

КІНЦЕВІ АВТОМАТИ В СХЕМАХ ЛОГІКИ УПРАВЛІННЯ

Пермяков О.А., Приходько О.Ю., Сліпченко С.Є.

Національний технічний університет

«Харківський політехнічний інститут», м. Харків

У дійсний період часу кінцеві автомати застосовуються ведучими фірмами світу задля програмування логічних контролерів, що програмуються, а також для опису поведінки окремих об'єктів у об'єктно-орієнтованому програмуванні. Вони використовуються при програмуванні протоколів, ігор та схем логіки, що програмується, а також у такій традиційній для їх застосування галузі, як створення компіляторів.

Доцільним є використання STATE-технології при здійсненні управління складними технологічними системами та виробничими автоматичними лініями.

У якості основного у STATE-технології використовується поняття "внутрішній стан". Стани розглядаються як деякі абстракції, що вводять на початку процесу алгоритмізації, наприклад шляхом однозначного зіставлення кожного з них з одним з фізичних станів об'єкту, що управляється, тому що звичайно "функціонування виробничих систем виявляє себе через зміну їх станів". При цьому кожний стан в алгоритмі підтримує об'єкт у відповідному стані, а перехід у новий стан у алгоритмі приводить до переходу об'єкта у новий відповідний стан, що і забезпечує процес логічного управління об'єктом. Наприклад, об'єкт "технологічна система" може знаходитись у одному з чотирьох робочих станах ("наладочний режим", "перехідний режим", "робочий режим", "відключений стан"), кожне з яких може підтримуватись відповідним станом в алгоритмі управління.

Для технологічної системи "з пам'яттю" алгоритм управління може мати і меншу кількість станів. При необхідності в алгоритм управління можуть бути введені також і інші стани, що пов'язані, наприклад, з несправностями об'єкту та неправильними діями оператора, що враховують різного роду блокування. Зв'язок станів з внутрішніми (управляючими) змінними з'являється далі на етапі кодування станів, що відсутній у традиційному програмуванні. При цьому число вводимих управляючих змінних залежить від прийнятого вигляду кодування. Такий підхід, відомий з теорії автоматів, принципово відрізняється від підходу, що звичайно застосовується у програмуванні, при якому у ході процесу програмування при необхідності вводяться внутрішні (звичайно двійкові) змінні, а потім кожний різний набір їх значень об'являється станом програми. Однак, так як поняття "стан" у програмуванні прикладних задач зазвичай не використовується, то відповідь на питання про кількість станів у програмі, що містить, наприклад, n двійкових внутрішніх змінних, залишається у більшості випадків відкритим. Треба зауважити, що у цьому випадку кількість станів може знаходитись у діапазоні від n до $2n$. Так як любий стан може бути представлений двома значеннями (логічних нуля чи одиниці). При розв'язанні задачі повинні бути враховані особливості деталей, що оброблюються, технологічні можливості верстатів та передбачені алгоритми мінімізації виробничих ресурсів.