

ДИАГНОСТИЧЕСКОЕ ОБЕСПЕЧЕНИЕ СИСТЕМЫ УПРАВЛЕНИЯ ПРОЦЕССОМ ВЫРАЩИВАНИЯ МОНОКРИСТАЛЛОВ

В промышленном производстве крупногабаритных монокристаллов в настоящее время широко используются установки типа "РОСТ", на которых практически реализованы процессы управляемой кристаллизации из расплава на затравку с использованием для управления информации о положении уровня расплава в тигле [1]. Существенная продолжительность рабочего цикла, требования все большего ограничения вмешательства человека в ростовой процесс в условиях промышленного производства определяют необходимость создания диагностического обеспечения многопроцессорных систем управления (МПСУ) процессом кристаллизации, решающего задачи тестового и функционального диагностирования.

Функциональное диагностирование системы управления построено на основе сигнатурного мониторинга управляющих программ контроллеров [2,3]. Для реализации метода сигнатурного мониторинга решаются три задачи: разбиение управляющих программ микроконтроллера на сегменты и нахождение оптимального числа контрольных точек, или точек возврата, программ для обнаружения неисправностей; вычисление эталонных сигнатур для каждой контрольной точки и определение минимального объема информации для повторной "прокрутки" программного сегмента; восстановление работоспособности МПСУ. В МПСУ проверка правильности функционирования микроконтроллеров (МК) в отдельных режимах и на отдельных участках выполнения управляющих программ осуществляется диагностическим ядром МК, что является по существу развитием метода наблюдателей Люенбергера, где под системами наблюдения за правильностью функционирования понимают линейные участки управляющих программ. Алгебраические инварианты на этих участках - время выполнения сегментов программы и коды сегментов.

Структурная схема диагностического ядра (ДЯ), представлена на рис. 1. Через блок управления (БУ) ДЯ осуществляется сопряжение по интерфейсному каналу с МК. В начале выполнения определенного сегмента управляющей программы МК блок управления загружает из памяти (ЗУ) эталонные значения инвариантов этого сегмента в регистр контрольной метки (РКМ) и регистр эталонных временных сигнатур (РЭС). В РЭС загружаются эталонные значения минимального и максимального времени выполнения каждого сегмента программы T_{min} и T_{max} .

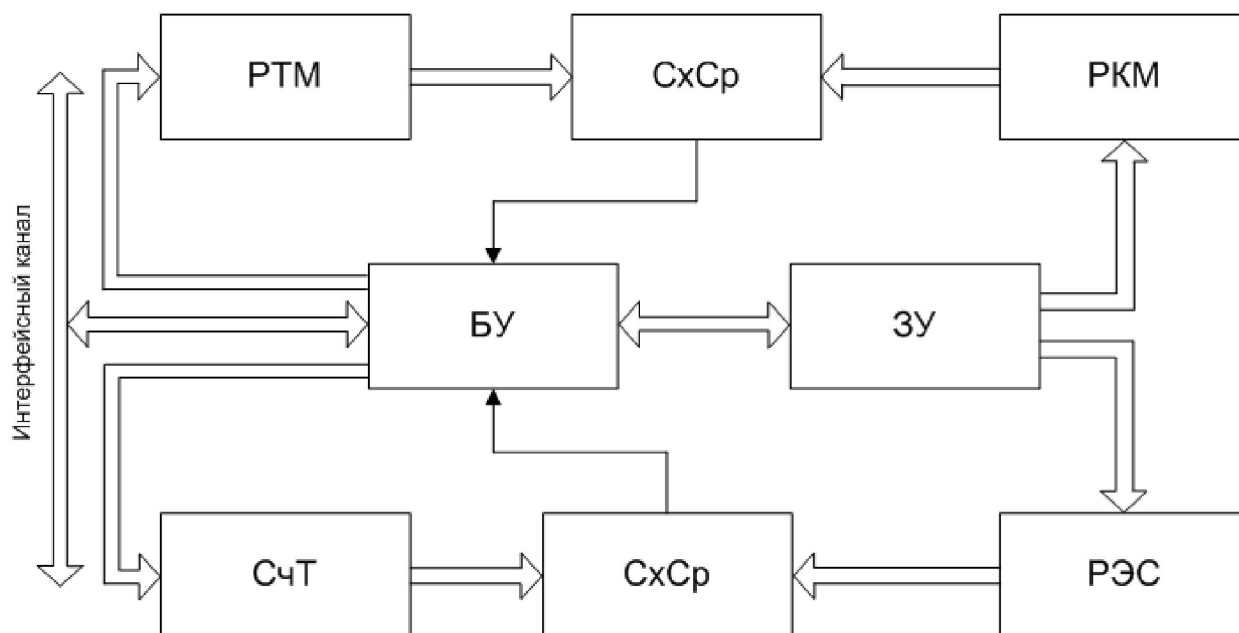


Рис. 1 Структурная схема ДЯ микроконтроллера

Выполнение сегмента управляющей программы сопровождается проверкой текущей метки обрабатываемого сегмента путем сравнения содержимого регистра текущей метки (РТМ) с содержимым РКМ, измерением времени выполнения программного сегмента счетчиком времени СчТ и сравнением с эталонной сигнатурой в РЭС. В СчТ записывается реальное время выполнения сегмента программы. Проверка времен выполнения сег-

мента программы осуществляется по команде БУ при изменении номера сегмента программы.

В сигнатурном мониторинге на основе контроля времени выполнения линейных сегментов управляющих программ число точек возврата (ТВ) должно быть достаточным для минимизации времени обнаружения ошибок. В то же время большое количество ТВ и сокращение интервала между ними увеличивает время выполнения управляющей программы и необходимый объем памяти МК. Исходя из этого, следует добиваться оптимального размещения точек возврата, что позволяет сократить объем памяти, необходимый для сохранения промежуточных состояний МК, и временные затраты на процедуру восстановления.

Рассмотрим процедуру нахождения оптимального числа точек возврата в управляющей программе МК. Критерий оптимизации – минимизация латентного периода обнаружения ошибки. На рис. 2 изображен линейный сегмент управляющей программы МК, который разбит на n интервалов, идентифицированных метками M_i . Управляющая программа написана на языке высокого уровня. Каждая метка является потенциальной точкой возврата в программе, поэтому для каждой метки в памяти МК запоминается состояние процессора, то есть объем памяти МК определяет максимальное количество меток в управляющей программе.

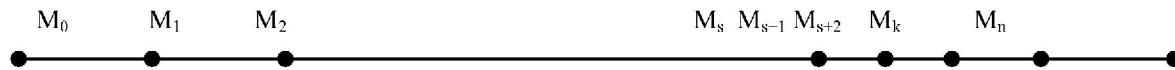


Рис. 2 Линейный участок управляющей программы МК

Пусть T_d – максимально допустимое время выполнения сегмента управляющей программы МК, обусловленное характеристиками управляемого оборудования. Предположим, что при выполнении управляющей программы мы находимся в точке с меткой M_s , при этом на прохождение программы от точки M_0 до точки M_s затрачено время T_s . Следовательно на выполнение программы от метки M_s до метки M_n должно быть затрачено время T_{n-s} , такое что

$$T_{n-s} < T_d - T_s \quad (1)$$

Определим время выполнения программы от метки M_s до метки M_n . Осуществим прогноз. Пусть после старта от метки M_s выполнение программы дойдет до метки M_{s+1} и вернется некоторое число раз к метке M_s для восстановления работоспособности программы МК. Будет пройден один интервал, т.е. $k=1$, за время t_{s+1} . Считаем, что в худшем случае, остальные интервалы программы будут выполнены за такое же время. При этом время T_{n-s} будет определяться суммой по t_{s+1} от $s+1$ до n . Если полученное значение T_{n-s} удовлетворяет условию (1), то выбираем следующей точкой возврата точку M_{s+1} и запускаем управляющую программу МК, если нет, то следовательно устанавливаем $k=1, 2, 3, \dots$ и для каждого k рассчитываем время T_{n-s} до тех пор, пока не выполнится условие (1). Полученное значение k будет определять количество интервалов программы до очередной точки возврата. Например, если в результате прогноза получим $k=3$ в точке возврата M_s , то следующая точка возврата находится в M_{s+3} .

В общем случае время T_{n-s} рассчитывается по выражению

$$T_{n-s} = \sum_{i=1}^{\left[\frac{n-s}{k} \right]} t_{s+k \cdot i} + \sum_{j=\left[\frac{n-s}{k} \right]+1}^n t_j ; k = 1, 2, 3, \dots, n,$$

t_{s+ki} , t_j – расчетное время выполнения интервала программы; s, j – номер ТВ; $[...]$ – целая часть от числа в скобках.

Метод расчета времени t_i выполнения интервала программы приведен в [4].

Предложенный метод динамического размещения контрольных точек в управляющих программах микроконтроллеров МПСУ минимизирует латентный период обнаружения неустойчивых неисправностей перемежающегося типа и сбоев и легко реализуется на современных МК.

Литература

1. Горилецкий В.И., Гринев Б.В., Заславский Б.Г., Смирнов Н.Н., Суздаль В.С. Рост кристаллов. – Харьков: АКТА, 2002. – 535 с.
2. Дербунович Л.В., Суздаль В.С., Тавровский И.И., Темников И.Н. Отказоустойчивые микроконтроллеры на основе сигнатурного мониторинга // Інформаційно-керуючі системи на залізничному транспорті. – Харків: ХарДАЗТ, 2002. - №4,5 (37). - С.71-73.
3. Заявка № а200704669 Украина. Устройство для регулирования роста монокристаллов и устройство для его диагностирования / Суздаль В.С., Епифанов Ю.М., Герасимчук Л.И., Дербунович Л.В. и др.. Приор. 26.04.2007.
4. Ярмола Е.А., Дербунович Л.В., Войтенко М.С. Метод динамического размещения контрольных точек в управляющих программах микроконтроллеров // Вестник Национального технического университета "Харьковский политехнический институт". - Харьков: НТУ "ХПИ", 2005. - №17. - С.100-103.