مجلة جامعة تشرين للدراسات والبحوث العلمية - سلسلة العلوم الهندسية المجلد (20) العدد (6+7) 1998
Tishreen University Journal for Studies and Scientific Research –Engineering Science Series Vol. (20) No (6+7) 1998

تحقيق العمليات التتابعية باستخدام الحاكمات المبرمجة

د.محمد عبد الكريم عالية*

د.محمد أبو زلطة**

(قبل للنشر في 1998/9/10)

☐ ملخص ☐

يتضمن هذا البحث محاولة تهدف إلى استقصاء تصميم أنظمة التحكم المنفصلة التتابعية التي تتضمن حاكمات برمجية. تضمنت الورقة خمسة برامج مختلفة تحقق خمس عمليات تتابعية شائعة الاستخدام في الصناعة.

بيّن البحث إمكانية استخدام دالة تفاضل الحافة الصاعدة من أجل تحويل عملية الانتقال من حالة التوقف إلى حالة التشغيل إلى نبضة تؤثر فقط خلال زمن يساوي مسحة واحدة للحاكم المنطقي، وإمكانية الاستفادة من هذه النبضة لإجراء عملية العدّ اللازمة بوساطة العداد الصاعد المتضمن في الحاكم المنطقي المبرمج.

ومن أجل تحقيق عمليات تتابعية دورية اقترح استعمال مؤقت أحادي الاستقرار من أجل قسمة أي عدد من النبضات أو الدورات على أي عدد صحيح. وبعد تحرّي جميع البرامج المتضمنة في البحث تبين أنها تعمل بدقة وتحقق الأهداف المحددة لها.

* أستاذ مساعد – كلية عمان الجامعية للهندسة التطبيقية – عمان – الأردن.

** أستاذ مساعد – كلية عمان الجامعية للهندسة التطبيقية – عمان – الأردن.

مجلة جامعة تشرين للدراسات و البحوث العلمية - سلسلة العلوم الهندسية المجلد (20) العدد (6+7) 1998
Tishreen University Journal for Studies and Scientific Research - Engineering Science Series Vol. (20) No (6+7) 1998

REALIZING SEQUENTIAL PROCESSES USING PROGRAMMABLE LOGIC CONTROLLERS (PLCS)

Dr. Mohammad A.K. Alia*
Dr. Mohammad Abu Zalata**

❐ RESUMÉ ❐

This paper includes an effort aimed at investigating the design of sequential discrete control systems using programmable logic controllers. Five different programmes are designed to realize five, commonly used in industry, sequential processes.

It was found, that it is possible to use the function of the leading edge differentiation to convert the OFF to ON, transition of the output to a one scan output pulse that shall be fed to the counting input of the (PLC) up counter.

To realize cyclic sequential operations, it was suggested to use a monostable timer, thus making it possible to divide any number of pulses or periods by any required integer number. Experimental results show that all the designed programmes are operating.

*Amman University College for Applied Engineering – Amman - Jordan.
**Amman University College for Applied Engineering – Amman - Jordan.

مقدمة:

تقوم أنظمة التحكم الآلي المنفصلة (Discrete systems) بتنفيذ سلسلة من العمليات المحددة مع وجود شرط محدد من أجل الشروع بتنفيذ أية عملية. وعندما يكون للعملية المنفصلة بداية ونهاية ونمط تحكم محدد، تسمى بالعملية التتابعية. وتصنف العمليات التتابعية إلى عمليات تبدأ بالاستفادة من الزمن، أي تقاد بالزمن، وأخرى تبدأ بسبب حدوث حدث (Event). أما العمليات المقادة بالزمن، فيتم فيها البدء بتشغيل كل خطوة من خطوات العملية في زمن محدد أو بعد انقضاء فترة زمنية محددة، بينما يتم في العمليات المقادة بالحدث تشغيل كل خطوة بسبب حدوث ما في نظام التحكم. وفي معظم الحالات تتكون العمليات التتابعية من كل النوعين المذكورين معاً.

وعندما نتحدث عن تتابع معين، فإن المقصود بذلك انتقال المخرج (Output) من أحد وضعيه (عالي أو منخفض، مغلق أو مفتوح، عامل أو متوقف عن العمل) إلى الوضع الآخر عدداً من المرات. وفي كثير من الصناعات والأنظمة الإنتاجية والخدمية يلزم تكرار هذا التتابع لعمليات معينة وبنمط محدد من أجل الحصول على نتاج بمواصفات معينة.

ونظراً للتطور السريع في تكنولوجيا الحاكمات المنطقية المبرمجة، سواء من ناحية الكيان المادي أو الكيان المرن، إضافة إلى انخفاض سعرها المستمر، فقد وجد لها تطبيقات شتى في معظم الأنظمة الصناعية المؤتمتة. لهذا، سنستعرض في هذا البحث بناء عدد من البرامج الخاصة بالعمليات التتابعية وكيفية عملها ونتائج تجريبها باستخدام الحاكم المبرمج من نوع ELC-2001[1].

أنواع العمليات التتابعية:

سنستعرض في هذا البحث بناء برامج العمليات التتابعية شائعة الإستعمال الموضحة في الشكل (1) وهي:

1- تكرار عملية تتابعية عدداً من المرات. إن هذا يعني أن العضو النهائي في النظام (P010) سينتقل من وضع التوقف إلى وضع التشغيل ومن وضع التشغيل إلى وضع التوقف عدداً معيناً من المرات. مثال ذلك تحريك أسطوانة في الاتجاه الأمامي وتوقفها وإعادتها إلى وضعها السابق عدداً من المرات.

2- تكرار عملية تتابعية عدداً من الدورات، كل دورة منها عبارة عن تكرار لعملية التشغيل والإيقاف عدداً من المرات، (P011).

3- تكرار تتابع ما عدداً من المرات يتبعه توقف لمدة معينة، ثم تكرار التتابع عدداً آخر من المرات يتبعه توقف المدة نفسها، ثم تكرار التتابع عدداً آخر من المرات يتبعه توقف للمدة السابقة نفسها، ثم تكرار التتابعات جميعها عدداً من المرات ثم توقف البرنامج، (P012).

4- تكرار عملية ما عدداً من المرات ثم توقف التتابع لمدة معينة يبدأ بعدها تتابع آخر لعدد آخر من المرات ثم يتوقف التتابع الثاني وهكذا حتى ينتهي عمل التتابع النوني (P013,P014,P015)، ثم يتوقف البرنامج. ويبين الشكل (1) مخططاً يمثل هذه الأنواع.

سننظر في البرنامج رقم (1) المبين في الشكل (2)، الذي يحقق النوع الأول من العمليات التتابعية. يتم التشغيل بالضغط على الكبسة الخارجية بتشغيل الملامس الساعاتي المتماثل (F096)، الذي زمن دورته يساوي 20 ثانية. إن هذا يؤدي إلى وضع المخرج (P010) في وضع التشغيل ON لمدة 10 ثوان وفي وضع التوقف (OFF) لمدة 10 ثوان أخرى، وتتكرر هذه العملية دورياً وبصورة مستمرة. يمكن استخدام هذا البرنامج في العمليات الصناعية التتابعية المتعلقة بتحويل مسارات نقل المشغولات أو في عمليات دفع القطع المرتبة بعضها فوق بعض في مستودع بحيث يتم دفع القطع من أسفل المستودع قطعة تلو الأخرى. في الحالة الأخيرة تهبط القطع من الأعلى إلى الأسفل بفعل الجاذبية ويتم دفعها أثناء الشوط الأمامي لاسطوانة قدرة تتحرك ترددياً إلى الأمام وإلى الخلف عدداً محدداً من الدورات. كما يمكن استخدام البرنامج في عملية المشغولات بزاوية ما أو

في عملية نقل المشغولات بزاوية ما باستخدام اسطوانة قدرة تعمل أيضاً بصورة دورية العدد المطلوب من الدورات.

من أجل عدّ المرات التي سيوضع فيها المخرج في وضع التشغيل يستخدم العداد التصاعدي CTU-C004، حيث (C004) هو عنوان العداد. تم تحديد عدد المرات بتسجيل الرقم (6) في مسجل العداد على سبيل المثال لا الحصر. عندما يصل العدُّ الصاعدُ إلى الرقم (6) يتم تشغيل العلم الداخلي (M001). وبسبب وجود ملامسة المغلق في الخطورة رقم (0005) يتم فتح هذا الملامس وإيقاف المخرج (P010) نهائياً. ومن أجل تحديد لحظة وضع المخرج (P010) في وضع التشغيل، وهو الحدث في هذا البرنامج، فإنه من الضروري تحويل الإشارة الداخلة المشغلة للمخرج (P010) إلى نبضة تؤثر على مدخل العداد. عندئذٍ في كل مرة يوضع فيها (P010) في وضع التشغيل تؤثر نبضة على مدخل العداد ويضاف واحد إلى مضمون مسجل العداد (C004). لهذه الغاية تستخدم دالة تفاضل الحافة الصاعدة (Leading Edge Differentiation) وهي الدالة (D017). فكما يبين البرنامج، عندما يوضع (P010) في وضع التشغيل تصدر نبضة إلى العلم الداخلي (M003)، وتظهر هذه النبضة على مدخل العدّ للعداد (الخطوة رقم 0010)، أما الخطوة رقم (0011) فتعمل على تصفير العداد وقت اللزوم.

ومن أجل رصد عدد مرات التشغيل بدون استخدام المخرج (P010)، يمكن استخدام الإشارة الدورية الصادرة عن الملامس الساعاتي المتماثل (F096) مباشرة. من أجل ذلك تستبدل الخطوة (7) بالخطوة:

0007 LOAD F096

ويضاف إلى البرنامج الخطوة:

0008 AND NOT M001

في هذه الحالة لا يعمل الخرج (P010) إلاّ إذا ضغطنا على الكبسة (P000)، ولكن عندما يصل العدد في (C004) إلى القيمة المحددة يتم تشغيل (M001) وفتح ملامسة المغلق الموجود في الخطوة (0008) السابقة، فتثبت قراءة العداد.

ومن أجل تحقيق النوع الثاني من العمليات التتابعية نقترح استخدام معداد (Scaler) يتم بوساطته قسمة عدد من النبضات على عدد معين. إن هذا يعني أننا سنحصل على تغير لوضع المخرج لمرة واحدة من وضع التوقف إلى وضع التشغيل بعد انقضاء عدد من الدورات لإشارة دورية مربعة. وميزة هذه الطريقة في أنها تُمكِّن من إجراء عملية القسمة باستخدام الكيان المرن دون أي تغيير في الكيان المادي كما هو الحال عند استعمال العدادات[2،3].

من أجل ذلك سنستخدم موقتاً أحادي الاستقرار متضمناً في الحاكم المبرمج وهو (TMON-T008) حيث (T008) هو عنوان الموقت. في هذه الحالة يتم وضع الموقت في وضع التشغيل بمجرد إغلاق الملامس الذي يسبقه أو بمجرد انتقال الموجة المربعة الدورية من وضع التوقف إلى وضع التشغيل، بعد هذه اللحظة لا يتأثر وضع الموقت بسبب تغير وضع الملامس الذي يسبقه أو بسبب تغير وضع الموجة المربعة. وفي لحظة انتقال الموقت إلى وضع التشغيل يبدأ العدُّ التنازلي لثابت الزمن المدون في مسجله، وعندما يصل العد إلى الصفر يصبح الموقت في وضع التوقف ويستمر في هذا الوضع إلى أن يغلق الملامس الذي أمامه أو إلى أن تتحول الموجة المربعة من وضع التوقف إلى وضع التشغيل كما هو مبين في الشكل (3). ولأن عمل هذا الموقت يتوقف على مقدار ثابت الزمن المسجل بداخله فإنه يمكن اختيار مقدار ثابت الزمن بحيث نضمن تحول الموقت من وضع التوقف إلى وضع التشغيل مرة واحدة لكل دورة من دورات الموجة المربعة. من أجل ذلك يفترض أن يكون ثابت الزمن للموقت أقل من الزمن الدوري للموجة المربعة. أما إذا رغبنا في أن يتحول الموقت من وضع التوقف إلى وضع التشغيل مرة واحدة لكل دورتين من دورات الموجة المربعة فيفترض أن يكون ثابتُ الزمن المسجل فيه أكبر من الزمن الدوري للموجة المربعة وأقل من ضِعفه. وعليه، فإن أردنا القسمة على خمسة، أي إذا كان المطلوب نقل الموقت من وضع التوقف إلى وضع التشغيل مرة واحدة لكل خمس دورات من دورات الموجة المربعة فإنه يفترض أن يكون ثابت الزمن أعلى من أربعة أضعاف الزمن الدوري وأقل من خمسة أضعافه. بعد ذلك وكما أوضحنا في بداية البحث فإنه يمكن كشف لحظة تحول الموقت من وضع التوقف إلى وضع التشغيل واستخراج نبضة تؤثر على مدخل العداد التصاعدي، حيث سيكون عدد النبضات المسجلُ في مسجل معبراً عن خارج القسمة.

يبدأ تشغيل البرنامج رقم (2) بالضغط على الكبسة (P000) فيؤدي ذلك إلى تشغيل ملامس ساعاتي غير متماثل تم تصميمه باستخدام موقت ذي تأخير زمني عند التشغيل (TON-T195) وموقت آخر أحادي وضع الاستقرار قابل لإعادة القدح (TRTG-T006). نتيجة وضع (M000) في وضع التشغيل وبعد انقضاء 5 ثوان يعمل العلم (M001) فيصبح (TRTG-T006) في وضع التشغيل ويفتح الملامس الموجود في الخطوة (0004) ويبدأ العد التنازلي للرقم المسجل بداخله حتى تنقضي مدة 10 ثوان. في هذه اللحظة يصبح مخرجه في وضع التوقف فينغلق الملامس الموجود في الخطوة (0004)، وبسبب ذلك يصبح (TON-T195) في وضع التشغيل وتتكرر الدورة ثانية وهكذا.

نلاحظ في هذا البرنامج أن زمن التشغيل يعتمد على الرقم المسجل في الموقت (TRTG-T006) ومقداره (100) أن 10 ثوان بينما يعتمد زمن التوقف على الرقم المسجل في المرجل (TON-T195) ومقداره (50) أي 5 ثوان. نتيجة لعمل الملامس الساعاتي يعمل المخرج (P011) في نمط دوري ويكون مقدار الزمن الدوري له 15 ثانية. أما الموقت (TMON-T008) فقد استخدم في البرنامج من أجل تحقيق عملية القسمة المطلوبة، وذلك عن طريق اختيار قيمة ثابت الزمن وهو الرقم المسجل في الخطوة (0019)، وأما باقي خطوات البرنامج فقد تم توضيح دورها سابقاً.

في هذا البرنامج، إذا ما رغبنا في إيقاف المخرج (P011) بعد أن يعمل ست دورات كل دورة منها تتكون من خمسة تتابعات نسجل رقماً في مسجل (TMON-T008) بحيث يكون ثابت الزمن فيه أكبر من 75 ثانية وأقل من 90 ثانية. وإذا ما لزم تكرار هذا التتابع ست مرات فإننا نسجل رقم (6) في الخطوة (0027)، عندئذٍ نضمن تكرار التتابع ست دورات كل دورة منها عبارة عن خمسة تتابعات لوضع (P011)، ثم يتوقف البرنامج بعد ذلك.

يستخدم مثل هذا البرنامج في العمليات التتابعية التي تتضمن دفعات إنتاجية (Batches) محددة. فمثلاً يستخدم هذا البرنامج في جميع الحالات التي يستخدم فيها البرنامج الأول عندما يتطلب الأمر إنجاز عدد معين من تلك التتابعات دفعة واحدة ثم يعقب ذلك توقف لمدة معينة يتم تحديد طولها حسب متطلبات العملية الصناعية ثم يعود البرنامج للعمل لإنجاز الدفعة التالية، وهكذا.

في كثير من العمليات الصناعية، إضافة إلى ضرورة معرفة عدد مرات حدوث حدث معين، يتطلب الأمر تنفيذ عمل آخر بناء على عدد مرات حدوث ذلك الحدث. فمثلاً، عندما يصل عدد العُلَب المنقولة على ناقل متحرك إلى قيمة معينة يفترض إيقاف الناقل.

وقد يتطلب الأمر كما هو الحال في بعض أنظمة الحماية والتحذير، اتخاذ إجراء معين إذا تكرر الحدث لمرتين أو أكثر.

في البرنامج رقم (3) يتم حساب عدد دورات موجة مربعة وقسمته على (2) وعلى (4) وتثبت النتائج في العدادات (C004,C005,C006). وهذا يعني أنه باستخدام البرنامج المشار إليه سيكون ممكناً اتخاذ إجراء معين عندما يحدث تتابع لحدث ما مرتين أو لأربع مرات. في هذا البرنامج يتم تحقيق الموجة المربعة باستخدام مولد النبضات (F103) الذي يبرمج عن طريق الدالة (FUN-205). عند تشغيل البرنامج يبدأ العد في العدادات (C004,C005,C006)، بنفس الوقت، ويسجل في كل منها الرقم (1). تتم إضافة رقم (1) إلى مضمون (C004) بعد كل دورة من دورات (F103)، بينما تتم إضافة رقم (1) إلى (C005) بعد كل دورتين من دورات (F103)، وتتم إضافة رقم (1) إلى (C006) بعد كل أربع دورات للمولد (F103). وبهذا تكون نسبة عدد النبضات المسجلة في (C006,C005,C004) هي 1:2:4. من المهم ملاحظة أن تحقيق هذه النسب قد تم بناء على اختيار قيم ثابت الزمن في كلٍ من (TMON-T008) ثم (TMON-T009). يفترض أن يكون ثابت الزمن من (T008) أقل من مقدار الزمن الدوري لدورة (F103) حتى يتم استخراج نبضة جديدة مع بداية الدورة الثانية. ولأن زمن المسحة (SCAN) يساوي 4.8ms فإن قيمة الزمن الدوري لدورة (F103) تساوي: 2*4.8*100=960ms. لهذا تم اختيار الرقم (6) في الخطوة (0019) الذي يعادل 6*100=600ms. أي تتحقق المتباينة 600ms<960ms. وبنفس الأسلوب تم اختيار الرقمين (16)، (38) في الخطوتين (00.34),(0058).

يعمل البرنامج رقم (4) على تحقيق تتابع خمس مرات متتالية يتبعها فترة زمنية مقدارها 10 ثوان ثم تحقيق تتابع أربعة مرات متتالية يتبعها فترة زمنية مقدارها 10 ثوان ثم تحقيق تتابع مقداره سبع مرات يتبعه فترة زمنية مقدارها 20 ثانية وبعد ذلك تتكرر جميع الخطوات السابقة ثلاث مرات حيث يتوقف بعدها البرنامج.

العداد CTU-C004 وقد سجل فيه العدد (5)

العداد CTU-C005 وقد سجل فيه العدد (5)

العداد CTU-C007 وقد سجل فيه العدد (7)

العداد CTU-C008 وقد سجل فيه العدد (3)

عندما يصل العد في العداد CTU-C004 إلى خمسة يُفَعَّلُ مخرجهُ فيعمل (M002)، وبعد انقضاء 10 ثوان على تشغيل (T192) يعود (P012) إلى العمل ويبدأ العد في العداد C005، ثم بعد وصول العدِّ بداخله إلى (4) يُفَعَّلُ مخرجه (M004) ويبدأ الموقت (T193) بالعدِّ حتى تنقضي 10 ثوان، في هذه اللحظة يبدأ العد في العداد (C007). وبنفس الأسلوب عندما يصل العدِّ بداخل (C007) إلى مقدار (7) يتم تسجيل رقم (1) في العداد (C008) ويبدأ العداد (T194) بالعدِّ لمدةٍ تساوي 20 ثانية وبعدها مباشرة يبدأ العد من جديد في العداد C004. ففي اللحظة التي يصل فيها العد إلى (7) بداخل العداد (C004) ويبدأ العد في (C004) قد وصل إلى (16)، وفي هذه اللحظة يتم تصفير العداد (C004) ويبدأ العد في التزايد من واحد وحتى خمسة. وفي لحظة بلوغ العدد (5) يكون العدد داخل العداد (C005) قد وصل إلى (16=5+7+4)، وعندها يتم تصفير العداد (C005) ويبدأ العد في التزايد من (1) وحتى (4) وهكذا.

وفي نهاية كل دورة عدٍّ داخلية يتم تسجيل رقم جديد بداخل (C008) إلى أن يصل العدد إلى القيمة المثبتة فيه، حيث سيعمل عندئذٍ مخرجه (M001) ويتوقف (P012) عن العمل.

ويستخدم هذا البرنامج عندما تتضمن العملية التتابعية دفعات إنتاجية مركبة. في هذه الحالة تتضمن الدفة المركبة الواحدة عدداً من الدفعات البسيطة. فمثلاً يمكن استخدام البرنامج هذا لإنجاز عملية تثقيب لمجموعة مختلفة من الثقوب أو لتنفيذ عملية برشمة في عدة مواقع على سطح معين، بحيث يتم تثقيب كل مجموعة من الثقوب بعد انقضاء زمن محدد على تثقيب المجموعة السابقة وهكذا، بعد تثقيب جميع المجموعات تتوقف عملية التثقيب لفترة محددة ثم تعاد بعد ذلك جميع العمليات السابقة بنفس الأسلوب.

أما بالنسبة للبرنامج رقم (5)، فبعد الضغط على الكبسة (P000) يبدأ عمل التتابع (P013). تم اختيار ثابت الزمن في الخطوة (0020) بحيث تتولد نبضة على مدخل (C004) لكل دورة من دورات (P013). عندما يصل عدد الدورات إلى خمسة يتوقف التتابع ويبدأ عمل الموقت (TON-T197)، وبعد انقضاء تأخير زمني مقداره 20 ثانية يبدأ عمل التتابع الثاني (P014)، ويعمل أيضاً بحيث تتم إضافة واحد إلى مضمون (C005) لكل دورة من دورات (P014)، وبعد بلوغ العدد في (C005) إلى سبعة يتم إيقاف التتابع وتصفير العدادين C005,C004. استخدمت في البرنامج تعليمة (JMP-00) وتعليمة (JME-00) من أجل إنهاء التتابع الأول أولاً والبدء في التتابع الثاني بعد ذلك. بعد انتهاء التتابع الثاني وانقضاء عشرين ثانية يبدأ التتابع الثالث (P015)، وعندما يصل عدد التتابعات للمخرج (P015) إلى أربعة، يتوقف التتابع ويتم تصفير العداد (C006) ويتوقف البرنامج.

يمكن بهذا البرنامج تشغيل عدد من الأجهزة التنفيذية النهائية المستخدمة في العمليات الصناعية بحيث ينفذ أي جهاز منها العدد المطلوب من التتابعات وبحيث يتم عمل هذه الأجهزة حسب أولويات محددة سلفاً. فمثلاً يمكن استخدام البرنامج في عملية ثني مشغولة معينة حيث تعمل الاسطوانة الأولى على قمط المشغولة وتثبيتها، بينما تعمل بعد ذلك اسطوانة ثانية على دفع طرف المشغولة لثنيها من جهة معينة، بينما تعمل اسطوانة ثالثة على دفع طرف المشغولة لثنيها من جهة أخرى. وينطبق هذا الأمر على عملية ختم المشغولات أيضاً، حيث تعمل الاسطوانة الأولى على تثبيت المشغولة بينما تعمل الاسطوانة الثانية على ختمها وتعمل الاسطوانة الثالثة على دفع المشغولة باتجاه ناقل متحرك، ثم تتكرر هذه العمليات بنفس الترتيب وحسب

طبيعة العملية الصناعية يمكن تنفيذ أية عملية أي عدد من المرات حسب الطلب.

ملاحظات حول البحث:

- إن استخدام لغة "C" لبرمجة الحاكمات المبرمجة ممكن من الناحية المبدئية ولكنه لا ينطبق على مجالات تطبيق البرامج المقترحة التي نرى أن تتم كتابتها باستعمال لغة منخفضة المستوى مثل المنطق السلمي أو قائمة التعليمات وأن يتم إدخالها إلى الحاكم المبرمج مباشرة عن طريق مبرمج يدوي مرفق مع الحاكم نفسه.
- إن استخدام لغة "C++" شأنها شأن أية لغة عالية المستوى، من أجل برمجة الحاكمات المبرمجة أمر ممكن من حيث المبدأ، ولكن من المهم التنويه إلى أن التحكم في العمليات التتابعية يتم في الصناعة عادة باستخدام الحاكمات المبرمجة وليس باستخدام الحاسوب. ويعود ذلك للأسباب الرئيسية التي أدت إلى استخدام الحاكمات المبرمجة بدلاً من الحواسيب في أتمتة العمليات الصناعية الإنتاجية. فتم تصميم الحاكمات المبرمجة لتعمل في البنية الصناعية القاسية، بحيث يمكن برمجتها بدون وجود مهارات برمجية خاصة، وبحيث يمكن تشخيصها وتحديد الأعطال فيها من قبل الفنيين والمهندسين في الموقع وبحيث يمكن تركيبها على الآلة وكأنها جزء من تلك الآلة.

ونرغب في هذا المجال أن نشير إلى أن الغالبية العظمى من الحاكمات المبرمجة في الوقت الحاضر تبرمج باستعمال لغات رمزية (Symbolic Languages)، تشبه المخططات السلميّة المستخدمة في الحاكمات المرحلية المنطقية (Relay Logic Controllers) وهاتان اللغتان هما:

1) التذكرة الرمزية (Mnemonic Code) وتسمى أيضاً بقائمة التعليمات (Instruction List) وهي اللغة المنطقية التي ورد بها البحث.
2) لغة المخطط السلّمي وتسمى أيضاً بالمنطق السلّمي (Ladder Logic) وكلا الأسلوبين قابل للتحويل بعضهما لبعض.

لقد بحثنا في هذا الأمر وسألنا لدى وكلاء الشركات المعنية فتوصلنا إلى ما يلي:

1) إن أكثر من 80% من الحاكمات المبرمجة في الولايات المتحدة تتم برمجتها باستعمال اللغات الرمزية المشار إليها أعلاه (صفحة رقم 185[4]).
2) تتم المعالجة باستعمال اللغات الرمزية على التوازي (Parallel Processing) وهذا يعطيها ميزة سرعة الأداء، علماً بأنه يمكنها أيضاً أن تصبح لغات معالجة على التوالي باستعمال ملامس مزلاجي (Latching Contact). ولكن هذا لا ينطبق على جميع اللغات عالية المستوى، فعلى سبيل المثال إن لغة بيسك مازالت لغة مناسبة للمعالجة على التوالي ولهذا فهي بطيئة نسبياً. وبالنسبة للغة (C++) فإنه يمكن من خلالها إجراء المعالجة على التوالي وعلى التوازي، ولكن العقبة الأساسية في استخدامها لبرمجة الحاكمات المبرمجة هي ضرورة وجود آلة حاسبة خاصة بهذا الأمر. إضافة إلى ذلك فهناك حاجة إلى دارة مؤلف (Compiler/Converter) يعمل على ترجمة البرنامج من لغة (C++) إلى الكودة التشغيلية الخاصة بالحاكم المبرمج. ومثل هذا المؤلف غير موجود في الوقت الحاضر، أضف إلى ذلك أن اختلاف الكودات التشغيلية في الحاكمات سيتطلب تصميم دارات مؤلفات مختلفة فيزداد الموقف تعقيداً.

ومع ذلك، فإن هذا الأمر يستحق الاهتمام في حالة استعمال الحاسوب كسيّد (Master) يدير شبكة من نوع واحد من الحاكمات المبرمجة في عملية صناعية معقدة تستحق تصميم مؤلف (عاكس) يضمن التقابل بين لغة (C++) وبين الكودة التشغيلية لجميع الحاكمات المستخدمة، وهذه حالة خاصة، لا ينطبق عليها وضع البرامج التي تم استقصاؤها في البحث.

كما تجدر الإشارة إلى أن ثمن المؤلف سيكون غالياً ويزيد على ثمن الحاكم المبرمج نفسه، وأن استخدام شبكة من الحاكمات المبرمجة التي تقودها آلة حاسبة واحدة مبرر في الغالب عندما توجد جميع الحاكمات في ورشة معينة أو في ورش متقاربة. لقد كانت مشكلة فعلية مثلاً، عملية إعادة برمجة مجموعة من الحاكمات المبرمجة بوساطة الحاسوب عندما كانت هذه المجموعة منتشرة على مساحة واسعة في المدينة من أجل التحكم في إشارات المرور، إذ كان يفترض إحضار الحاسوب إلى

كل تقاطع من التقاطعات لهذه الغاية بينما كان من الأسهل فحص البرامج وتعديلها بوساطة الطرف الخاص بإدخال البرنامج المرفق مع الحاكم المبرمج نفسه.

نرغب في النهاية أن نضع وجهة نظر المؤلف (Gray N. Johnson) حول استعمال اللغات عالية المستوى، في كتابه بعنوان: Labview – Graphical Programming الذي صدر عن Mc-Graw-Hill عام 1994، يقول في الصفحة رقم (7):

In the beginning, there was only machine language, and all was darkness assembly language was invented, and there was a glimmer of light in the programming world, then came Fortran, and the light went out.

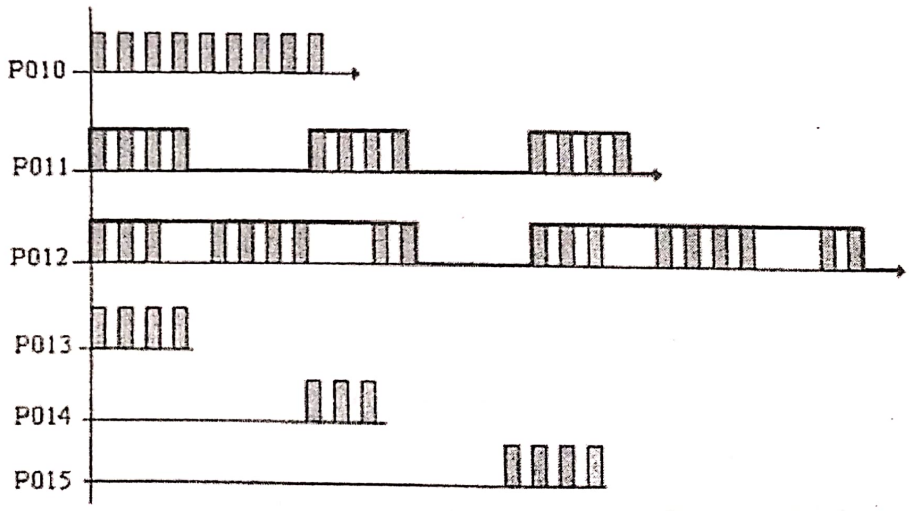
الاستنتاجات

1- آخذين في الاعتبار التطورات المذهلة على الكيانين المادي والمرن للحاكمات المبرمجة وشيوع مجالات تطبيق العمليات التتابعية الدورية فإن استخدام الحاكمات المبرمجة لتحقيق هذه العمليات هو الخيار الأفضل في الوقت الحاضر. ومن أجل تحقيق عمليات دورية استخدمت الملامسات الساعاتية المتماثلة الموجودة في الحاكم المبرمج وهي (F094,F095,F096)، كما استخدمت مولدات الموجات النبضية المربعة عن طريق الدالة (FUN-205) كما اقترح الباحثان برنامجاً فرعياً لتوليد إشارة دورية نبضية في البرنامج رقم (2) باستخدام الموقتين (TON-T195) و(TRTG-T006).

2- تم بناء خمسة من البرامج تحقق خمسة أنواع شائعة من العمليات الصناعية التتابعية وتم تجريبها وتوضيح تكوينها ومبدأ عملها.

3- من أجل استخلاص لحظة تشغيل العنصر النهائي تمت الاستفادة من دالة تفاضل الحافة الصاعدة، وبذلك أمكن تحويل حالة الانتقال من وضع التوقف إلى وضع التشغيل إلى نبضة قصيرة تؤثر على مدخل العداد التصاعدي.

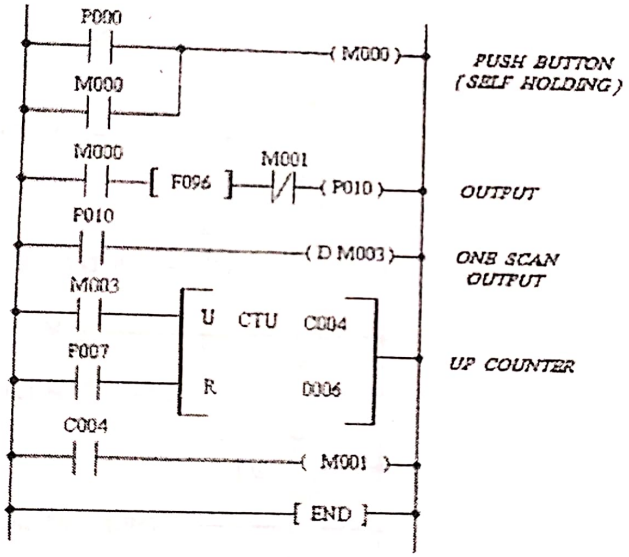
4- ومن أجل قسمة عدد الدورات المعبرة عن انتقال العنصر النهائي من وضع التوقف إلى وضع التشغيل، تمت الاستفادة من موقت أحادي الاستقرار متضمن في الحاكم المبرمج، وتم توضيح اعتمادية خارج القسمة على مقدار ثابت الزمن الخاص بالموقت وكيفية اختياره لضمان إنجاز عملية تتابعية بنمط معين.



شكل (1): مخطط العمليات التتابعية الأربعة مدار البحث

0000	LOAD	P000	*PUSH BUTTON*
0001	OR	M000	
0002	OUT	M000	*Flag*
0003	LOAD	M000	
0004	AND	F096	*Symmetric Contact*
0005	AND NOT	M001	
0006	OUT	P010	*Output*
0007	LOAD	P010	
0008	FUN	017	*One Scan Output*
0009		M003	
0010	LOAN	M003	*Count Input*
0011	LOAD	P007	*Reset Input*
0012	CTU	C004	
0013		6	*Preset Number*
0015	LOAD	C004	
0016	OUT	M001	
0017	FUN	1	*END*

أ- قائمة التعليمات

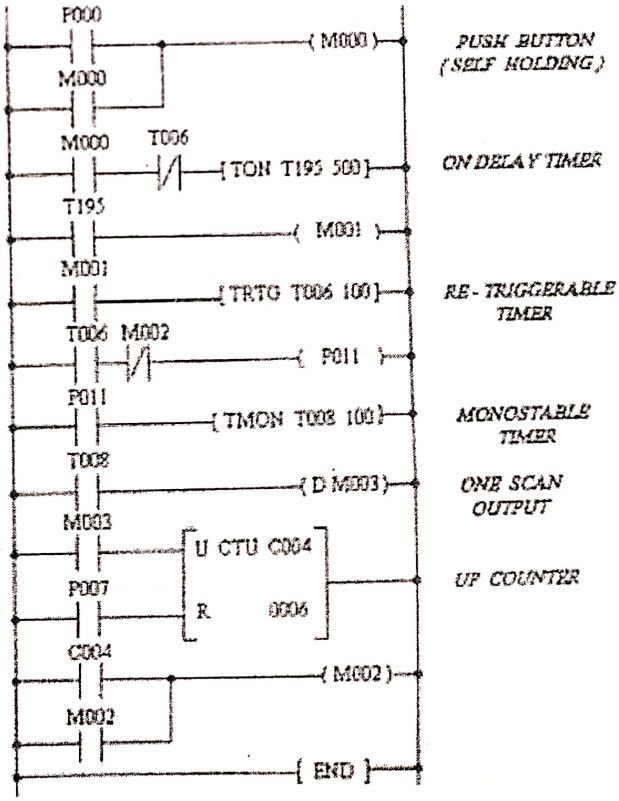


ب- المخطط السلمي

شكل (2): البرنامج رقم (1)

0000	LOAD	P000		0017	LOAD	P001	
0001	OR	M000		0018	TMON	T008	*Monostable timer*
0002	OUT	M000		0019		100	
0003	LOAD	M000		0020	LOAD	T008	
------	-------	------		------	------	------	
0004	AND OUT	T006		0022	FUN	017	
0005	TON	T195		0023		M003	
0006		500	*5 Sec*	0024	LOAD	M003	*Coun Input*
0008	LOAD	T195		0025	LOAD	P007	*Reset Input*
0009	OUT	M001		0026	CTU	C004	*Up Counter*
0010	LOAD	M001		0027		6	*Pre-set Value*
0011	TGTR	T006		0029	LOAD	C004	
0012		100	*10Sec*	0030	OR	M002	
0014	LOAD	T006		0031	OUT	M002	
0015	AND NOT	M002		0032	FUN	1	
0016	OUT	P011					

أ- قائمة التعليمات



المخطط السلمي

شكل (3): البرنامج رقم (2)

0000	LOAD	P000
0001	OR	M000
0002	OUT	M000
0003	LOAD	M000
------	-----	------
FUN		205
0005		F103
0007		100
0009		100
0011	LOAD	F103
0012	OUT	M001
-----	-----	-------
0013	LOAD	M000
0014	AND	M001
0015	AND NOT	M002
0016	OUT	P012
-----	--------	-------
0017	LOAD	F103
0018	TMON	T008
0019		0006
0021	LOAD	T008
-----	--------	----------
0022	FUN	017
0023		M003
0024	LOAD	M003
0025	AND NOT	M002
0026	LOAD	P007
0027	CTU	C004
0028		0040
0030	LOAD	C004
0031	OUT	M002
0032	LOAD	M000
0033	FUN	110
0034		0016

0036		
0038		
------	-------	
0040	LOAD	
0041	TMON	
0042		
0044	LOAD	
------	-------	
0045	FUN	
0046		
0047	LOAD	
0048	AND OUT	
0049	LOAD	
0050	CTU	
0051		
0053	LOAD	
0054	OR	
0055	OUT	
0056	LOAD	
0057	FUN	
0058		
0060		
0062		
------	----------	
0064	LOAD	
0065	TMON	
0066		
0068	LOAD	
-----	------	
0069	FUN	
0070		
0071	LOAD	
0072	AND NOT	
0073	LOAD	

0000	
D0000	

F103	
T009	
D0000	
T009	

017	
M004	
M004	
M005	
P007	
C005	
20	
C005	
M005	
M005	
M000	
110	
38	
0000	
D0001	

F103	
T013	
D000	
T013	

017	
M006	
M006	
M007	
P007	

0074	CTU	C006
0075		10
0077	LOAD	C006
0078	OR	M007
0079	OUT	M007
0080	FUN	1

شكل (4): البرنامج رقم (3)

0000	LOAD	P000	*PUSH BUTTON FLAG*	0027		5	
0001	OR	M000		0029	LOAD	C004	
0002	OUT	M000		0030	AND NOT	T192	
0003	LOAD	M000		0031	OUT	M002	
-------	--------	-------		0032	LOAD	C004	
0004	FUN	205	*DUTY CYCLE GENERTOR*	0033	TON	T192	*ON DELAY TIMER*
0005		F103		0034		1000	
0007		500		0036	LOAD	T192	
0009		500		0037	OR	M005	
0011	LOAD	F103		0038	OUT	M005	
0012	OUT	M001		0039	LOAD	M005	
------	-------	-------		0040	AND	P012	*OUTPUT*
0013	LOAD	M000		0041	FUN	017	
				0042		M006	
0014	AND	F103		0043	LOAD	M006	*COUNT INPUT*
0015	AND NOT	M002		0044	AND NOT	M004	
0016	AND NOT	M004		0045	LOAD	M002	*RESET INPUT*
0017	AND NOT	M009		0046	CUT	C005	
0018	AND NOT	M011		0047		4	
0019	OUT	P012	*OUTPUT*	0049	LOAD	C005	
-------	--------	-------		0050	AND NOT	T193	
0020	LOAD	P012		0051	OUT	M004	
0021	FUN	017	*ONE SCAN OUTPUT*	0052	LOAD	C005	
0022		M003		0053	TON	T193	
0023	LOAD	M003	*COUNT INPUT*	0054		1000	
0024	AND NOT	M002		0056	LOAD	T193	
0025	LOAD	M009	*RESET INPUT*	0057	OR	M007	
0026	CTU	C004	*UP COUNTER*	0058	OUT	M007	

0058 OUT M007 0076 LOAD T194

0059	LOAD	M007		0077	LOAD	M009	
0060	AND	P012		0078	FUN	017	
0061	FUN	017		0079		M010	
0062		M008		0080	LOAD	M010	*Count Input*
0063	LOAD	M008	*Count Input*	0081	LOAD	C008	*Reset Input*
0064	AND	M009		0082	CTU	C008	*Up counter*
	NOT						
0065	LOAD	M004	*Reset Input*	0083		3	
0066	CTU	C007	*Up Counter*	0085	LOAD	C008	
0067		7		0086	OR	M011	
0069	LOAD	C007		0087	OUT	M011	
0070	AND	T194		0088	FUN	1	
	NOT						
0071	OUT	M009					
0072	LOAD	C007					
0073	TON	T194					
0074		2000	*20Sec.*				

شكل (5): البرنامج رقم (4)

0000 LOAD P000 *Push Button* 0033 LOAD M000

0001	OR	M000	*Flag*	0034	AND NOT	M002	
0002	OUT	M000		0035	FUN	012	*JMP 00*
0003	LOAD	M000		0036	LOAD	M002	
0004	FUN	205	*Duty Cycle Generator*	0037	TON	T197	*On Delay Timer*
0005		F103		0038		2000	
0007		100		0040	LOAD	T197	
0009		100		0041	AND	F094	*Clock contact 2sec. Symmetric*
0011	LOAD	F103		0042	AND NOT	M006	
0012	OUT	M001		0043	OUT	P014	*Out put*
0013	LOAD	M000		0044	LOAD	P014	
0014	AND	M001		0045	TMON	T009	*Monostable Timer*
0015	AND NOT	M002		0046		0010	
0016	AND NOT	M006		0048	LOAD	T009	
0017	OUT	P013	*Out put*	0049	FUN	017	*One Scan output*
0018	LOAD	P013		0050		M004	
0019	TMON	T008	*Monostable Timer*	0051	LOAD	M004	*Count input*
0020		0010		0052	AND NOT	M006	
0022	LOAD	T008		0053	LOAD	M006	*Reset input*
0023	Fun	017	*One Scan Output*	0054	CTU	C005	*Up-counter*
0024		M003		0055		0007	
0025	LOAD	M003		0057	LOAD	C005	
0026	AND NOT	M002		0058	OUT	M006	
0027	LOAD	M006	*Reset*	0059	FUN	013	*JMP 00*
0028	CTU	C004	*Up Counter*	0060	LOAD	M000	*Flag*
0029		5		0061	AND NOT	M006	
0031	LOAD	C004		0062	LOAD	M006	
0032	OUT	M002		0063	TON	T198	*On Delay Timer*

0064		2000	*20Sec.DEL*
0066	LOAD	T198	
0067	AND	F095	*Clock Contact 10 Sec.Symmetric*

0068	AND NOT	M007	
0069	OUT	P015	*Out Put*
0070	LOAD	P015	
0071	TMON	T007	*Monostable Timer*
0072		10	
0074	LOAD	T007	
0075	FUN	017	*One Scan Output*
0076		M008	
0077	LOAD	M008	*Count Input*
0078	AND NOT	M007	
0079	LOAD	M007	*Reset Input*
0080	CTU	C006	*Up Counter*
0081		4	
0083	LOAD	C006	
0084	OR	M007	
0085	OUT	M007	
0086	FUN	0001	*END*

شكل (6): البرنامج رقم (5)

Reference المراجع

]- ELC – 2001 - Programmable controller, Hardware manual, arlo Gavazzi, 1996, Denmark.

]- A.J. Bouwens - Digital Instrumentation. McGraw-Hill International Editions, 1986, Singapore.

]- Larry D. Jones – Principles and applications of digital electronics, Macmillan Publishing Company, 1986, New York.

]- John W. Webb; Ronald A. Reis – Programmable Logic Controllers, Principles and applications, 3rd edition, Prentice Hall, 1995, USA.