

0.1. *Пермяшкин Д.А.* Разрешение конфликтов процессов в процесс-ориентированной программе путем коррекции моделей процесса

Микроконтроллеры в виде программируемых логических контроллеров (ПЛС) являются основным вычислительным устройством, используемым при автоматизации производств, и за последние 40 лет активного обсуждения не было предложено приемлимой замены. Причиной для популярности данной темы является постоянно растущая сложность производственных процессов на фабриках и невозможность человека управлять данными процессами на лету самостоятельно. И раз тема популярная, то логично появление стандартов для автоматизации производств - IEC 61131 [1], который стандартизирует ПЛС и языки программирования для них.

Дальнейшим результатом обсуждения данной темы в 2015 году стала Индустрия 4.0 (или же Четвертая Промышленная Революция) [2], чьи идеи подняли требования к автоматизации производств на новый уровень. Стали появляются различные языки, нацеленные на выполнение идей Индустрии 4.0. Например, процесс-ориентированное программирование, разработанное Институтом автоматики и электрометрии Сибирского отделения Российской академии науки и апробированное на реальном производстве [3]. Основная идея процесс-ориентированной парадигмы заключается в представлении программы как большого числа мелких процессов, представляемых в виде модифицированного конечного автомата и выполняющихся по внешнему событию.

Но просто организовать параллельные вычисления недостаточно для автоматизации производства, требуется минимизировать ущерб в случае отказа (и просто минимизировать отказы). Дополнительно учтем низкую вычислительную мощность ПЛС, идеи Индустрии 4.0 по уменьшению роли человека (вплоть до нуля) в управлении сложными киберфизическими системами, и получим очень сложную задачу по организации параллельных вычислений.

Данная задача не является новой и существуют уже алгоритмы решения конфликтов между параллельными процессами при доступе к разделяемым ресурсам [4]. Но данные алгоритмы решают конфликты путем детерминирования поведения при доступе, чего недостаточно [5]. Поэтому автором в данной работе предлагается провести коррекцию модели процесса в процесс-ориентированном программировании путем замены входного состояния в конечном автомате (представляющим процесс) на множество входных состояний. Эта коррекция не решит абсолютно все конфликты, целью данной коррекции является разрешение конфликтов, когда оба конфликтующих процесса и читают, и пишут в общую переменную (порядок исполнения процессов в общем случае тогда не определен [5]).

Данная коррекция разрешает конфликты ещё до их появления, потому что в процесс-ориентированном программировании процесс может поменять состояние другого процесса только если другой процесс находится в состоянии покоя или ошибки [3]. И используя пример с клапаном из [5], при попытке запустить процесс открытия клапана пока он закрыт - мы получим стандартную ошибку, как будто мы пытались открыть клапан дважды.

Основным минусом данной коррекции является удаление основного стимула для автора процесс-ориентированной программы декомпозировать промышленный процесс на как можно меньшие процессы, что может привести к потере основного преимущества парадигмы.

Научный руководитель — д-р техн. наук, доц. Зюбин В. Е.

Список литературы

- [1] TIEGELKAMP M., JOHN K.H. IEC 61131-3: Programming industrial automation systems. / Berlin, Germany: Springer, 2010.
- [2] PHILBECK T., DAVIS N. The fourth industrial revolution // Journal of International Affairs. 2018. Vol. 72. N. 1. P. 17–22.
- [3] ZYUBIN V.E. ET AL. poST: A process-oriented extension of the IEC 61131-3 structured text language // IEEE Access. 2022. Vol. 10. P. 35238-35250.
- [4] ANDERSON J.H., KIM Y.J., HERMAN T. Shared-memory mutual exclusion: major research trends since 1986 // Distributed computing. 2003. Vol. 16. N. 2. P. 75-110.
- [5] PERMIASHKIN D.A. Towards conflict resolution methods in process-oriented programs // 2023 IEEE 24th International Conference of Young Professionals in Electron Devices and Materials (EDM). IEEE, 2023. P. 1790-1793.