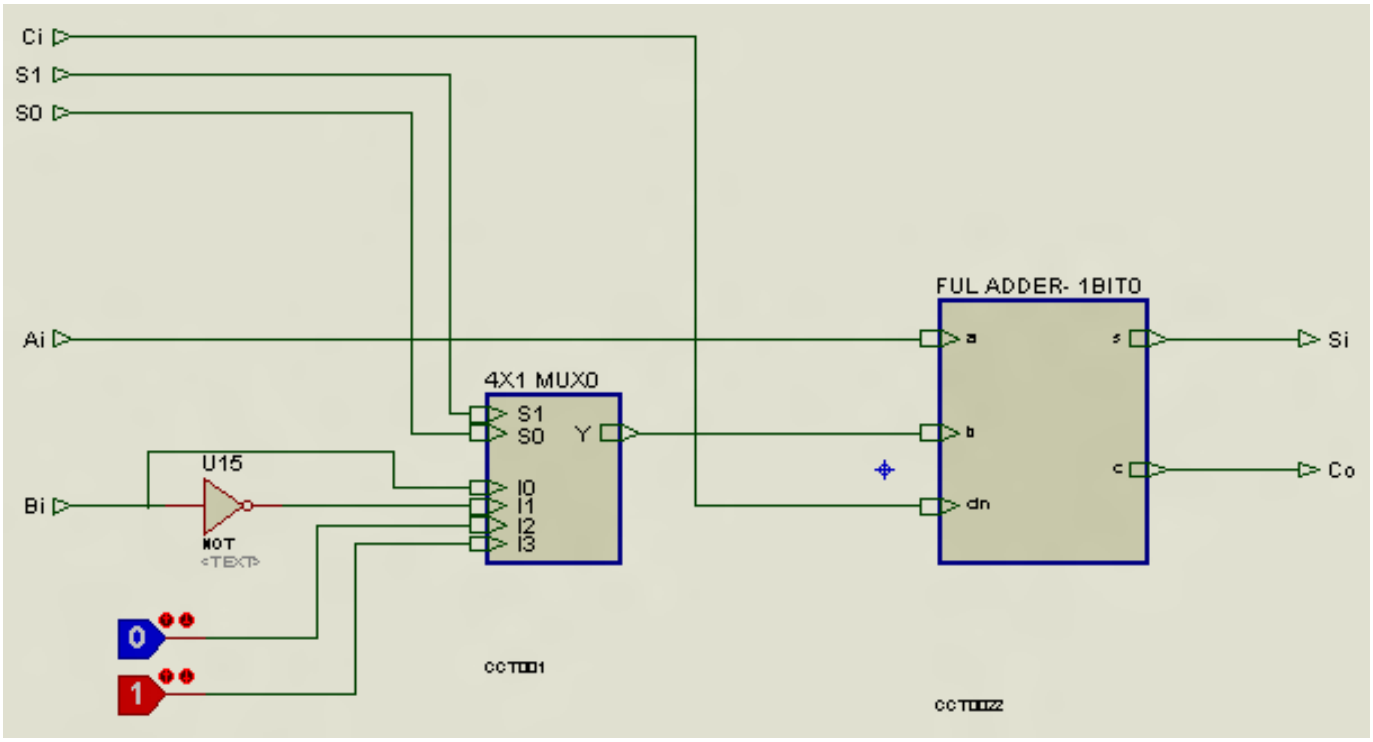
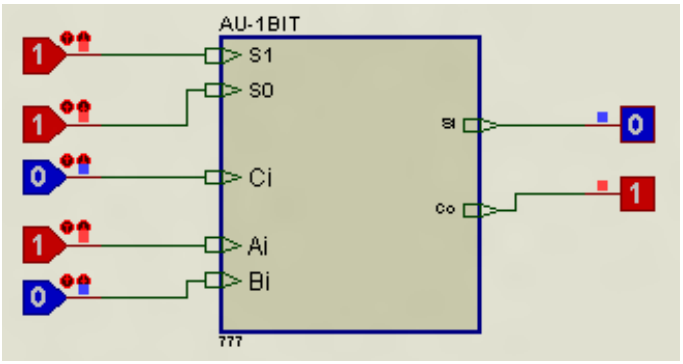
ثامناً: تصمم وحدة حساب 1-bit:

نقوم بذلك وفق ما هو مبين بالشكل (1):



الشكل (1)

ثم نقوم بتوضيب الدارة بصندوق[1] Subcircuit وفق الشكل (2):

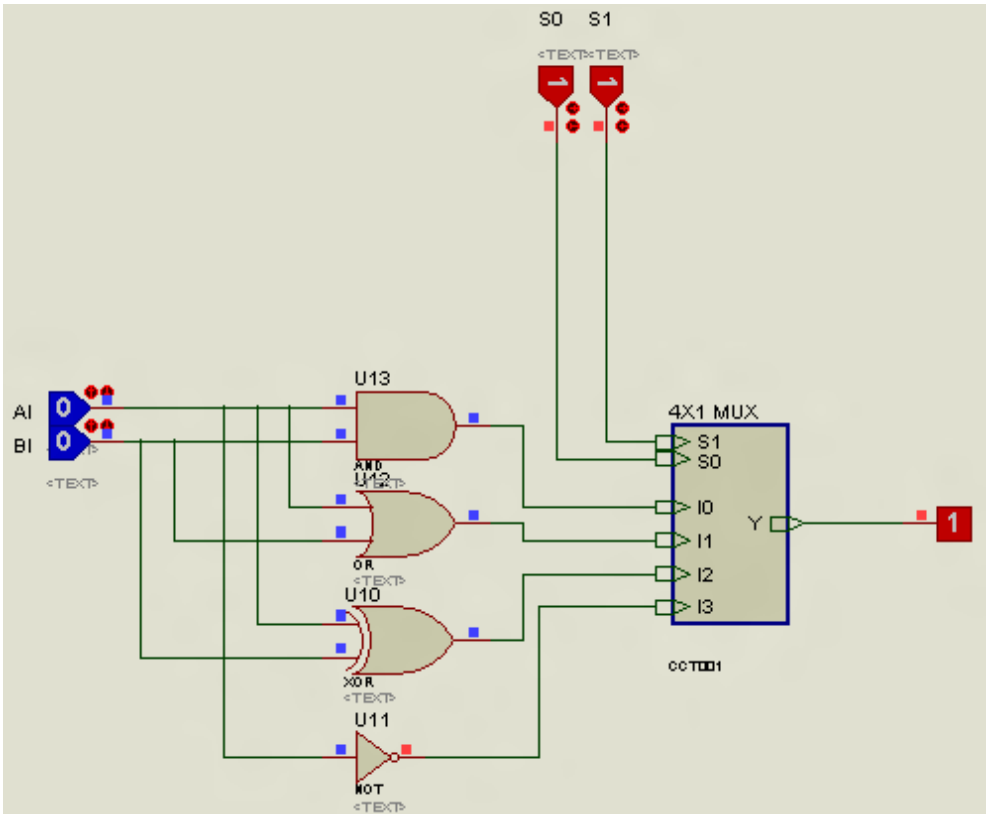


الشكل (2)

تاسعاً: وحدة منطق بطول 1-bit:

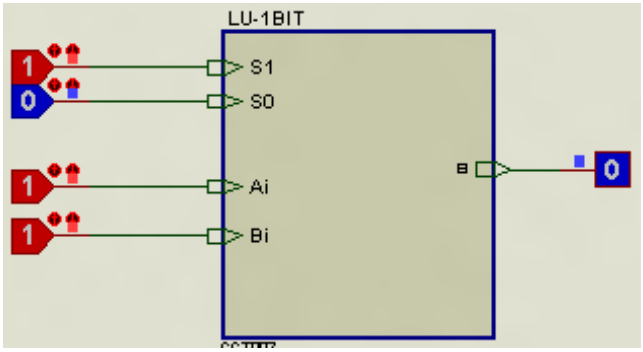
[1] كما تعلمنا سابقاً، فإنه عند توضيب الدارة بصندوق Subcircuit يجب مطابقة تسمية الأطرف بين الورقة الأب (Parent Sheet) والورقة الابن (Child Sheet).

سنعمل على تصميم وحدة منطق (Logic Unit: LU) قادرة على تنفيذ أربع عمليات منطقية، وهذه العمليات هي: AND، OR، XOR، NOT. سنحتاج لناخب لاختيار إحدى هذه العمليات حسب القيمة المطبقة على مداخله. ستكون الدارة وفق الشكل (3):



الشكل (3)

ثم نقوم بتوضيب الدارة بصندوق Subcircuit كما في الشكل (4):



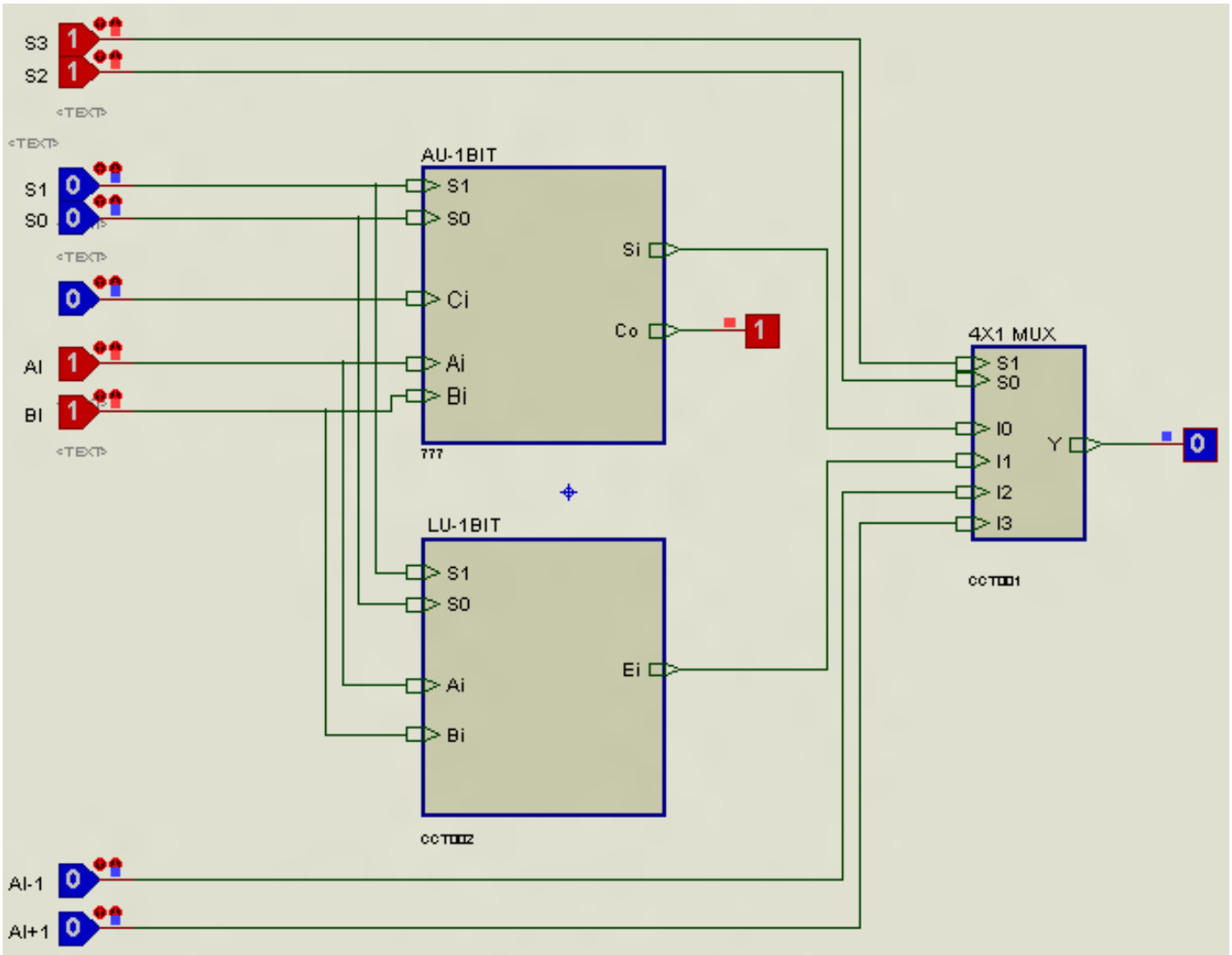
الشكل (4)

يتطابق عمل الدارة السابقة مع الجدول الوظيفي التالي:

S_1	S_0	Output	Operation
0	0	$E = A \wedge B$	AND
0	1	$E = A \vee B$	OR
1	0	$E = A \oplus B$	XOR
1	1	$E = \overline{A}$	Complement

عاشراً وأخيراً: تصميم وحدة حساب ومنطق بطول 1-bit (Arithmetic Logic Unit: ALU):

سيتم فيها دمج وحدة الحساب مع وحدة المنطق اللتين قد تم تصميمهما سابقاً في وحدة واحدة. تقوم ALU أيضاً بعملية إزاحة إلى اليمين أو إلى اليسار بمقدار خانة واحدة. نحتاج إلى ناخب للاختيار بين الدارة الحسابية أو الدارة المنطقية أو الإزاحة اليمينية أو الإزاحة اليسارية. يوضح ذلك الشكل (5):



الشكل (5)

يتطابق عمل الدارة السابقة مع الجدول الوظيفي التالي:

Operation select						
S_3	S_2	S_1	S_0	C_{in}	Operation	Function
0	0	0	0	0	$F = A$	Transfer A
0	0	0	0	1	$F = A + 1$	Increment A
0	0	0	1	0	$F = A + B$	Addition
0	0	0	1	1	$F = A + B + 1$	Add with carry
0	0	1	0	0	$F = A + \overline{B}$	Subtract with borrow
0	0	1	0	1	$F = A + \overline{B} + 1$	Subtraction
0	0	1	1	0	$F = A - 1$	Decrement A
0	0	1	1	1	$F = A$	Transfer A
0	1	0	0	×	$F = A \wedge B$	AND
0	1	0	1	×	$F = A \vee B$	OR
0	1	1	0	×	$F = A \oplus B$	XOR
0	1	1	1	×	$F = \overline{A}$	Complement A
1	0	×	×	×	$F = \text{shl } A$	Shift left A into F
1	1	×	×	×	$F = \text{shr } A$	Shift right A into F