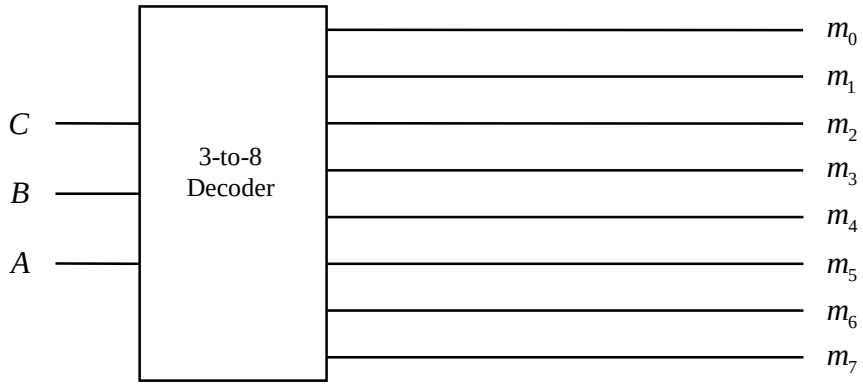
1. نقوم بكتابة التعبيرات المنطقية في صورة مجموع الحدود الصغرى (Sum of minterms)

$$x = f(A,B,C) = \sum m\,(1,3,7)$$

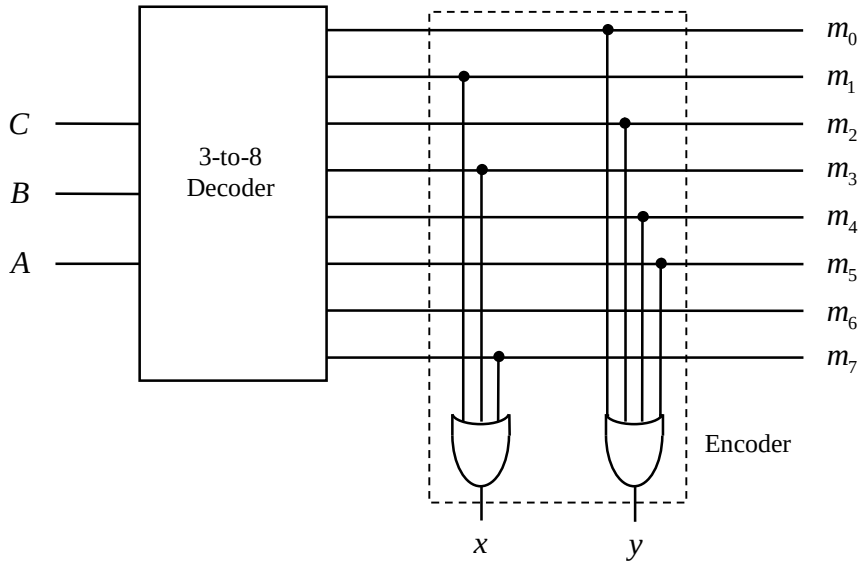
$$y = f(A,B,C) = \sum m\,(0,2,4,5)$$

2. نحتاج إلى فاك شفرة عدد أطراف العنوان (Address Lines) له يساوي عدد متغيرات الدخل للدائرة المنطقية المطلوب تصميمها، أي فاك شفرة من نوع 3 إلى 8.

3. نقوم بإدخال متغيرات الدخل للدائرة المنطقية المطلوب تصميمها إلى أطراف العنوان لفاك الشفرة، مع مراعاة الترتيب (أي أن الـــ MSB في متغيرات الدخل يجب إدخاله إلى الـــ MSB في أطراف العنوان)، فيقوم فاك الشفرة بتوليد الحدود الصغرى (minterms) لمتغيرات الدخل في أطراف الخرج له، كما هو موضح أدناه



4. نستخدم بوابات OR في جمع الحدود الصغرى المناسبة لتوليد متغيرات الخرج



لاحظ أن بوابات OR هنا تمثل مشفراً من نوع 8 إلى 2.

PDF created with pdfFactory Pro trial version www.pdffactory.com

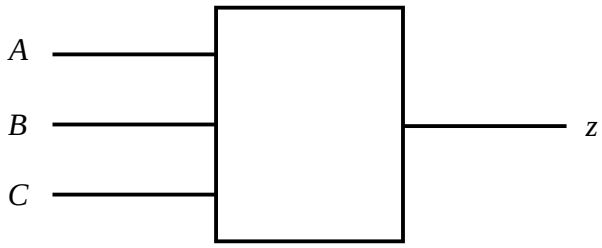
احتجنا لبناء الدائرة المنطقية هنا إلى ثلاثة دوائر متكاملة:

- مشفر من نوع 3 إلى 8
- بوابات OR بثلاثة مداخل
- بوابات OR بأربعة مداخل

و في واقع الأمر فإنه توجد دائرة متكاملة (I.C.) تحتوي على فاك الشفرة و المشفر معاً و هي ذاكرة ROM كما سنرى فيما بعد.

التصميم باستخدام الدامج (Multiplexer)

المطلوب تصميم الدائرة المنطقية الموضح المخطط المنطقي و جدول الصواب لها أدناه باستخدام دامج



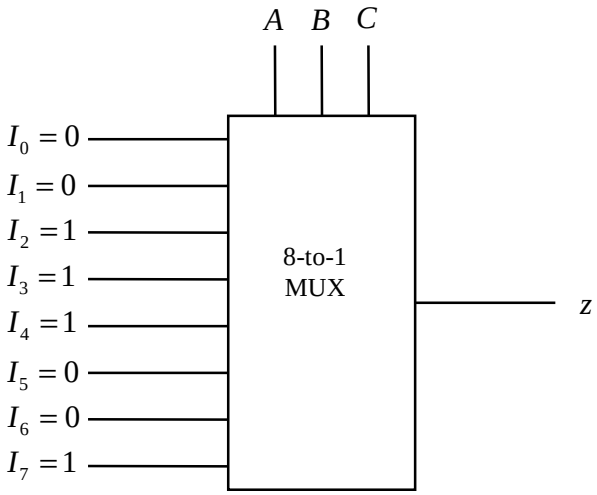
A	*B*	*C*	*z*
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

خطوات التصميم:

1. نحتاج إلى دامج عدد أطراف الإختيار (Select Lines) له يساوي عدد متغيرات الدخل للدائرة المنطقية المطلوب تصميمها، أي دامج من نوع 8 إلى 1 (لاحظ أن عدد أطراف الدخل للدامج هنا يكون مساوياً لعدد أسطر جدول الصواب).
2. نقوم بإدخال متغيرات الدخل للدائرة المنطقية المطلوب تصميمها إلى أطراف الإختيار للدامج، مع مراعاة الترتيب.
3. نأخذ قيم الخرج من جدول الصواب و نضعها بالترتيب على أطراف الدخل للدامج.
4. نأخذ خرج الدائرة المنطقية من طرف الخرج للدامج.

PDF created with pdfFactory Pro trial version www.pdffactory.com

كما هو موضح بالشكل التالي



لاحظ أننا قد احتجنا هنا لبناء الدائرة المنطقية إلى دائرة متكاملة واحدة فقط.

لاحظ أيضاً أن هذا الأسلوب في تصميم الدوائر المنطقية يصلح للدوائر ذات طرف الخرج الواحد، فإذا كان للدائرة أكثر من طرف خرج واحد نحتاج لدامج لكل طرف من أطراف الخرج.

التصميم باستخدام دامج عدد أطراف الإختيار له أقل من عدد متغيرات الدخل بواحد

المطلوب الآن تصميم نفس الدائرة المنطقية أعلاه و لكن باستخدام دامج من نوع 4 إلى 1.

خطوات التصميم:

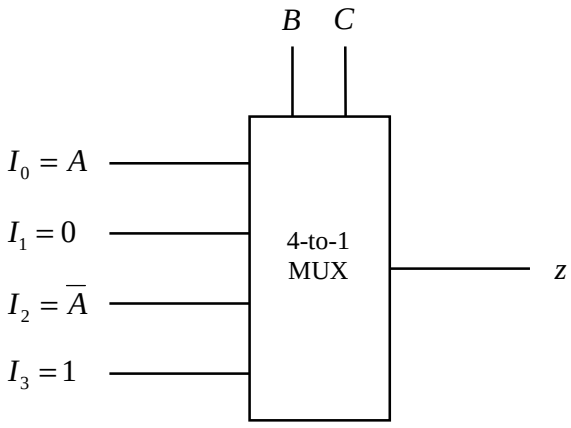
1. نحتاج إلى دامج عدد أطراف الإختيار (Select Lines) له أقل من عدد متغيرات الدخل للدائرة المنطقية المطلوب تصميمها بواحد، أي دامج من نوع 4 إلى 1 (لاحظ أن عدد أطراف الدخل للدامج هنا يكون مساوياً لنصف عدد أسطر جدول الصواب).
2. نقوم بفصل متغير الدخل الأعلى A عن بقية متغيرات الدخل في جدول الصواب، مما يؤدي إلى تقسيم جدول الصواب إلى نصفين: النصف الأعلى و فيه $A = 0$ ، و النصف الأسفل و فيه $A = 1$

PDF created with pdfFactory Pro trial version www.pdffactory.com

A	B	C	z
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

3. نقوم بمقارنة قيمة متغير الخرج z في كل سطر من أسطر النصف الأعلى من جدول الصواب مع السطر المقابل له من أسطر النصف الأسفل من جدول الصواب، و ذلك لإيجاد علاقة مابين متغير الخرج z و متغير الدخل الأعلى A. فبمقارنة السطر الأول من النصف الأعلى مع السطر الأول من النصف الأسفل نجد أن $z = A$، و بمقارنة السطر الثاني من النصف الأعلى مع السطر الثاني من النصف الأسفل نجد أن $z = 0$، و بمقارنة السطر الثالث من النصف الأعلى مع السطر الثالث من النصف الأسفل نجد أن $z = \overline{A}$، و بمقارنة السطر الرابع من النصف الأعلى مع السطر الرابع من النصف الأسفل نجد أن $z = 1$.
4. نقوم بإدخال متغيرات الدخل الدنيا للدائرة المنطقية المطلوب تصميمها إلى أطراف الإختيار للدامج، مع مراعاة الترتيب.
5. نقوم بإدخال العلاقات التي توصلنا إليها في الخطوة 3 بالترتيب إلى أطراف الدخل للدامج.
6. نأخذ خرج الدائرة المنطقية من طرف الخرج للدامج.

كما هو موضح بالشكل التالي



PDF created with pdfFactory Pro trial version www.pdffactory.com

تدريب 17:

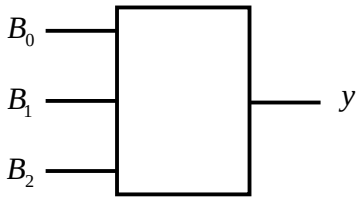
صمم الدائرة المنطقية الموضح المخطط المنطقي وجدول الصواب لها أدناه، و ذلك باستخدام:

(أ) فاك شفرة و مشفر (Decoder & Encoder).

(ب) دامج من نوع 8 إلى 1 (8-to-1 MUX).

(ج) دامج من نوع 4 إلى 1 (4-to-1 MUX).

B_2	B_1	B_0	y
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1



الخلاصة

تعرفنا في هذه الوحدة على بعض الدوائر المنطقية الترابطية التي تقوم بأداء وظائف مفيدة، و التي يشيع استخدامها في الأنظمة الرقمية، مثل الجامع بأنواعه و فاك الشفرة و المشفر و الدامج و المفرق. حيث تعرفنا على وظيفة كل دائرة من هذه الدوائر و تركيبها و استخداماتها. كما تعرفنا على طريقة ربط وحدات صغيرة من تلك الدوائر لبناء وحدات أكبر. أيضاً عرفنا أن هذه الدوائر متوفرة تجارياً في صورة دوائر متكاملة (IC's) بحيث يمكن شراؤها و استخدامها مباشرة في بناء الأنظمة الرقمية. و أخيراً تعلمنا طرق بديلة لتصميم الدوائر المنطقية تستخدم فيها تلك الدوائر المنطقية الترابطية الجاهزة.

لمحة مسبقة عن الوحدة التالية

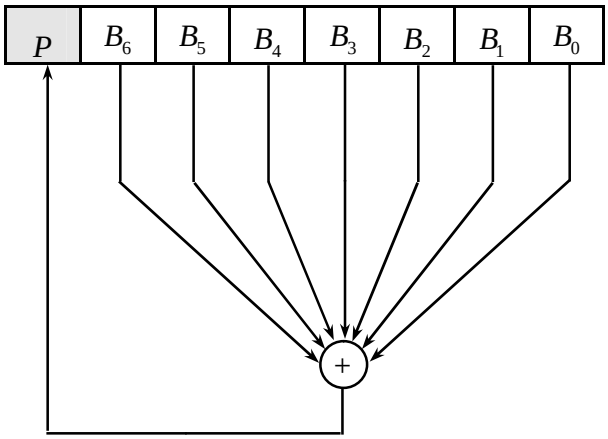
نعلم الآن أن جميع الدوائر المنطقية التي تعاملنا معها حتى الآن هي دوائر منطقية ترابطية (Combinational)، و أنها سميت بالترابطية لأن وظيفة الدائرة تقتصر على ربط متغيرات الدخل بعمليات منطقية لتوليد متغيرات الخرج. كما نعلم أن الخرج في الدوائر الترابطية يعتمد فقط على القيم الحالية للدخل، فمتى ما تغيرت قيم الدخل تغيرت معها قيم الخرج. في الوحدة التالية من المقرر سنتعرف على النوع الآخر من الدوائر المنطقية و هو الدوائر المنطقية التتابعية (Sequential)، مثل المراجيح (Flip Flops) و المسجلات (Registers) و العدّادات (Counters). حيث سنعرف أن الخرج في هذا النوع من الدوائر المنطقية لا يعتمد فقط على القيم الحالية للدخل، و إنما يعتمد أيضاً على القيم السابقة للخرج. أي أن هذا النوع من الدوائر له ذاكرة (Memory) تستطيع اختزان ماضي الدائرة بحيث يؤثر على خرجها الحالي.

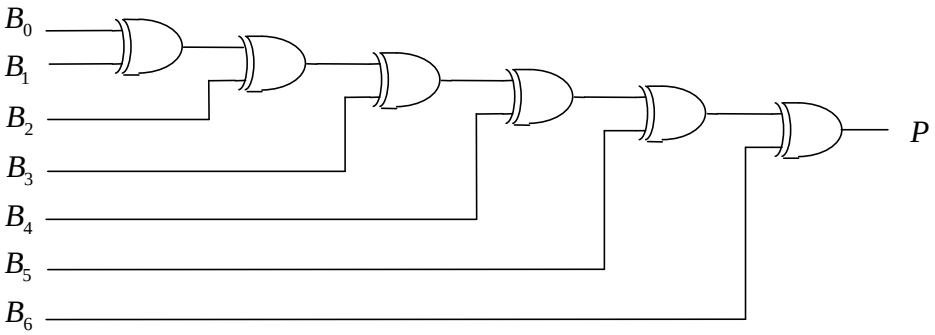
PDF created with pdfFactory Pro trial version www.pdffactory.com

إجابات التدريبات

تدريب 1:

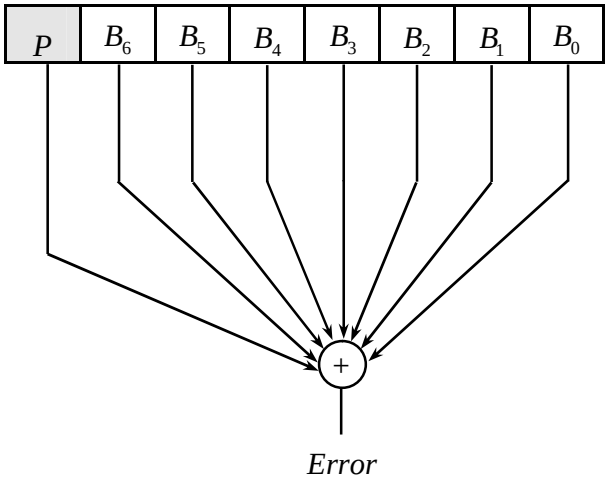
دائرة توليد خانة التحقق الزوجي (Even Parity bit Generator):

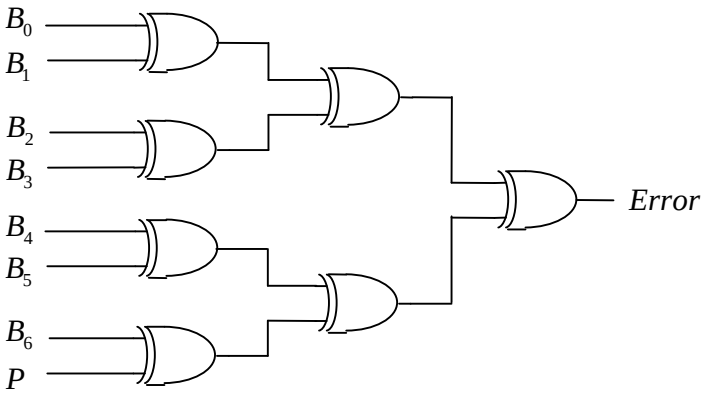




PDF created with pdfFactory Pro trial version www.pdffactory.com

دائرة التحقق الزوجي (Even Parity Checker):

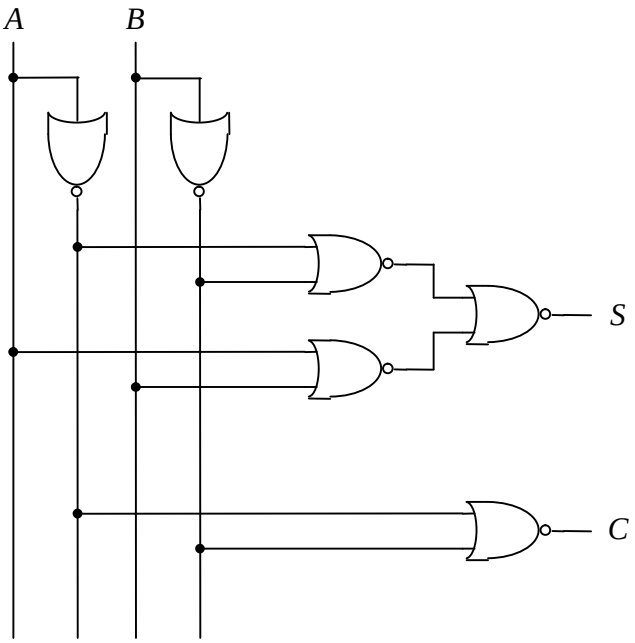




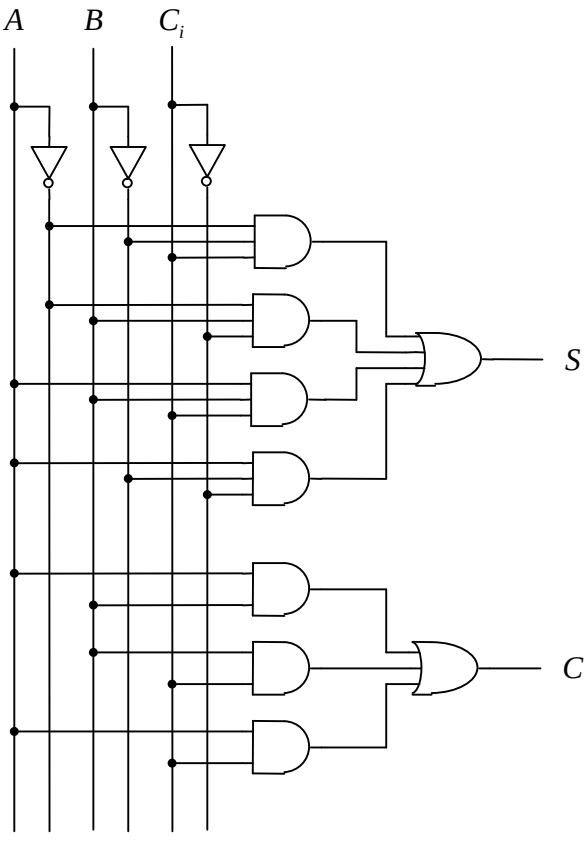
إذا كان فإن $Error = 1$ فإن هنالك خطأ في الرمز، أما إذا كان $Error = 0$ فلا يوجد خطأ.

PDF created with pdfFactory Pro trial version www.pdffactory.com

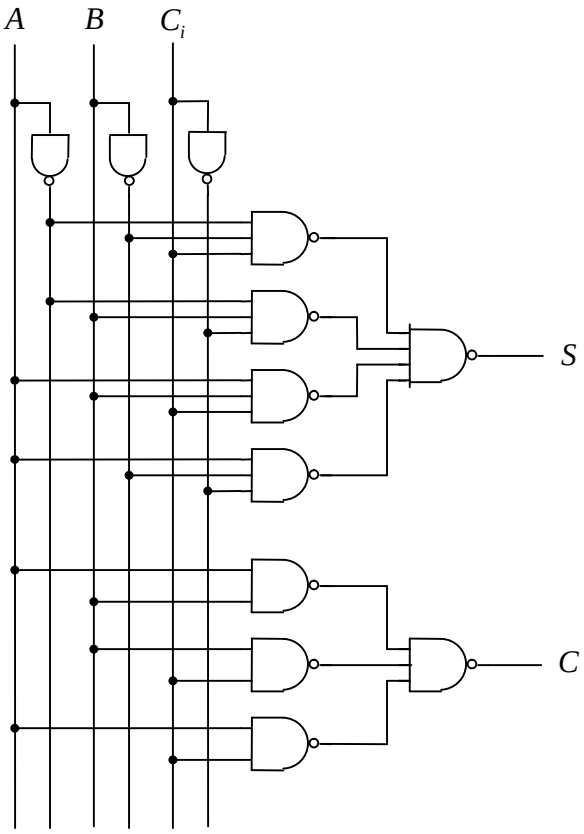
تدریب 2:



تدریب 3:

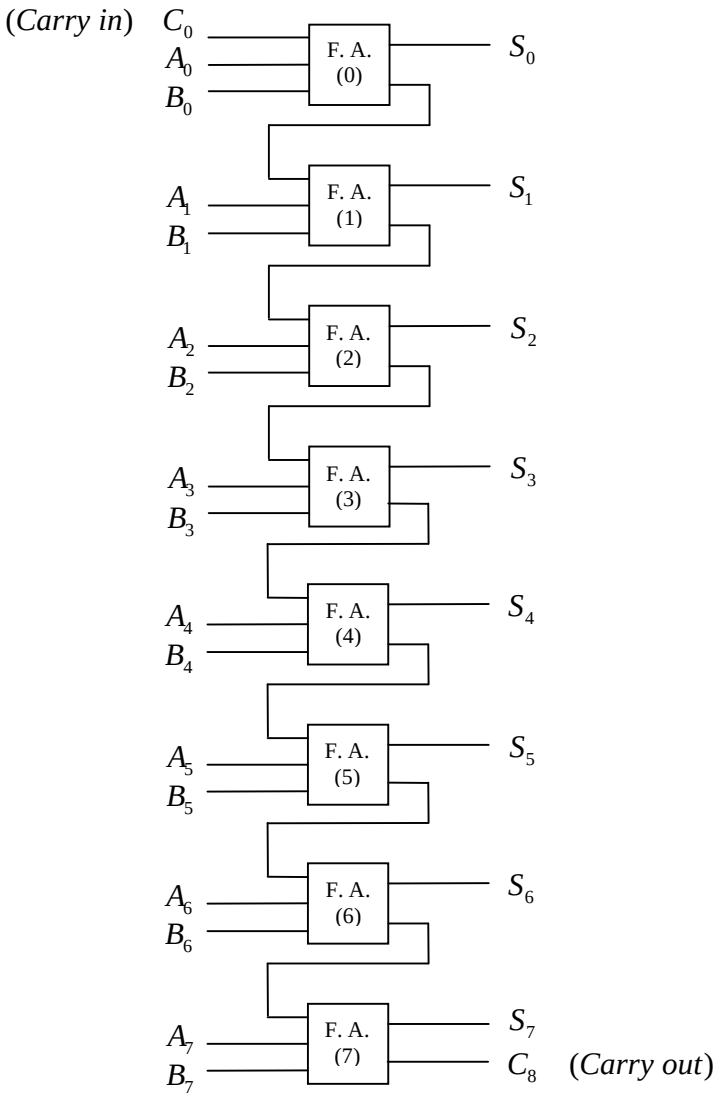


PDF created with pdfFactory Pro trial version www.pdffactory.com



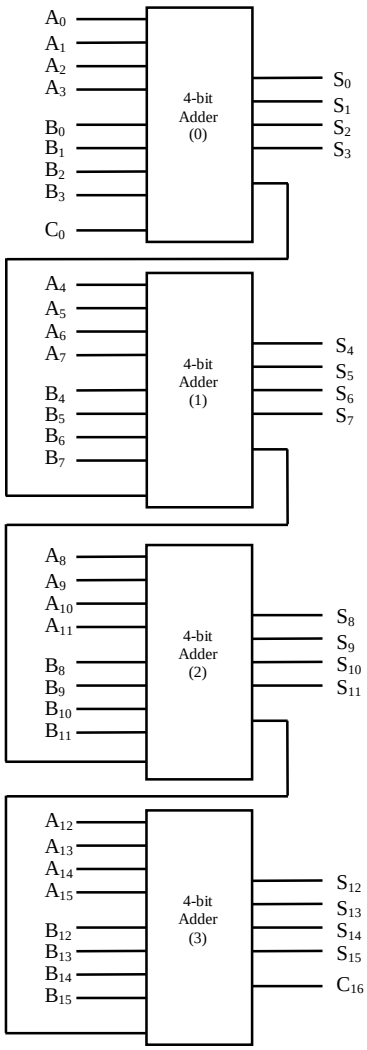
PDF created with pdfFactory Pro trial version www.pdffactory.com

تدریب 4:



PDF created with pdfFactory Pro trial version www.pdffactory.com

تدریب 5:



PDF created with pdfFactory Pro trial version www.pdffactory.com

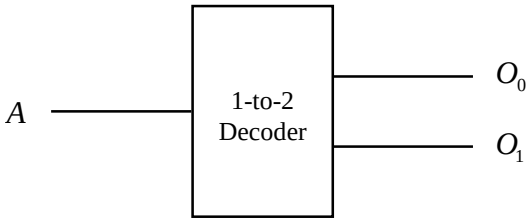
تدریب 6:

G_4	G_3	G_2	G_1	G_0	Operation
0	0	0	0	0	0
0	0	0	0	1	1
0	0	0	1	0	-1
0	0	0	1	1	0
0	0	1	0	0	-1
0	0	1	0	1	0
0	0	1	1	0	-2
0	0	1	1	1	-1
0	1	0	0	0	A
0	1	0	0	1	$A+1=A++$
0	1	0	1	0	$-A-1=\overline{A}$
0	1	0	1	1	$-A$
0	1	1	0	0	$A-1=A--$
0	1	1	0	1	A
0	1	1	1	0	$-A-2$
0	1	1	1	1	$-A-1=\overline{A}$
1	0	0	0	0	B
1	0	0	0	1	$B+1=B++$
1	0	0	1	0	$B-1=B--$
1	0	0	1	1	B
1	0	1	0	0	$-B-1=\overline{B}$
1	0	1	0	1	$-B$
1	0	1	1	0	$-B-2$
1	0	1	1	1	$-B-1=\overline{B}$
1	1	0	0	0	$A+B$
1	1	0	0	1	$A+B+1$
1	1	0	1	0	$B-A-1$
1	1	0	1	1	$B-A$
1	1	1	0	0	$A-B-1$
1	1	1	0	1	$A-B$
1	1	1	1	0	$-A-B-2$
1	1	1	1	1	$-A-B-1$

PDF created with pdfFactory Pro trial version www.pdffactory.com

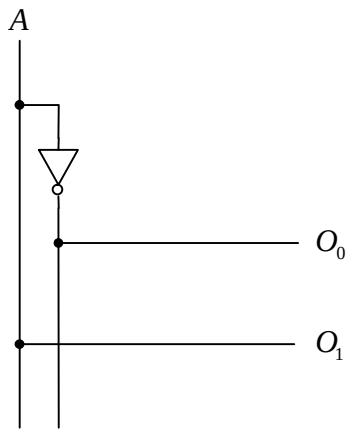
تدريب 7:

(أ) فاك شفرة من نوع 1 إلى 2 (1-to-2 Decoder)



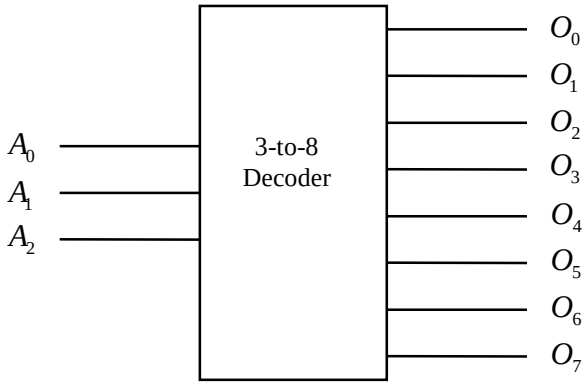
A	O_1	O_0
0	0	1
1	1	0

$$O_0 = \overline{A}$$
$$O_1 = A$$



PDF created with pdfFactory Pro trial version www.pdffactory.com

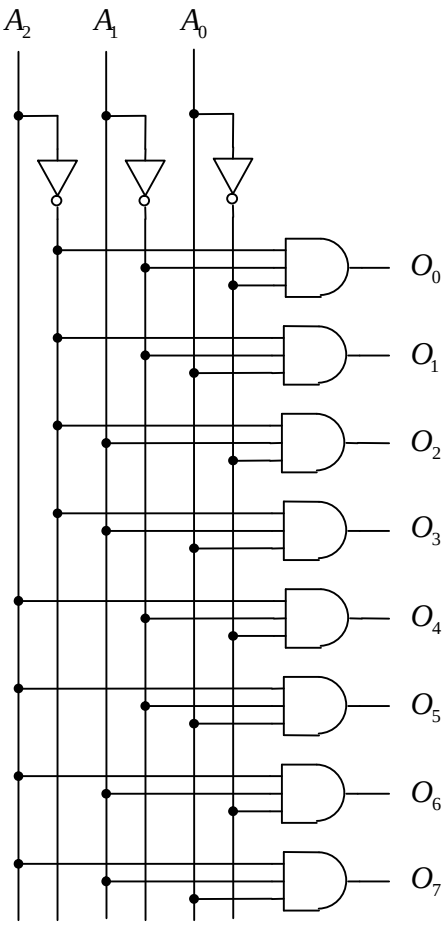
(ب) فاك شفرة من نوع 3 إلى 8 (3-to-8 decoder)



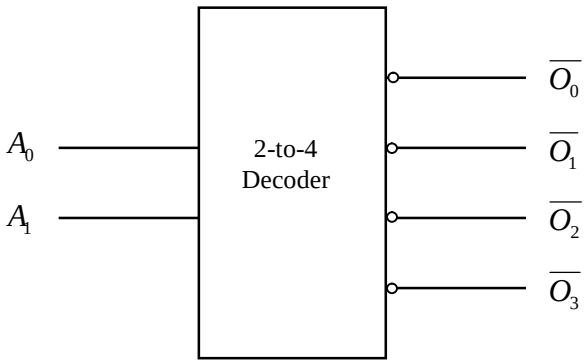
A_2	A_1	A_0	O_7	O_6	O_5	O_4	O_3	O_2	O_1	O_0
0	0	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	1	0	0	0	0	0	0	1	0	0
0	1	1	0	0	0	0	1	0	0	0
1	0	0	0	0	0	1	0	0	0	0
1	0	1	0	0	1	0	0	0	0	0
1	1	0	0	1	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0	0

$$O_0 = \overline{A_2}\,\overline{A_1}\,\overline{A_0}$$
$$O_1 = \overline{A_2}\,\overline{A_1}A_0$$
$$O_2 = \overline{A_2}A_1\overline{A_0}$$
$$O_3 = \overline{A_2}A_1A_0$$
$$O_4 = A_2\overline{A_1}\,\overline{A_0}$$
$$O_5 = A_2\overline{A_1}A_0$$
$$O_6 = A_2A_1\overline{A_0}$$
$$O_7 = A_2A_1A_0$$

PDF created with pdfFactory Pro trial version www.pdffactory.com



تدريب 8:



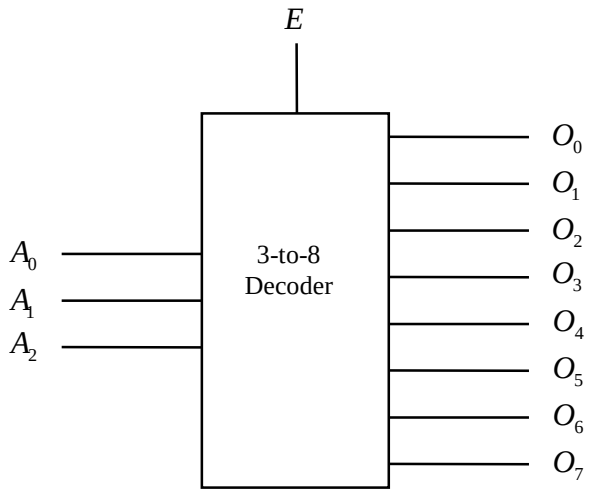
PDF created with pdfFactory Pro trial version www.pdffactory.com

A_1	A_0	$\overline{O_3}$	$\overline{O_2}$	$\overline{O_1}$	$\overline{O_0}$
0	0	1	1	1	0
0	1	1	1	0	1
1	0	1	0	1	1
1	1	0	1	1	1

$$\overline{O_0} = \overline{\overline{A_1}\,\overline{A_0}}$$
$$\overline{O_1} = \overline{\overline{A_1}A_0}$$
$$\overline{O_2} = \overline{A_1\overline{A_0}}$$
$$\overline{O_3} = \overline{A_1 A_0}$$

تدريب 9:

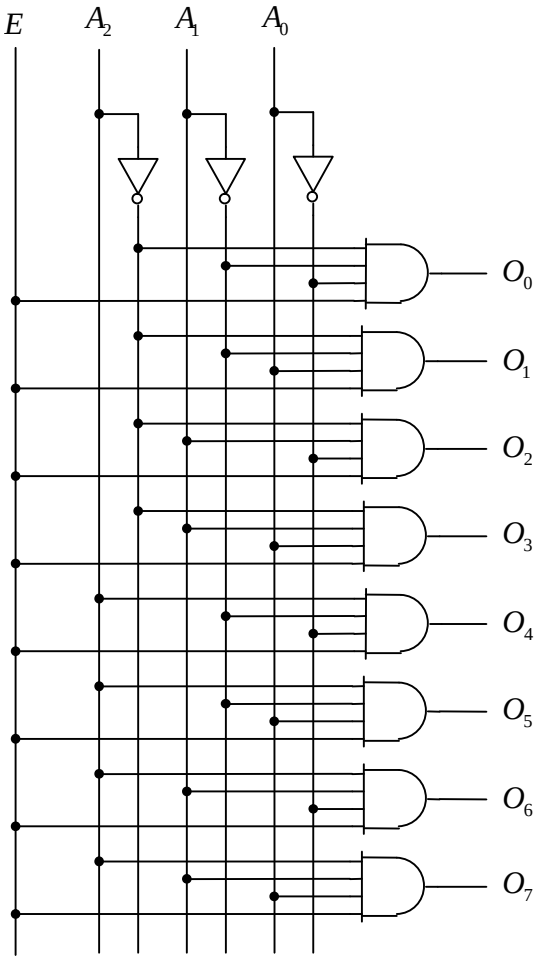
(أ) فاك شفرة من نوع 3 إلى 8 بخط سماح (3-to-8 Decoder with Enable)



E	A_2	A_1	A_0	O_7	O_6	O_5	O_4	O_3	O_2	O_1	O_0
0	×	×	×	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	1
1	0	0	1	0	0	0	0	0	0	1	0
1	0	1	0	0	0	0	0	0	1	0	0
1	0	1	1	0	0	0	0	1	0	0	0
1	1	0	0	0	0	0	1	0	0	0	0
1	1	0	1	0	0	1	0	0	0	0	0
1	1	1	0	0	1	0	0	0	0	0	0
1	1	1	1	1	0	0	0	0	0	0	0

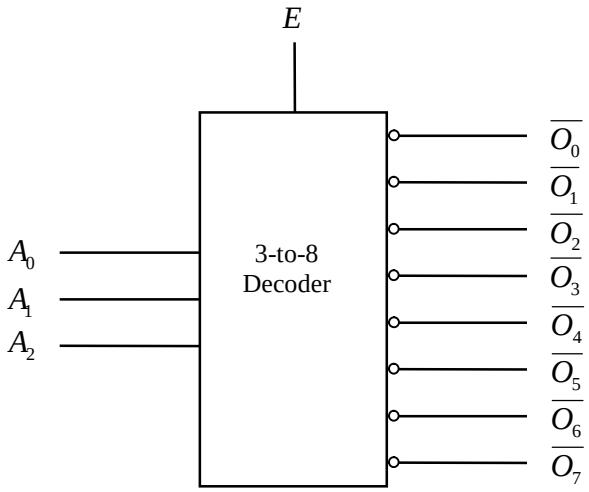
PDF created with pdfFactory Pro trial version www.pdffactory.com

$$O_0 = E\,\overline{A_2}\,\overline{A_1}\,\overline{A_0}$$
$$O_1 = E\,\overline{A_2}\,\overline{A_1}\,A_0$$
$$O_2 = E\,\overline{A_2}\,A_1\,\overline{A_0}$$
$$O_3 = E\,\overline{A_2}\,A_1\,A_0$$
$$O_4 = E\,A_2\,\overline{A_1}\,\overline{A_0}$$
$$O_5 = E\,A_2\,\overline{A_1}\,A_0$$
$$O_6 = E\,A_2\,A_1\,\overline{A_0}$$
$$O_7 = E\,A_2\,A_1\,A_0$$



PDF created with pdfFactory Pro trial version www.pdffactory.com

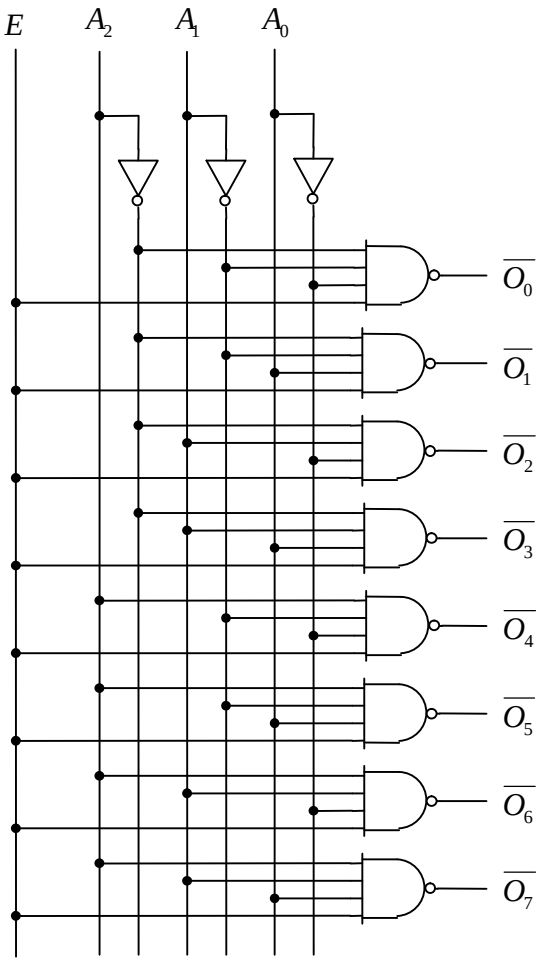
(ب) فاك شفرة من نوع 3 إلى 8 بخط سماح و خرج نشط منخفض (3-to-8 Decoder with Enable and Active Low Outputs)



E	A_2	A_1	A_0	$\overline{O_7}$	$\overline{O_6}$	$\overline{O_5}$	$\overline{O_4}$	$\overline{O_3}$	$\overline{O_2}$	$\overline{O_1}$	$\overline{O_0}$
0	×	×	×	1	1	1	1	1	1	1	1
1	0	0	0	1	1	1	1	1	1	1	0
1	0	0	1	1	1	1	1	1	1	0	1
1	0	1	0	1	1	1	1	1	0	1	1
1	0	1	1	1	1	1	1	0	1	1	1
1	1	0	0	1	1	1	0	1	1	1	1
1	1	0	1	1	1	0	1	1	1	1	1
1	1	1	0	1	0	1	1	1	1	1	1
1	1	1	1	0	1	1	1	1	1	1	1

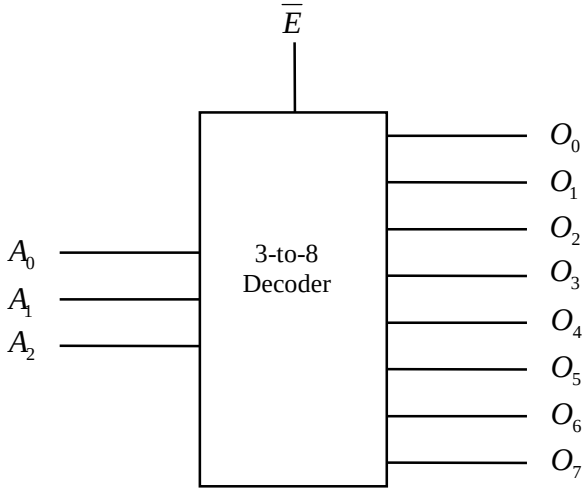
$$\overline{O_0} = \overline{E\,\overline{A_2}\,\overline{A_1}\,\overline{A_0}}$$
$$\overline{O_1} = \overline{E\,\overline{A_2}\,\overline{A_1}\,A_0}$$
$$\overline{O_2} = \overline{E\,\overline{A_2}\,A_1\,\overline{A_0}}$$
$$\overline{O_3} = \overline{E\,\overline{A_2}\,A_1\,A_0}$$
$$\overline{O_4} = \overline{E\,A_2\,\overline{A_1}\,\overline{A_0}}$$
$$\overline{O_5} = \overline{E\,A_2\,\overline{A_1}\,A_0}$$
$$\overline{O_6} = \overline{E\,A_2\,A_1\,\overline{A_0}}$$
$$\overline{O_7} = \overline{E\,A_2\,A_1\,A_0}$$

PDF created with pdfFactory Pro trial version www.pdffactory.com



PDF created with pdfFactory Pro trial version www.pdffactory.com

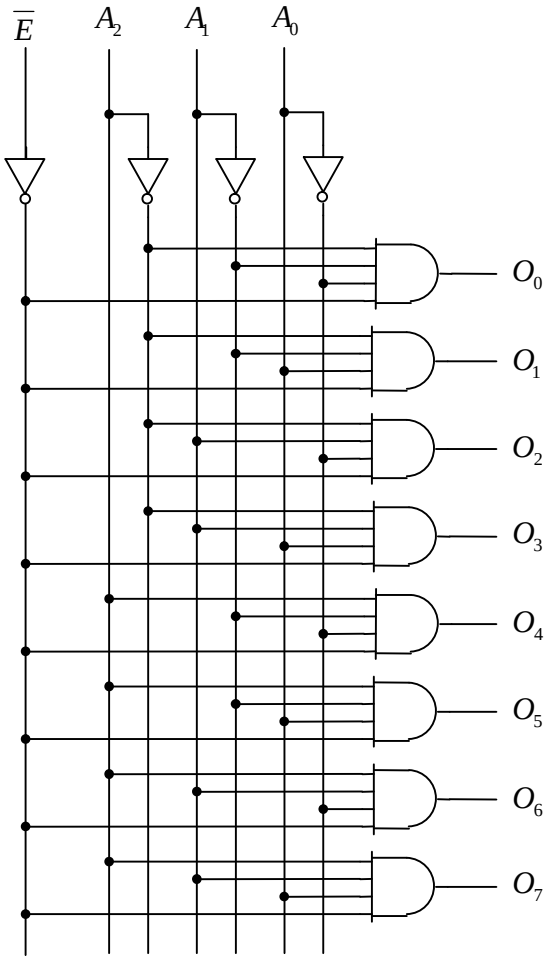
(ج) فاك شفرة من نوع 3 إلى 8 بخط سماح نشط منخفض (3-to-8 Decoder with Active Low Enable)



E	A_2	A_1	A_0	O_7	O_6	O_5	O_4	O_3	O_2	O_1	O_0
0	0	0	0	0	0	0	0	0	0	0	1
0	0	0	1	0	0	0	0	0	0	1	0
0	0	1	0	0	0	0	0	0	1	0	0
0	0	1	1	0	0	0	0	1	0	0	0
0	1	0	0	0	0	0	1	0	0	0	0
0	1	0	1	0	0	1	0	0	0	0	0
0	1	1	0	0	1	0	0	0	0	0	0
0	1	1	1	1	0	0	0	0	0	0	0
1	×	×	×	0	0	0	0	0	0	0	0

$$O_0 = \overline{E}\,\overline{A_2}\,\overline{A_1}\,\overline{A_0}$$
$$O_1 = \overline{E}\,\overline{A_2}\,\overline{A_1}A_0$$
$$O_2 = \overline{E}\,\overline{A_2}A_1\overline{A_0}$$
$$O_3 = \overline{E}\,\overline{A_2}A_1A_0$$
$$O_4 = \overline{E}\,A_2\overline{A_1}\,\overline{A_0}$$
$$O_5 = \overline{E}\,A_2\overline{A_1}A_0$$
$$O_6 = \overline{E}\,A_2A_1\overline{A_0}$$
$$O_7 = \overline{E}\,A_2A_1A_0$$

PDF created with pdfFactory Pro trial version www.pdffactory.com

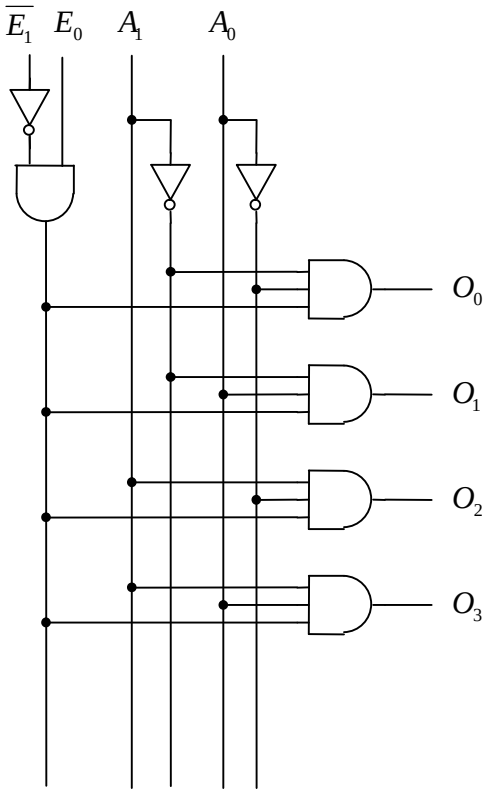


تدریب 10:

E_1	E_0	A_1	A_0	O_3	O_2	O_1	O_0
0	0	×	×	0	0	0	0
0	1	0	0	0	0	0	1
0	1	0	1	0	0	1	0
0	1	1	0	0	1	0	0
0	1	1	1	1	0	0	0
1	0	×	×	0	0	0	0
1	1	×	×	0	0	0	0

PDF created with pdfFactory Pro trial version www.pdffactory.com

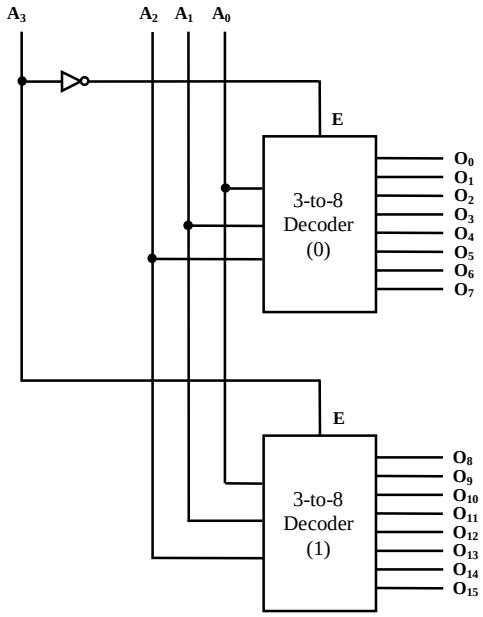
$$O_0 = \overline{E_1} E_0 \, \overline{A_1} \, \overline{A_0}$$
$$O_1 = \overline{E_1} E_0 \, \overline{A_1} A_0$$
$$O_2 = \overline{E_1} E_0 \, A_1 \overline{A_0}$$
$$O_3 = \overline{E_1} E_0 \, A_1 A_0$$



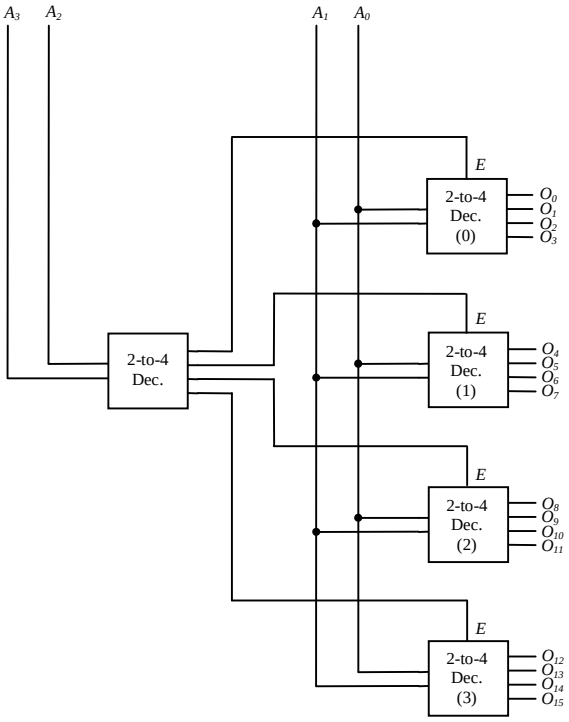
PDF created with pdfFactory Pro trial version www.pdffactory.com

تدريب 11:

(أ) بناء فاك شفرة من نوع 4 إلى 16 باستخدام دوائر فاك شفرة من نوع 3 إلى 8

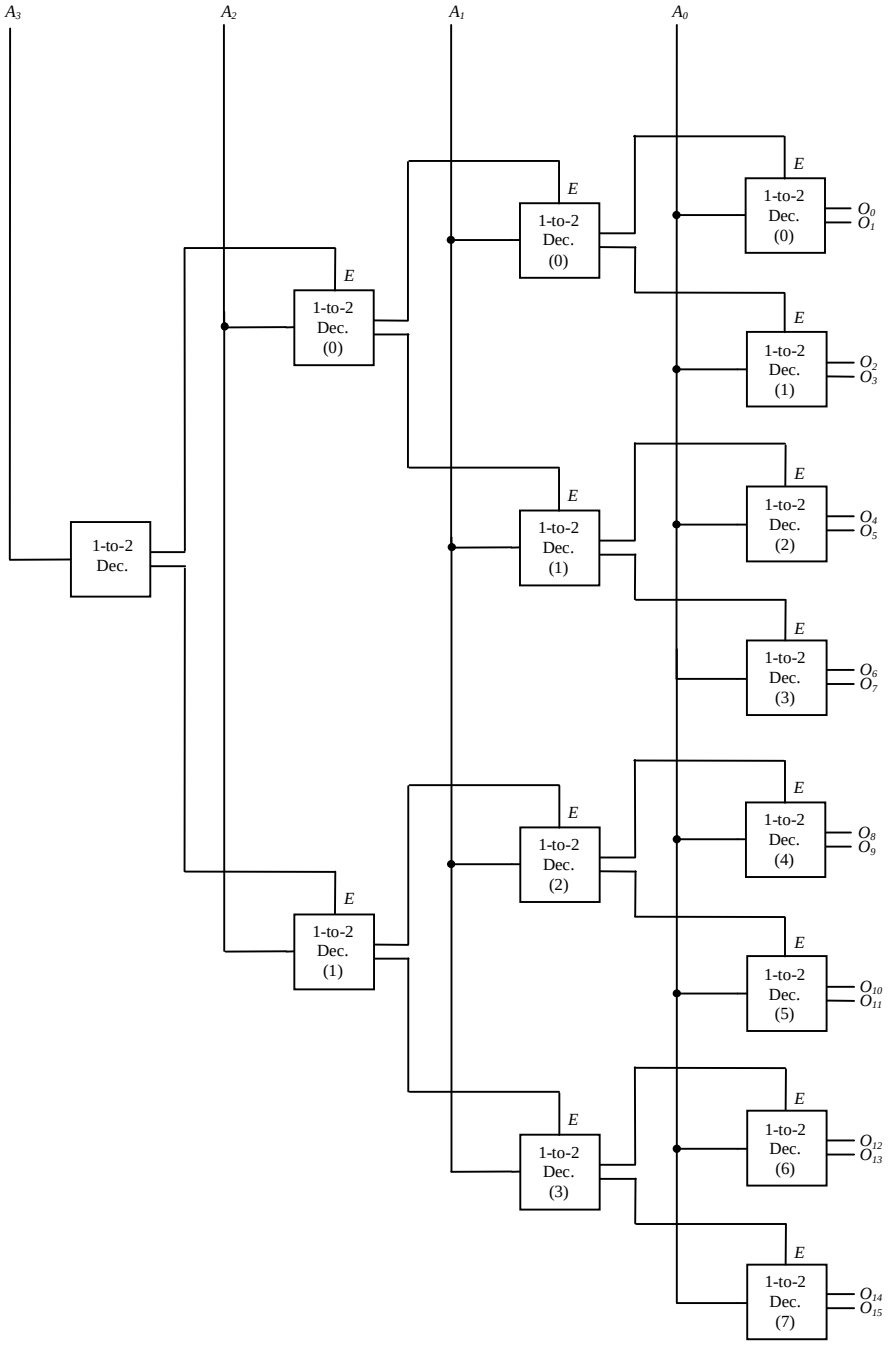


(ب) بناء فاك شفرة من نوع 4 إلى 16 باستخدام دوائر فاك شفرة من نوع 2 إلى 4



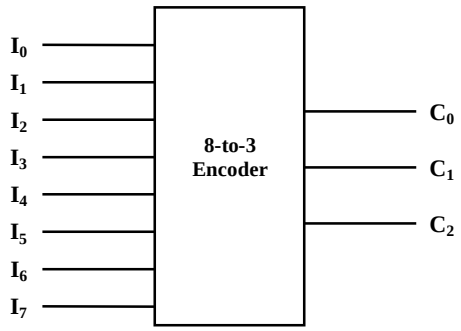
PDF created with pdfFactory Pro trial version www.pdffactory.com

(ج) بناء فاك شفرة من نوع 4 إلى 16 باستخدام دوائر فاك شفرة من نوع 1 إلى 2



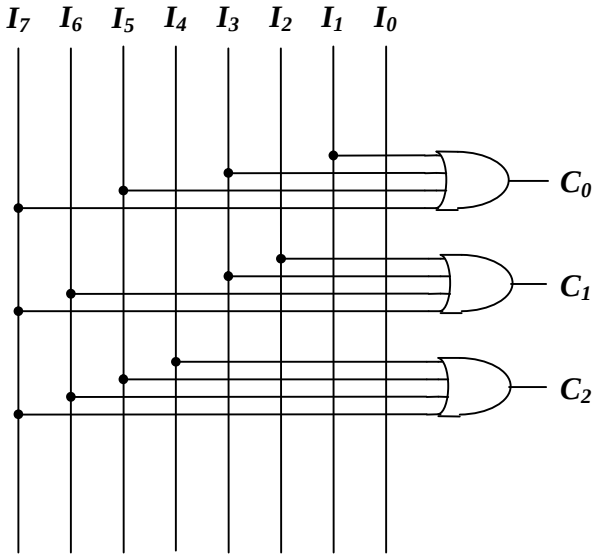
PDF created with pdfFactory Pro trial version www.pdffactory.com

تدریب 12:



I_7	I_6	I_5	I_4	I_3	I_2	I_1	I_0	C_2	C_1	C_0
0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	1	0	0	0	1	0
0	0	0	0	1	0	0	0	0	1	1
0	0	0	1	0	0	0	0	1	0	0
0	0	1	0	0	0	0	0	1	0	1
0	1	0	0	0	0	0	0	1	1	0
1	0	0	0	0	0	0	0	1	1	1

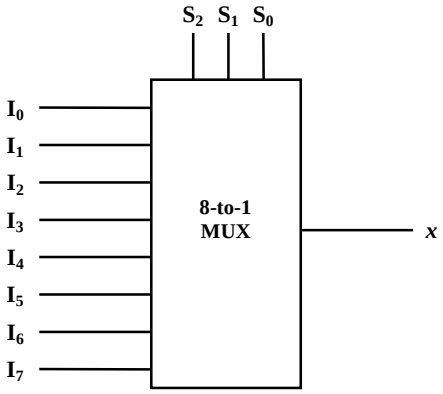
$$C_0 = I_1 + I_3 + I_5 + I_7$$
$$C_1 = I_2 + I_3 + I_6 + I_7$$
$$C_2 = I_4 + I_5 + I_6 + I_7$$



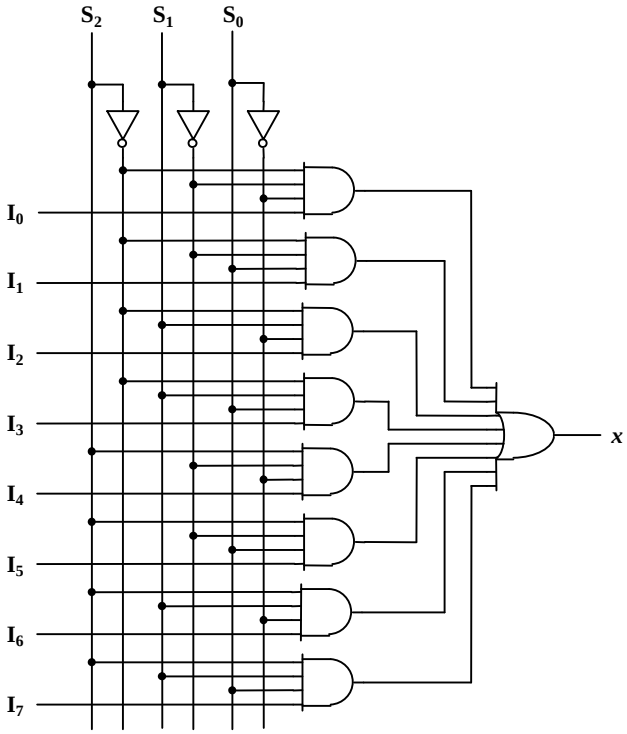
PDF created with pdfFactory Pro trial version www.pdffactory.com

تدریب 13:

S_2	S_1	S_0	x
0	0	0	I_0
0	0	1	I_1
0	1	0	I_2
0	1	1	I_3
1	0	0	I_4
1	0	1	I_5
1	1	0	I_6
1	1	1	I_7



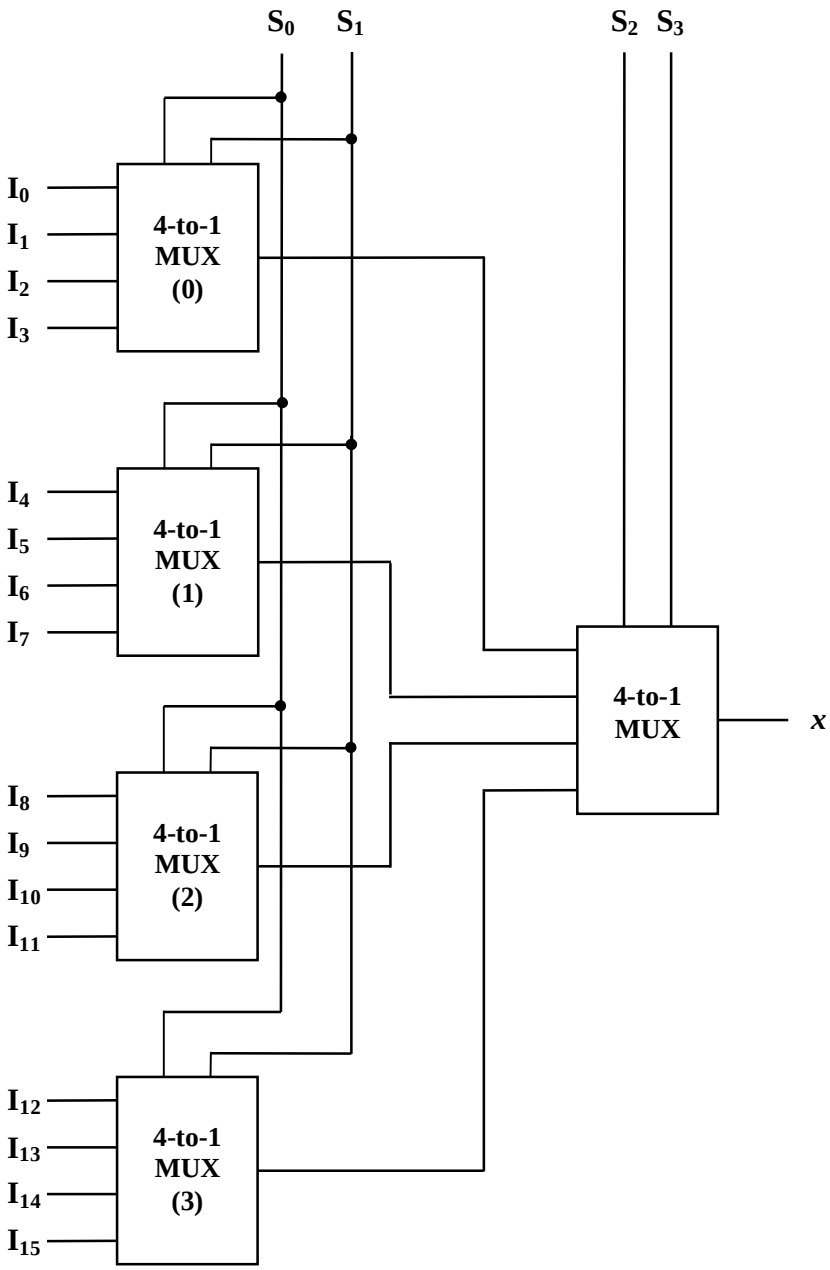
$$x = \overline{S_2}\,\overline{S_1}\,\overline{S_0} I_0 + \overline{S_2}\,\overline{S_1} S_0 I_1 + \overline{S_2} S_1 \overline{S_0} I_2 + \overline{S_2} S_1 S_0 I_3 + S_2 \overline{S_1}\,\overline{S_0} I_4 + S_2 \overline{S_1} S_0 I_5 + S_2 S_1 \overline{S_0} I_6 + S_2 S_1 S_0 I_7$$



PDF created with pdfFactory Pro trial version www.pdffactory.com

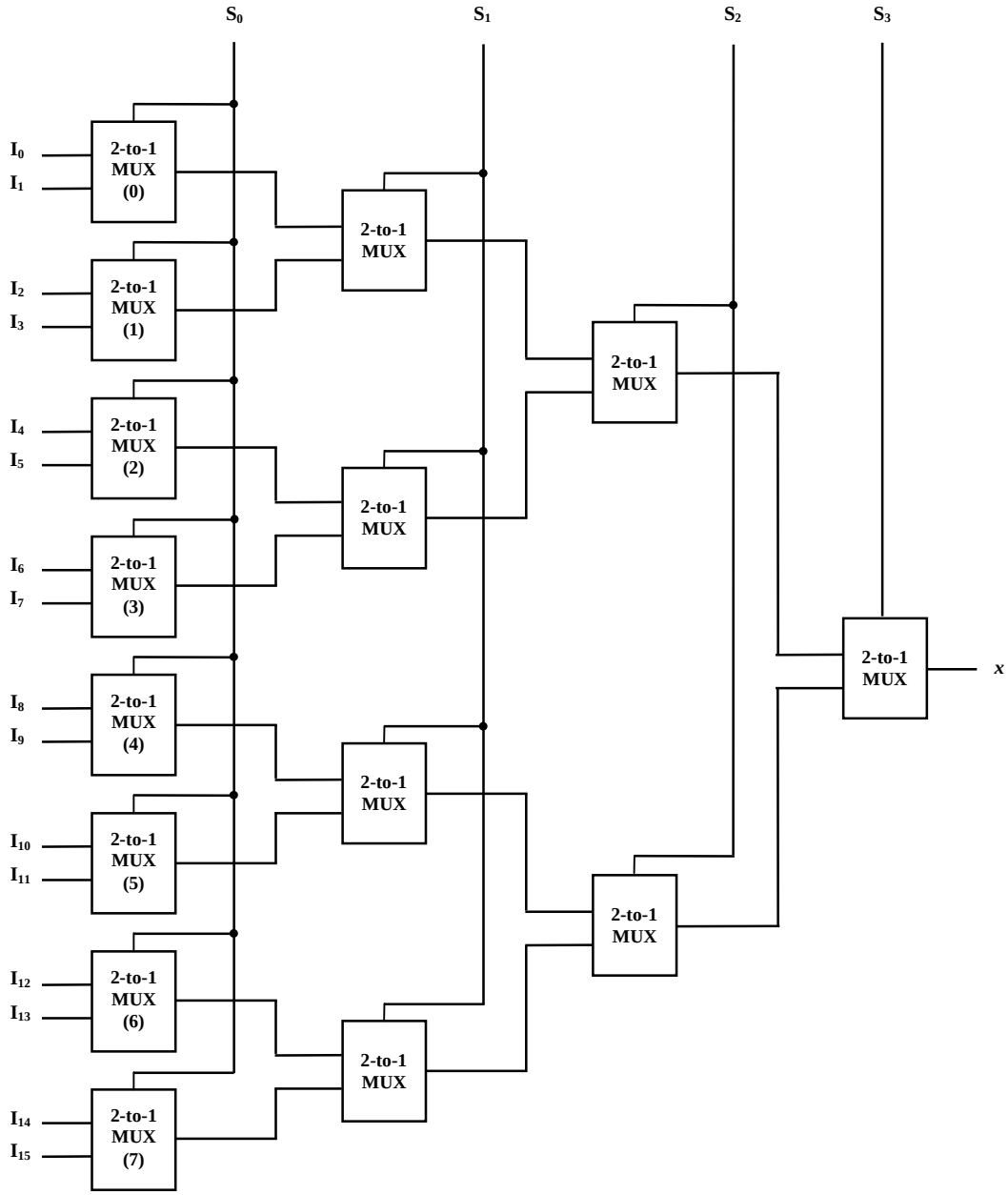
تدريب 14:

(أ) بناء دامج من نوع 16 إلى 1 باستخدام وحدات دامج من نوع 4 إلى 1



PDF created with pdfFactory Pro trial version www.pdffactory.com

(ب) بناء دامج من نوع 16 إلى 1 باستخدام وحدات دامج من نوع 2 إلى 1

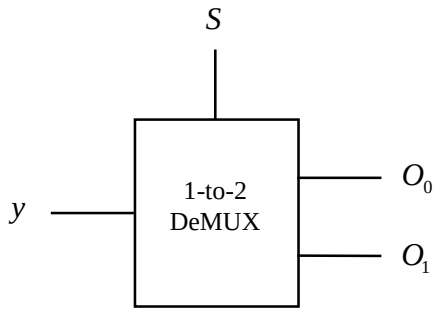


PDF created with pdfFactory Pro trial version www.pdffactory.com

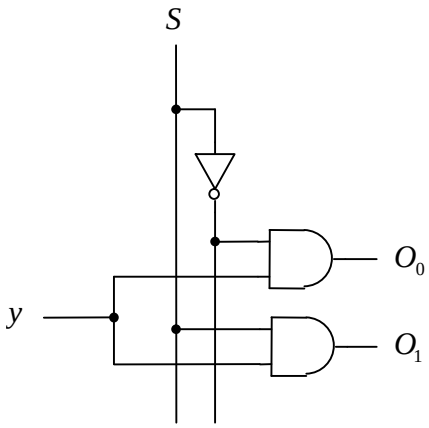
تدريب 15:

(أ) مفرق من نوع 1 إلى 2 (1-to-2 Demultiplexer)

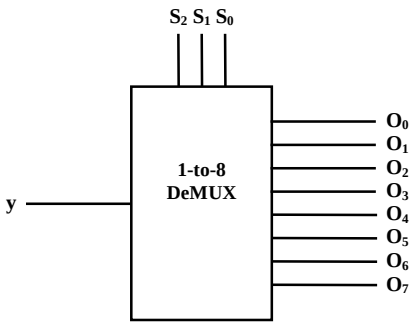
S	O_1	O_0
0	0	y
1	y	0



$$O_0 = \overline{S}y$$
$$O_1 = Sy$$



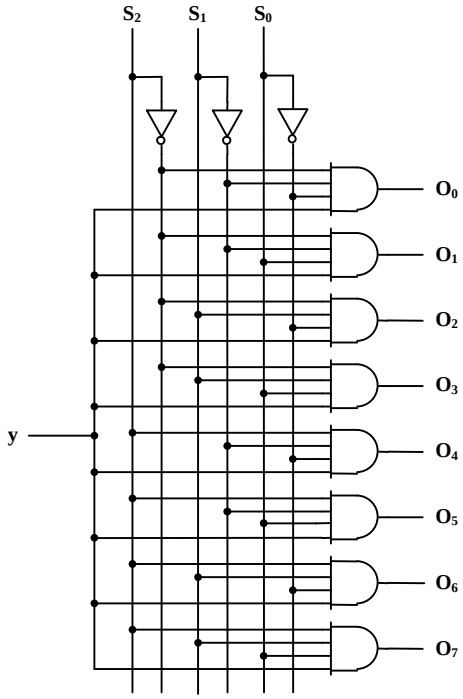
(ب) مفرق من نوع 1 إلى 8 (1-to-8 Demultiplexer)



PDF created with pdfFactory Pro trial version www.pdffactory.com

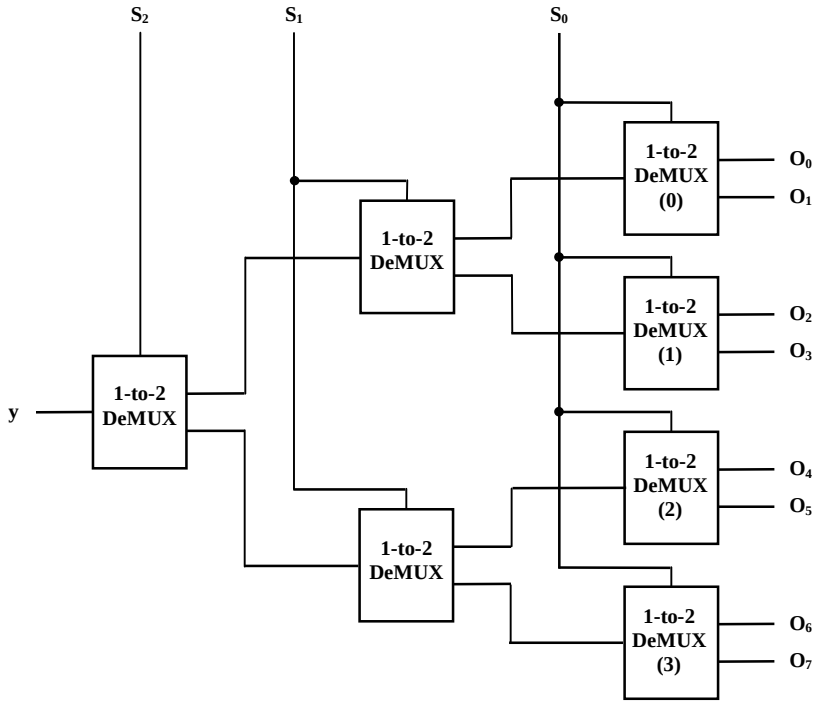
S_2	S_1	S_0	O_7	O_6	O_5	O_4	O_3	O_2	O_1	O_0
0	0	0	0	0	0	0	0	0	0	y
0	0	1	0	0	0	0	0	0	y	0
0	1	0	0	0	0	0	0	y	0	0
0	1	1	0	0	0	0	y	0	0	0
1	0	0	0	0	0	y	0	0	0	0
1	0	1	0	0	y	0	0	0	0	0
1	1	0	0	y	0	0	0	0	0	0
1	1	1	y	0	0	0	0	0	0	0

$$O_0 = \overline{S_2}\,\overline{S_1}\,\overline{S_0}y$$
$$O_1 = \overline{S_2}\,\overline{S_1}S_0y$$
$$O_2 = \overline{S_2}S_1\overline{S_0}y$$
$$O_3 = \overline{S_2}S_1S_0y$$
$$O_4 = S_2\overline{S_1}\,\overline{S_0}y$$
$$O_5 = S_2\overline{S_1}S_0y$$
$$O_6 = S_2S_1\overline{S_0}y$$
$$O_7 = S_2S_1S_0y$$



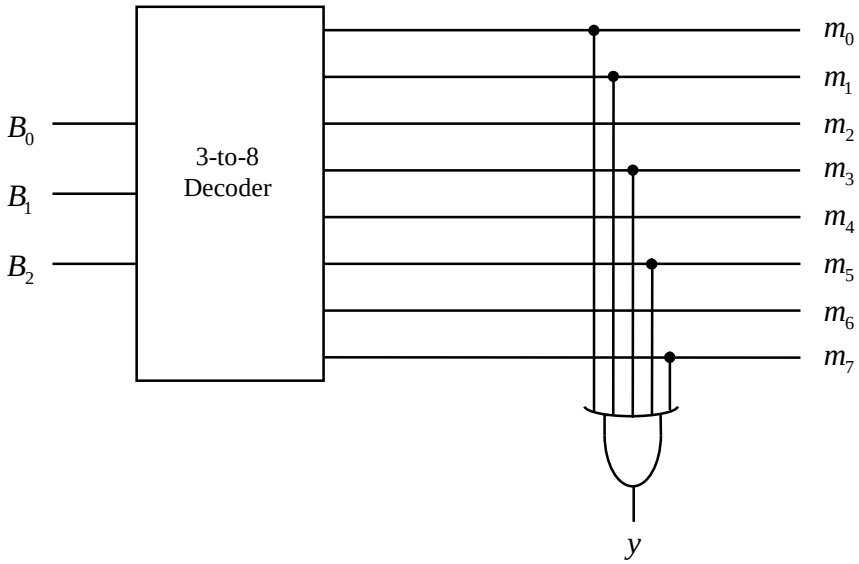
PDF created with pdfFactory Pro trial version www.pdffactory.com

تدريب 16:



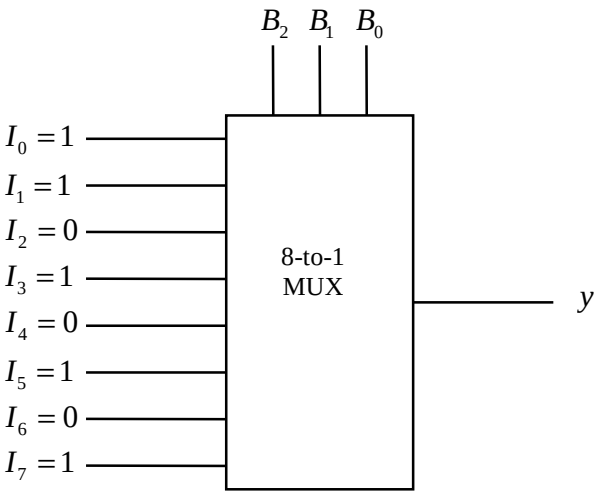
تدريب 17:

(أ) باستخدام فاك شفرة و مشفر (Decoder & Encoder)

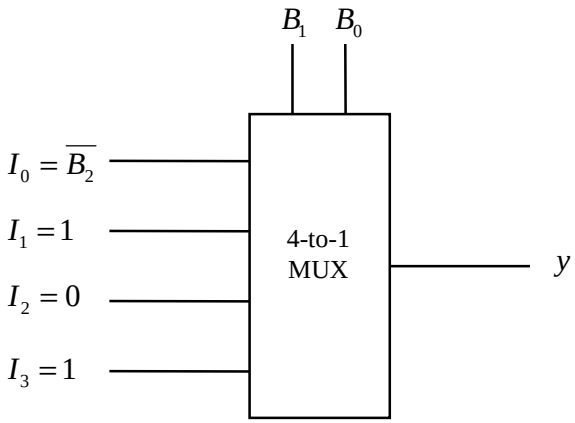


PDF created with pdfFactory Pro trial version www.pdffactory.com

(ب) باستخدام دامج من نوع 8 إلى 1 (8-to-1 MUX)



(ج) باستخدام دامج من نوع 4 إلى 1 (4-to-1 MUX)



PDF created with pdfFactory Pro trial version www.pdffactory.com