সরলীকরণ ঃ

$$\bar{X} = \bar{B}C(\bar{A} + A) + \bar{A}B(\bar{C} + C)$$

$$= \bar{B}C + \bar{A}B$$

যোগের গুণ গঠন ঃ উপরের সমীকরণটিতে ডি-মরগ্যানের উপপাদ্য ব্যবহার করে,

$$X = \bar{\bar{X}}$$

$$= \overline{\bar{B}C + \bar{A}B} = \overline{\bar{B}C}\;\overline{\bar{A}B} = (\bar{\bar{B}} + \bar{C})(\bar{\bar{A}} + \bar{B}) = (B + \bar{C})(A + \bar{B})$$

$$= (B + \bar{C})(A + \bar{B})$$

৯.৬ (খ) নং চিত্রে বর্তনীটির যুক্তি চিত্র দেয়া হলো ঃ

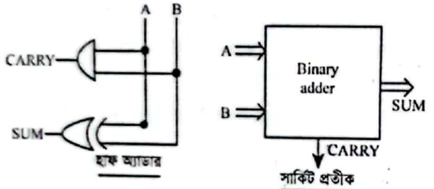
শুধু NAND বর্তনী ঃ এতক্ষণ মৌলিক গেইট দিয়ে যুক্তি বর্তনী বাস্তবায়নের উদাহরণ দেয়া হলো। সাধারণত NAND ও NO গেইট বহুল পরিমাণে পাওয়া যায় এবং AND ও OR গেইটের তুলনায় এসব গেইটের দামও কম। AND ও OR গেইট NAND NOR গেইটের তুলনায় বেশি শক্তিক্ষয় করে এবং তাদের সংকেত প্রবাহে বিলম্ব (Propagation delay) তুলনামূলকভাবে বে এসব কারণে অনেক সময় শুধু NAND (অথবা শুধু NOR) গেইট দিয়ে যুক্তি বর্তনী বাস্তবায়ন করা হয়।

৯.২ হাফ অ্যাডার ও হাফ সাবট্রাকটরের কার্যাবলি (Operation of half adder and Sub tractor) ঃ

● **হাফ অ্যাডার (Half adder) ঃ** দুটি বিট যোগ করার জন্য যে যোগের বর্তনী ব্যবহৃত হয় তাকে অর্ধ যোগের বর্তনী বা হা অ্যাডার (Half adder) বলে। দুটি বিট যোগ করে এই বর্তনী যোগফল (SUM) এবং হাতের সংখ্যা (Carry) প্রস্তুত করে। দুটি বি যোগ করার জন্য নিচের চারটি অবস্থা পাওয়া যায়–

(i) 0 + 0 = 0 (ii) 1 + 0 = 1 (iii) 0 + 1 = 1 (iv) 1+1 = 10

নিচের চিত্রে Half adder বর্তনী দেখানো হলো ঃ



চিত্র ঃ ৯.৭

হাফ অ্যাডারের আউটপুটে SUM এবং CARRY পাওয়া যায়, এর বুলিয়ান সমীকরণ নিম্নরূপ–

SUM = $A \oplus B$

CARRY = AB

এখানে SUM হলো B XOR B এবং CARRY হলো A and B

এই বর্তনীতে ক্যারি যোগের ব্যবস্থা থাকে না বলে একে হাফ অ্যাডার বলে।

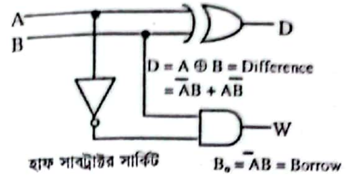
নিম্নের চিত্রে হাফ অ্যাডারের ট্রুথ টেবিল দেয়া হলো–

Input		Output	
A	B	Carry	Sum
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

ট্রুথ টেবিল হতে দেখা যাচ্ছে, যখন A = 0, B = 0 হয়, তখন Sum = 0 এবং Carry = 0 হয়। আবার যখন A = 0, B = 1 হ তখন Sum = 1 এবং Carry = 0 হয়। যখন A = 1, B = 0 তখন Sum = 1 এবং Carry = 0 হয়। আবার যখন A = 1, B = 1 তখ Sum = 0 এবং Carry = 1 হয়।

● **হাফ সাবট্রাকটর (অর্ধ বিয়োগের বর্তনী) ঃ**

অর্ধ বিয়োগের বর্তনী বা হাফ সাবট্রাক্টর দিয়ে দুটি বিটের বিয়োগফল এবং ধার বা বরো (Borrow) তৈরি করা হয়। নিম্নের ৯.৮ নং চিত্রে A ও B এই দুই বিটের বিয়োগফল D এবং বরো (Borrow) এর মান দেখানো হলো–



Input		Output	
A	B	D	W
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

ট্রুথ টেবিল

চিত্র ঃ ৯.৮

টেবিল হতে পাওয়া যায়–

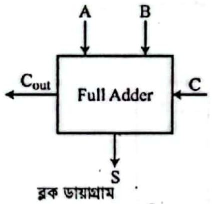
$$D = \bar{A}B + A\bar{B} = A \oplus B$$

$$W = \bar{A}B$$

৯.৩ ফুল অ্যাডার ও ফুল সাবট্রাক্টরের কার্যাবলি (Operation of full adder and subtractor) ঃ

ফুল অ্যাডার (Full adder) ঃ

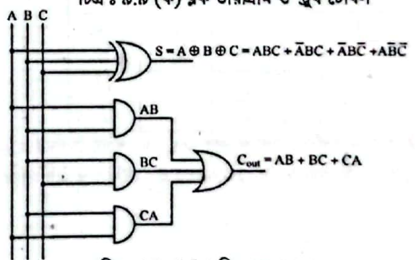
যে অ্যাডার সার্কিট তিনটি বাইনারি বিট (bit)-কে একত্রে যোগ করতে পারে, তাকে ফুল অ্যাডার বলে। এখানে তৃতীয় বিট সাধারণত ক্যারি বিট হয়।



Input			Output	
A	B	C	S	C_{out}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

ট্রুথ টেবিল

চিত্র ঃ ৯.৯ (ক) ব্লক ডায়াগ্রাম ও ট্রুথ টেবিল



চিত্র ঃ ৯.৯ (খ) লজিক ডায়াগ্রাম

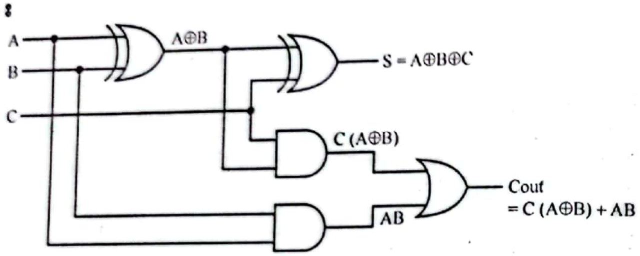
উপরের লজিক ডায়াগ্রাম A, B এবং C হলো Input লাইন এবং S (Sum) এবং Carry (C_{out}) হলো আউটপুট লাইন। এখানে তিনটি বিট (Bit) যোগ করার জন্য X–OR গেইট ব্যবহার করা হয়েছে এবং তিনটি ইনপুট থেকে AND গেইট অপারেশনের পর OR গেইটের আউটপুট হতে Final Carry (C_{out}) আউটপুটে পাওয়া যায়। ট্রুথ টেবিল অনুসারে Sum (S) এবং Carry (C) নির্ণয় হয়। যদি A = B = C = 0 হয়, তবে S = 0 এবং C = 0 হবে। আবার A = 1, B = 0, C = 0 হলে, S = 1 এবং C = 0 হবে। এক্ষেত্রেও হাফ অ্যাডার-এ বর্ণিত বিষয়টি বিবেচনা করা যায়। যেমন ঃ A = 0, B = 0, C = 1 হলে, A + B + C = 1 (sum) এবং Carry (C) = 0 আবার, A = 1, B = 1 এবং C = 0 হলে, A + B + C = 1 + 1 + 0 = 10 এক্ষেত্রে, S = 0 এবং C = 1 যদি A = B = C = 1 হয়, তবে A + B + C = 1 + 1 + 1 = 11. তাহলে, S = 1, C = 1.

Sum ও Carry সমীকরণ ঃ

$$\text{Sum} = \bar{A}\bar{B}C + \bar{A}B\bar{C} + A\bar{B}\bar{C} + A\bar{B}\bar{C} + ABC$$
$$= \bar{A}(\bar{B}C + B\bar{C}) + A(\bar{B}\bar{C} + BC)$$
$$= \bar{A}(B \oplus C) + A(\overline{B \oplus C})$$
$$= A \oplus B \oplus C$$

$$\text{Carry} = \bar{A}BC + A\bar{B}C + AB\bar{C} + ABC$$
$$= C(\bar{A}B + A\bar{B}) + AB(C + \bar{C})$$
$$= C(A \oplus B) + AB$$

লজিক সার্কিট ঃ



চিত্র ঃ ৯.৯ (গ) ফুল অ্যাডার লজিক সার্কিট

ফুল সাবট্রাক্টর (Full Subtractor) ঃ পূর্ণ বিয়োগের বর্তনী দিয়ে দুই বিটের বিয়োগফল ও ধার নির্ণয়ের সময় পূর্ববর্তী স্থান হ সৃষ্ট ধার বিবেচনা করা হয়।

নিচে পূর্ণ বিয়োগের ট্রুথ টেবিল দেয়া হলো–

A	B	C	D	W_o
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

এখানে C হলো পূর্ববর্তী স্থান হতে আগত অর্থাৎ গ্রহণমুখ ধার এবং W_o হলো নির্গমন ধার। বিয়োগের সময় বেশি গুরুত্বের পরবর্তী স্থান হতে ধার করলে নির্গমন ধারও সৃষ্টি করতে হয়। বিয়োগের ট্রুথ টেবিল B এর সাথে C যোগ করে A হতে সেই যোগফল বিয়োগ করা হয়।

সারণি হতে,

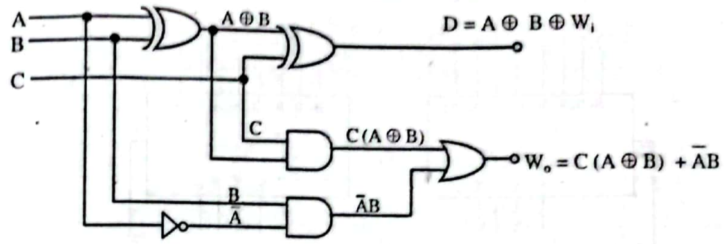
$$D = \bar{A}\bar{B}C + \bar{A}B\bar{C} + A\bar{B}\bar{C} + ABC$$
$$W_o = \bar{A}\bar{B}C + \bar{A}B\bar{C} + \bar{A}BC + ABC$$

সমীকরণ দুটিকে নিম্নের গঠনে লেখা সম্ভব ঃ

$$D = C(AB + \bar{A}\bar{B}) + \bar{C}(\bar{A}B + A\bar{B})$$
$$= C(\overline{A \oplus B}) + \bar{C}(A \oplus B)$$
$$= A \oplus B \oplus C$$

$$W_o = C(\bar{A}\bar{B} + AB) + \bar{A}B$$
$$= C(\bar{A}B + A\bar{B}) + \bar{A}B$$
$$= C(A \oplus B) + \bar{A}B$$

উপরের সমীকরণ অনুযায়ী পূর্ণ বিয়োগের জন্য যুক্তি বর্তনী ৯.১০ নং চিত্রে দেখানো হলো।

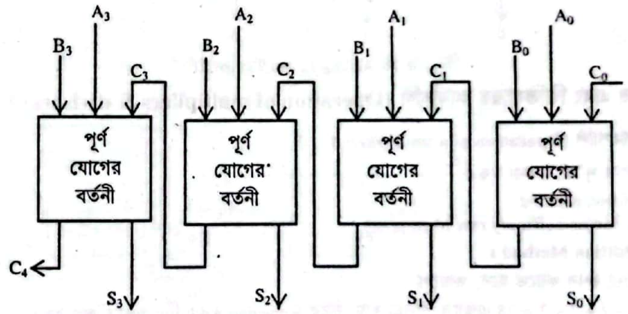


চিত্র ঃ ৯.১০ ফুল সাবট্রাক্টর সার্কিট

৯.8 4-বিট প্যারালাল অ্যাডার সার্কিট (4-bit parallel adder circuit)/4 bit যোগ বর্তনী ঃ

নিম্নের ৯.১১ নং চিত্রে A_3 A_2 A_1 A_0 এবং B_3 B_2 B_1 B_0 এই দুটি 4 বিট বাইনারি সংখ্যা যোগের জন্য একটি সমান্তরাল যোগকারী দেখানো হলো। চারটি পূর্ণ যোগের বর্তনী দিয়ে এই যোগকারী বর্তনী বাস্তবায়ন করা হয়েছে। N-বিটবিশিষ্ট সংখ্যা যোগের জন্য N সংখ্যক পূর্ণ যোগের বর্তনী দরকার। সংখ্যা দুটির বিটগুলো একসাথে সরবরাহ করা হয় বলে এটিকে সমান্তরাল যোগকারী (Parallel adder) বলা হয়। সমান্তরাল যোগকারী একীভূত বর্তনী হিসেবে পাওয়া যায়।

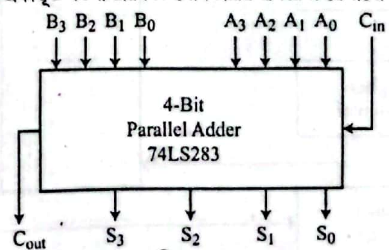
7483 ও চার বিট আইসি. সমান্তরাল যোগকারীর উদাহরণ।



চিত্র ঃ ৯.১১ 4-bit parallel adder

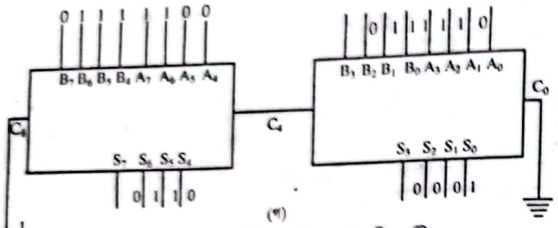
● 74LS283 ও সমান্তরাল যোগকারী ঃ

নিচের ৯.১২ নং চিত্রে এই TTL একীভূত বর্তনী চার বিট যোগকারীর রূপরেখা দেখানো হলো–



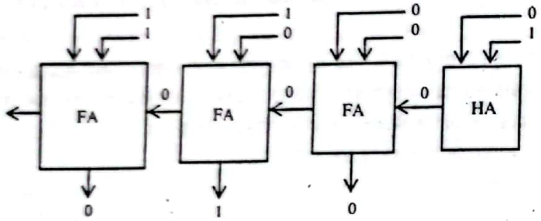
চিত্র ঃ ৯.১২

এই বর্তনী দুটি চার বিট সংখ্যা ও একটি গ্রহণ ক্যারি যোগ করে চার বিট যোগফল ও নির্গমন ক্যারি তৈরি করে। সংখ্যা দুটি ও গ্রহণ ক্যারি যোগফল 0 ও 15 এর মধ্যে থাকলে $C_4 = 0$ থাকে এবং যোগফল $S_3\ S_2\ S_1\ S_0$ রেখায় পাওয়া যায়। যোগফল 16 ও ... এর মধ্যে থাকলে $C_4 = 1$ হয় এবং S রেখাসমূহের বাইনারি মান যোগফলের চেয়ে 16 কম হয়। চার বিটের চেয়ে বড় সংখ্যা যোগের জন্য একাধিক 7483 বর্তনীর অনুক্রম সংযোগ (Serial adder) সম্ভব, যা চিত্র ৯.১৩ নং এ দেখানো হলো। এখানে প্রথম 7483 হতে নির্গত C_4 কে পরবর্তী 7483 বর্তনীর C_0 রেখায় সংযুক্ত করতে হয়। এভাবে এই বর্তনীতে দুটি ৪ বিট বাইনারি সংখ্যার যোগ দেখানো হয়েছে।



চিত্র ঃ ৯.১৩ 7483 দিয়ে ৪ বিট যোগকারী বর্তনী

9 এবং 12 যোগ করে কীভাবে যোগফল 21 পাওয়া যায় তার সমতুল্য বাইনারি অ্যাডার সার্কিট নিম্নে দেখানো হলো ঃ



চিত্র ঃ ৯.১৪ Adding 12 and 9 to get 21

৯.৫ মাল্টিপ্লায়ার এবং ডিভিসরের কার্যাবলি (Operation of multipliers & divisors) ঃ

মাল্টিপ্লায়ারের কার্যপ্রণালি (Operation of a multiplier) ঃ

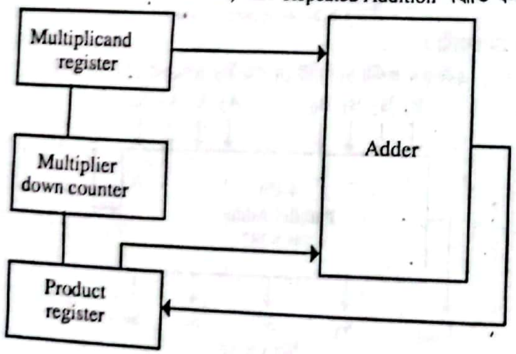
গুণের কাজ সাধারণত দু'ভাবে করা হয় ঃ

1. Repeated Addition Method
2. Add and Shift Method (Binary rate Multiplier).

■ **Repeated Addition Method ঃ**

১। ধরি 7 কে 4 দ্বারা গুণন করতে হবে, তাহলে–

$7 \times 4 = 28$ কে $7 + 7 + 7 + 7 = 28$ এভাবে পাওয়া যায়, যাকে Repeated Addition পদ্ধতি বলা হয়।



$7 \rightarrow$ Multiplicand
$\times 4 \rightarrow$ Multiplier
$28 \rightarrow$ Product

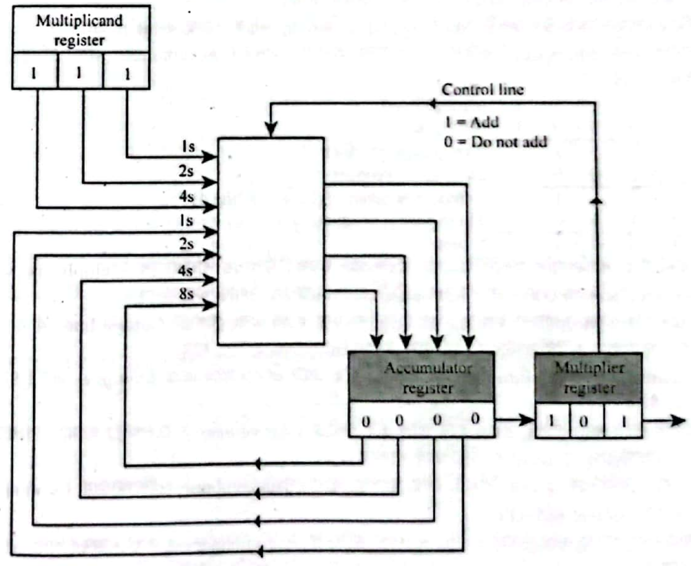
Multiplicand $7 + 7 + 7 + 7 = 28$ Product

$111 = 7$
$\times 100 = 4$
000
$000\times$
$111\times$
$11100 = 28$

	Load with binary	After 1 down count	After 2 down counts	After 3 down counts	After 4 down counts
Multiplicand register	111	111	111	111	111
Multiplier counter	100	011	010	001	000
Product register	00000	00111	01110	10101	11100
	Load				Stop

এই পদ্ধতির গুণনের একটি ব্লক চিত্র দেখানো হয়েছে, যাকে গুণন করতে হবে অর্থাৎ Multiplicand কে উপরের রেজিস্টারে রাখা হয়েছে। এই উদাহরণে এর মান দশমিকে 7 এবং বাইনারিতে 111 এবং গুণক (Multiplier) এর মান 4 বা বাইনারিতে 100 নিচের ব্লকটিতে গুণফল জমা থাকে, যাকে Product register নামে দেখানো হয়েছে। এর গুণন প্রক্রিয়াটি উক্ত টেবিলে দেখানো হয়েছে। শুরুতে Multiplicand register এর মান 111 আছে, Down counter এর মান 100 আছে, Product রেজিস্টারের মান 00000 আছে এবং Multiplicand register এর মান ও Product regisiter এর মান Adder ckt- এ গিয়ে $111 + 00000 = 00111$ হবে এবং এই মান আবার P রেজিস্টারে আসবে তখন Down counter এর মান হবে 011 (3); আবার P রেজিস্টার এর মান এবং M রেজিস্টার এর মান Adder বর্তনীতে গিয়ে যোগ হয়ে 01110 (14) হয়ে P রেজিস্টারে আসবে। এভাবে Down counter 0 হলে P রেজিস্টারে সর্বশেষ যোগফল পাওয়া যাবে 11100 (28)। এটি সাধারণত বেশি ব্যবহৃত হয় না। কারণ এতে সময় অনেক বেশি লাগে। ব্যবহারিক ক্ষেত্রে বেশি ব্যবহৃত হয় Add and shift method বা Binary rate multiplier method. এই পদ্ধতির গুণন প্রক্রিয়া ব্লক চিত্রের সাহায্যে ব্যাখ্যা করা হলো ঃ

■ **Add and Shift Method (Binary rate Multiplier)**



চিত্র ঃ ৯.১৫ (a) বাইনারি রেট মাল্টিপ্লায়ারের সাহায্যে গুণন প্রক্রিয়া