**Контроль арифметических и логических операций.**

Для контроля арифметических операций обычно используется числовой контроль по модулю $q$. Возможность проверки правильного результата выполнения операций сложения и умножения с помощью контрольного кода, представленного в СОК, вытекает из теорем 2.1 и 2.2.
При контроле операции деления используется зависимость вида

$$a = c \cdot b + r \ ,$$

где a – делимое, c – частное, b – делитель и r – остаток от деления, т.е. в процедуре контроля сравнивается контрольный код делимого с результатом вычисления правой части выражения, содержащего только допустимые в СОК операции.

На рис.2.2. представлена структурная схема операционного блока с числовым контролем по модулю (схему можно модифицировать с учетом контроля операции деления, что предлагается сделать в качестве упражнения). Ее аппаратурная избыточность определяется схемами формирования контрольного кода и контролирующего операционного блока, сложность которых зависит от значения $q$.
Этим же значением определяется полнота контроля. При увеличении значения модуля увеличивается число кратных ошибок, обнаруживаемых системой контроля, однако, при этом возрастает и сложность кодирующей и контролирующей аппаратуры. Например, в качестве модуля $q$ не следует выбирать основание системы счисления, используемой для представления основной информации, если она является однородной и позиционной. Это приводит к тому, что контролируются только младшие разряды чисел (последнее уже отмечалось). Избыточность будет минимальной при условии $q = r \pm 1$, где $r$ – основание системы счисления, т.е. для СВТ значение $q$ принимается равным «3». Ограничения на класс обнаруживаемых ошибок для СВТ широкого назначения при этом считаются приемлемыми ..

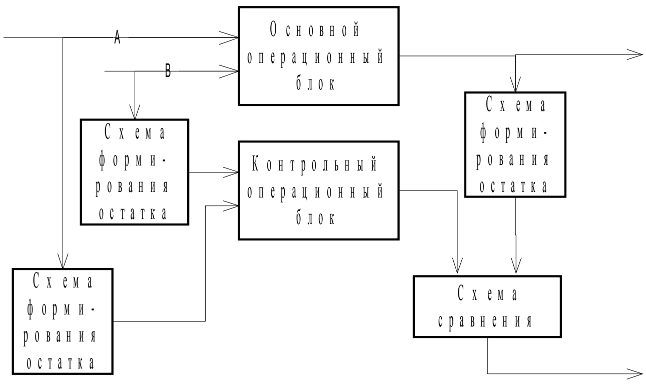


Рис.2.2.

Не смотря на то, что аналоги теорем для суммы цифр операндов и результата сложения и умножения отсутствуют, для контроля арифметических операций можно использовать цифровой контроль, что поясняется следующим примером.
Пусть контрольным кодом является остаток по модулю 2 суммы единиц в каждом из операндов.
При сложении чисел $a$ и $b$ разряды суммы $S$ образуются следующим образом:

$$\begin{cases} S_1 = a_1 \oplus b_1 \oplus P_1; \\ S_2 = a_2 \oplus b_2 \oplus P_2; \\ \ldots \\ S_n = a_n \oplus b_n \oplus P_n. \end{cases}$$

где $S_i, a_i, b_i, P_i, i = \overline{1, n}$ – соответственно разряды суммы, операндов и переноса; $\oplus$ – операция сложения по модулю 2.

Тогда, сложив все $n$ равенств по модулю 2 $|S|_2^c = |a|_2^c \oplus |b|_2^c \oplus |P|_2^c$ ,т.е. при правильном результате сложения четность суммы его цифр должна совпадать со значением, вычисленным согласно правой части приведенного выражения.
Применение такого контроля может быть оправдано специализацией СВТ, например, в случае, когда в системе команд предусмотрена только операция сложения и для контроля пересылок используется цифровой контроль по модулю 2. Контроль логических операций
К логическим принято относить операции сдвига, а также операции and, or, not, xor. Последние, начиная с and, могут применяться либо к так называемым логическим значениям true (истина) и false (ложь), кодами которых при n-разрядном представлении являются единица в младшем разряде при остальных нулях и все нули соответственно, либо ко всем разрядам произвольных операндов поразрядно.
Модели, с помощью которых могли бы строиться избыточные коды для контроля логических операций, вообще говоря, отсутствуют. Для теории кодов, исправляющих ошибки, такая задача (впрочем, как и контроль арифметических операций) не является предметом анализа. Применение используемой выше модели, точнее построение контрольного кода как отображения в СОК, жестко ограничено теоремами 2.1 и 2.2 (здесь «жестко ограничено» означает, что, например, для контроля сдвига такая возможность есть, поскольку операция сдвига

– это умножение на $p^n$ , где p – основание системы счисления, а n – количество разрядов, на которое сдвигается код информации). Перечень подобных «экзотических» приемов можно продолжить, рассматривая операцию *xor* как поразрядную сумму операндов по модулю «2» без учета переносов и корректируя в соответствии с этим выражение (2.1), что, однако, не применимо к другим операциям, и т.д.
Единственным очевидным обобщением может быть рекомендация строить избыточный код как разделимый. В этом случае всегда остается возможность использовать в качестве контрольного кода копию кода информации, т.е. дублировать информацию (собственно, представление контрольного кода в виде отображения ее в СОК – это то же дублирование, но не полное и со «сжатием» копии).

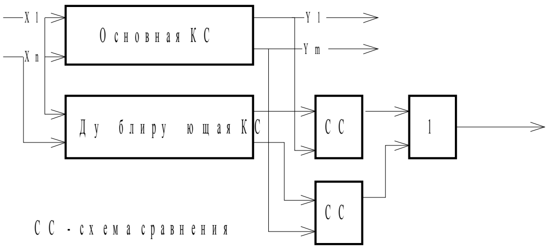


Рис..2.3.