مجلة جامعة تشرين للبحوث والدراسات العلمية _ سلسلة العلوم الهندسية المجلد (38) العدد (5) 2016
Tishreen University Journal for Research and Scientific Studies - Engineering Sciences Series Vol. (38) No. (5) 2016

دراسة استخدام تقنية "OPC SERVER" في حل مشاكل الاتصال والمواءمة وإزالة عدم التوافق بين التجهيزات المكونة لنظم التحكم المبرمجة

د. ثائر أحمد ابراهيم *
حازم فؤاد زيدان **

(تاريخ الإيداع 2 / 3 / 2016. قُبِل للنشر في 20/ 9 / 2016)

□ ملخّص □

يقدم المقال دراسة استخدام معيار أو تقنية "OPC" بهدف المواءمة وإزالة عدم التوافق بين التجهيزات المختلفة المكوّنة لنظام التحكم المبرمج. حيث تعمل تقنية "OPC" كوسيط يقوم بتأمين الاتصال بين الأجهزة التي تعد مصدراً للبيانات من جهة وبين البرامج التي تحتاج لقراءة هذه البيانات من جهة أخرى، وذلك عندما لا يمكن لهذه البرامج أن تتصل بشكل مباشر مع الأجهزة المصدّرة للبيانات بسبب عدم وجود توافق بينها، وذلك نتيجة اختلاف الشركات المنتجة لكل من الأجهزة والبرمجيات أو قدم الأجهزة المستخدمة وعدم قدرتها على الاتصال مع البرمجيات الحديثة وغيرها. هنا يظهر دور تقنية "OPC" في تقديم الحل المناسب لمشاكل الاتصال بين الأجهزة والبرمجيات، بحيث تقوم بدور الوسيط بينهما عن طريق الاتصال مع كل منهما على حدى ونقل البيانات فيما بينهما بشكل يؤدي إلى عمل منظومة التحكم بشكل ممتاز وكأن الاتصال يتم بشكل مباشر.

سيتم عرض أهمية تقنية "OPC" المقترحة في هذا المقال عملياً من خلال تطبيق هذه التقنية في حل أحد مشاكل منظومة التحكم بالضخ في الشركة السورية لنقل النفط في بانياس "SCOT"، وذلك عن طريق استخدام برنامج (Kepware OPC Server) وبرنامج (WinCC Flexible) للاستعاضة عن برنامج المراقبة والتحكم الذي كان مستخدما مع الإبقاء على نفس تجهيزات "PLC" القديمة، وكنتيجة لذلك سيتم التخلص من المشاكل التي كانت موجودة في نظام المراقبة والتحكم السابق المستخدم من قبل الشركة وبالتالي ضمان عمل النظام بالشكل المطلوب دون أية مشاكل، مما يعود بالنفع الاقتصادي الكبير والموثوقية العالية في العمل.

الكلمات المفتاحية : OPC Server ، SCADA ، WinCC Flexible ، نظام تحكم ، بروتوكول اتصال

* **مدرس – قسم الأتمتة الصناعية – قسم هندسة الأتمتة الصناعية – كلية الهندسة التقنية – جامعة طرطوس – سورية.**
** **طالب دراسات عليا – ماجستير – قسم هندسة الأتمتة الصناعية – كلية الهندسة التقنية – جامعة طرطوس – سورية.**

مجلة جامعة تشرين للبحوث والدراسات العلمية _ سلسلة العلوم الهندسية المجلد (38) العدد (5) 2016
Tishreen University Journal for Research and Scientific Studies - Engineering Sciences Series Vol. (38) No. (5) 2016

Studying The Use of OPC SERVER Technology To Solve Communication Problems And Adaptation And Remove The Incompatibility Between Different Components And Devices In The Programmable Control System

Dr. Their Ibrahim[*]
Hazem Zidan[**]

(Received 2 / 3 / 2016. Accepted 20 / 9 / 2016)

□ ABSTRACT □

In this essay we will study the using of OPC standard in adapting and removing the incompatibility between different components and devices in the control system . The OPC works as a third party that connect between devices (data sources) and programs that require this data , in case those programs may can't connect directly with that devices as a result of incompatibility between them. This incompatibility happens because these devices and programs are from different vendors or because the devices are very old and can't connect with newer programs. OPC will be a suitable solution for communications problems between the devices and programs in the control system, the OPC will connect to each of them and transfer data as good as if the connection between them is direct without any third party.

The importance of this suggested technique will be shown practically in this essay through the implementation of it in solving one of the pumping control system problems in the Syrian Company for Oil Transport (SCOT). the suggestion was to use a OPC Server program (Kepware OPC Server) and a SCADA program (WinCC flexible) instead of the old SCADA program and keep the old PLC devices , in this way we can get rid of the problems that appear in the old SCADA program and therefore we can ensure that the system will work as wanted with no problem. That will bring benefits to the company and results to high reliability in the work.

Keywords: OPC Server, SCADA, WinCC Flexible, Control System, Communication Protocol

[*] **Assistant Professor, Department Of Industrial Automation - Faculty of Technical Engineering - Tartous University- Syria.**
[**] **Postgraduate Student – Master - Faculty of Technical Engineering - Tartous University - Syria.**

مقدمة:

تطورت الشبكات الصناعية المستخدمة في أنظمة التحكم تطورا كبيرا خلال العقود الماضية، وظهرت العديد من أنواع الشبكات المختلفة، بدءاً من الشبكات البسيطة التي تربط بين نقطتين وصولا إلى الشبكات التي تصل بين عدد هائل من النقاط المنتشرة في جميع أنحاء العالم. هذا التنوع والاختلاف في أنواع الشبكات يعود إلى البروتوكولات العديدة المستخدمة، والتي قامت بتطويرها شركات مختلفة كلٌ بما يحقق إمكانية الاتصال مع أجهزة معينة بحد ذاتها، قد تكون هذه الأجهزة تابعة لنفس الشركات المطورة لبروتوكولات الاتصال أو مرتبطة معها باتفاقيات تجارية وتقنية. وهذا التشعب في البروتوكولات والأجهزة والشركات المصنّعة أدى مع مرور الزمن إلى ظهور ثغرات بين الأجهزة القديمة المستخدمة وبروتوكولات الاتصال الحديثة التي تظهر تباعاً وذلك بسبب عدم التوافق فيما بينها، مما أوجب الحاجة إلى وجود وسيط يؤمن الاتصال بالأجهزة القديمة دون الحاجة إلى تغييرها، ولأجل ذلك تم الوصول إلى بروتوكول معياري عالمي متفق عليه من قبل العديد من الشركات الصناعية العالمية الكبيرة المختصة وتم تسميته "OPC" اختصاراً لجملة (OLE for Process Control) ، حيث "OLE" هي اختصار (Object Linking and Embedding) وهي بيئة عمل في ويندوز تستخدم لربط الكائنات وتضمينها في النوافذ والتطبيقات [1] ، وبذلك تم إيجاد حلول لكثير من المشاكل المتعلقة بعملية الاتصال في نُظم التحكم.

أهمية البحث وأهدافه:

يهدف البحث إلى تطبيق تقنية "OPC" المستخدمة على نطاق واسع في عمليات التحكم للتغلب على مشاكل الاتصال في الأنظمة ذات المكونات المتنوعة والمختلفة، حيث سيتم استخدام هذه التقنية في حل مشكلة الاتصال لمنظومة تحكم مبرمج "PLC" قديمة مستخدمة في الشركة السورية لنقل النفط، ودمجها مع أحد برامج المراقبة "SCADA" الحديثة للحصول على نظام مراقبة وتحكم جديد بدلاً من النظام القديم دون تغيير كبير في البنية التحتية القديمة ذات التكلفة المادية المرتفعة، وبذلك يمكن توفير مقدار كبير من الجهد والوقت والمال الذي قد تتطلبه إعادة بناء هذه المنظومة المخصصة للتحكم في عملية الضخ العكسي للمشتقات النفطية في مقر الشركة في بانياس.

طرائق البحث ومواده:

تم في البداية شرح تقنية "OPC" وخصائصها ومواصفاتها وأشكالها والمميزات التي تقدمها لمنظومات التحكم، ثم تم التعريف بأحد بروتوكولات الاتصال المستخدمة في الشبكات الصناعية وهو (Modbus) والأنماط الرئيسة لهذا البروتوكول وطرق عنونة البيانات فيه ثم عملنا على الوصل بينه وبين أحد برامج تقنية "OPC" وتطبيق ذلك في حل مشكلة الاتصال في نظام التحكم بالضخ في الشركة السورية لنقل النفط في بانياس. حيث تم تطبيق هذا العمل وتجريبه في الشركة السورية لنقل النفط نهاية عام 2015

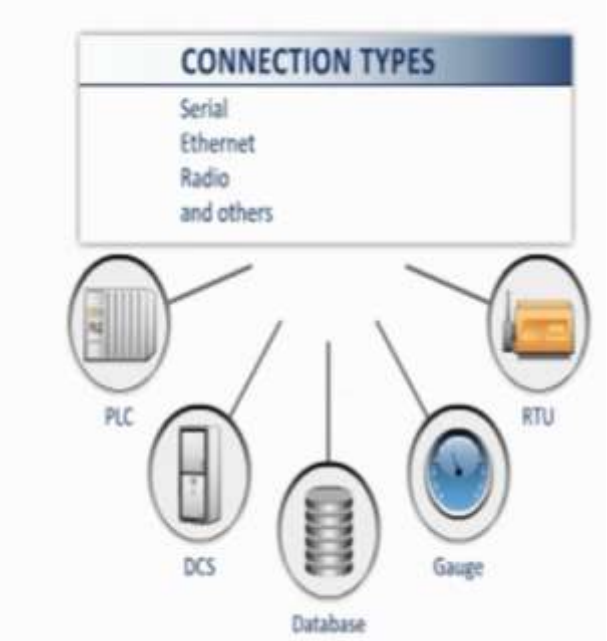
1- تقنية "OPC" :

تقنية "OPC" هي طريقة معيارية تسمح للتطبيقات الصناعية المتعددة أن تشارك البيانات بأسلوب سريع وقوي متلافيةً مشاكل عدم التوافق بين مكونات منظومة التحكم. [2]

وهذا يعني أن تقنية "OPC" تؤمن نقل البيانات بسرعة وضمن الوقت (الزمن) الحقيقي أو يكون التأخير في عملية النقل يكاد لا يُذكر كما أن ذلك يتم بجودة عالية بدون وجود أية ضياعات قد تؤثر على جودة عملية الاتصال.

ويمكن تعريف هذه التقنية بأنها موائم بيني (Interface) مستقل عن نوعية الجهاز المتصل به وبالتالي فإن المستخدم غير مقيد بنوعية الأجهزة والبرمجيات التي يحتاجها في عمله وما يحتاجه فقط هو التكامل بينها عبر برنامج (OPC) مناسب. [3]

الهدف من تقنية "OPC" هو تقديم بنية تحتية معيارية لتبادل بيانات عمليات التحكم، فالشركات المنتجة لمكونات عملية التحكم تقوم بإنتاج العديد من مصادر البيانات (PLC's – DCS's – RTU's – قواعد بيانات – عدادات وأجهزة أخرى)، وهذه البيانات تكون متاحة من خلال عمليات اتصال مختلفة (Ethernet – Serial – Radio Transmissions وغيرها) ويكون ذلك عبر أنظمة تشغيل مختلفة مثل (Windows – Unix – DOS – VMS) ليتم في النهاية استخدام هذه البيانات من قبل تطبيقات تحكم مختلفة[4] ، وهذا موضح في الشكل(1):



الشكل (1) : بعض أنواع مصادر البيانات في أنظمة التحكم وبعض أشكال الاتصالات اللازمة لنقل هذه البيانات.

في الماضي كانت مصادر البيانات والأجهزة كلها من نفس الشركة الصانعة في أغلب الأحيان أو من منتجين متوافقين، وبعبارة أخرى كان يمكن الوصول إلى البيانات فقط باستخدام أدوات من الشركات التي تقوم بتزويد هذه البيانات، وبالتالي لا بد من العودة لهذه الشركات عند الحاجة إلى توسيع المشاريع والعمليات أو إضافة أجهزة جديدة إليها وهذا بالتالي يسبب المزيد من الكلفة والتبعية لشركات التصنيع. مما أدى لوجود الحاجة لوسيط يقوم بنقل البيانات بين المنتجات من شركات مختلفة، فكانت تقنية "OPC" هي هذا الوسيط الذي سيمكّن من نقل البيانات من أي مصدر لهذه البيانات إلى أي تطبيق يحتاجها، وذلك عند تأمين الاتصال بين "OPC" وطرفي عملية تبادل البيانات. وأصبح لتقنية OPC معايير عالمية يتم وضع مواصفاتها من قبل مؤسسة (OPC Foundation) والتي تقوم باستمرار بإصدار مواصفات جديدة لهذا المعيار [4].

تقوم تقنية "OPC" باستخدام تقنيتي (com)[1] و (Dcom)[2] من شركة مايكروسوفت لتتيح للتطبيقات أن تتبادل البيانات على حاسب واحد أو عدة حواسيب باستخدام بنية Client–Server (زبون – مُخدّم). [4]

الزبون يسمح للمستخدم بإجراء طلب قراءة أو كتابة من المُخدّم والذي يقوم بتحويل هذا الطلب إلى الجهاز الذي يتم طلب البيانات منه، وتتم عملية التحويل هذه عن طريق ترجمة الطلب إلى البروتوكول الذي يستخدمه جهاز مصدر البيانات، يمكن تشبيه الزبون بشخص يدخل إلى المطعم ثم يقوم باختيار طعام ما ويطلبه من النادل الذي يكون بمثابة المُخدّم، ويقوم النادل بإيصال هذا الطلب إلى الطباخ في المطبخ الذي يقوم بتحضيرها ثم إعطائها للنادل الذي بدوره يجلب الوجبة إلى الزبون. [7]

2- مواصفات تقنية OPC (OPC Specifications):

يوجد العديد من المواصفات لتقنية "OPC" ، سوف نذكر أهمها:

1) OPC Data Access (OPC DA): وهي تقنية تساعد في الوصول إلى البيانات ضمن الوقت(الزمن) الحقيقي وباستخدامها يمكن الحصول على القيم اللحظية للمتغيرات المختلفة.

2) OPC Historical Data Access (OPC HDA): ويستخدم لاستعادة وتحليل البيانات المخزنة مسبقا.

3) OPC Alarms & Events (OPC A & E): وتستخدم لتبادل بيانات الإنذارات والأحداث في عملية التحكم.

4) OPC Data Exchange (OPC DX): يقوم بتحديد كيفية تبادل البيانات بين عدة مُخدّمات OPC (OPC Servers).

5) OPC XML (Extensible Markup Language): يقوم بتغليف (Encapsulate) بيانات عملية التحكم وجعلها متاحة لكل أنظمة التشغيل .

وكما ذكرنا فإن هذه المواصفات يتم تحديثها بشكل مستمر من قبل مؤسسة (OPC Foundation) والإضافة عليها بشكل دائم. [4]

3- أشكال تقنية "OPC":

تكون تقنية "OPC" في شكلين رئيسيين : مُخدّم (OPC Server) وزبون (OPC Client) . [3]

-(OPC Client) هو برنامج يقرأ البيانات من المُخدّم (OPC Server) من أجل معالجتها بالشكل المطلوب وتعتبر أنظمة "SCADA" و "HMI" أفضل مثال على برنامج الزبون.

-(OPC Server) هو برنامج يقوم بتزويد البيانات إلى الزبون ويقوم بقراءة هذه البيانات وجلبها من جهاز خارجي يعد مصدراً لهذه البيانات مثل "PLC" .

---

(1) Component Object Model : هي معيار اتصال لمكونات البرامج تم تقديمه من قبل شركة Microsoft عام 1993 ويستخدم لتمكين الاتصال والتخاطب بين مجال كبير من لغات البرمجة. [5]

(2) Distributed Component Object Model: هي تقنية مملوكة لشركة Microsoft لتأمين الاتصال بين مكونات البرامج الموزعة على الحواسب الخاصة بشبكة ما وهذه التقنية تعتبر توسيعا لتقنية COM وتوفر ركائز الاتصال للتطبيقات عبر الشبكة. [6]

الاتصال بين الزبون (OPC Client) والمُخدّم (OPC Server) يتم عبر بروتوكولات خاصة يفهمها الطرفان، مما يعني أن أي (OPC Client) يمكنه الاتصال مع أي (OPC Server) بوجود هذه البروتوكولات وبغض النظر عن نوع الأجهزة التي يتصل معها المخدّم (Server) ويقوم بجلب البيانات منها. [3]

4- مبادئ نقل البيانات:

(OPC Server) هو تطبيق يعمل في بيئة ويندوز (MS-Windows) يقوم بالاتصال مع جهاز خارجي مثل المتحكم "PLC" وذلك عبر بروتوكولات محددة خاصة بهذا الجهاز الخارجي. فمثلا يمكن أن يتصل (OPC Server) مع المتحكم "PLC" عبر الوصلة التسلسلية (serial) أو وصلات مُخصصة أخرى.

البيانات التي يستقبلها (OPC Server) من الجهاز يتم تقديمها إلى تطبيقات أخرى تعمل كـ (OPC Client) وتقوم بقراءة هذه البيانات.

معيار "OPC" يبين أن عدة عملاء (Clients) حتى لو كانوا من شركات مختلفة يمكنهم أن يتصلوا مع المُخدّم (OPC Server) بشكل متوافق واستقبال البيانات منه وفهمها حتى لو كان هذا المُخدّم من شركة مختلفة ومعظم أنواع (OPC Client) يمكنها أن تتصل مع أنواع مختلفة من المُخدّمات (OPC Servers) وتتوافق معها كما ذكرنا سابقا. [3]

3-5- تهيئة (OPC Server):

عند تنصيب (OPC Server) يجب أن يتم إدخال الإعدادات اللازمة لتأمين الاتصال مع الأجهزة اللازم التعامل معها بحيث يتم نقل البيانات بينهما بدون وجود أية مشاكل وبشكل متوافق وبالتالي يجب وضع تهيئة معينة له وهذه التهيئة تشمل عادة ما يلي :

- قائمة الأجهزة المراد الاتصال بها ونوعها.
- تحديد البارامترات الخاصة بالاتصال والبروتوكول المستخدم لتأمين هذا الاتصال.
- تحديد قائمة المتغيرات والبارامترات المراد قراءتها من الجهاز.
- إعطاء هذه المتغيرات العنوان المناسب لكل منها بحيث يسهل قراءة هذه المتغيرات عبر العناوين الخاصة بها.

ملاحظة: إن طريقة التهيئة خاصة بكل شركة من الشركات المصنِّعة وما ذكرناه هي خطوط عامة للعمل. [3]

3-6- كيف تقوم تقنية "OPC" بحل مشاكل الاتصال في الأنظمة المؤتمتة:

يكاد لا يوجد مصنع في العالم لا يستخدم الأتمتة في عمله فالأتمتة أصبحت جزءاً لا يتجزأ من أي عملية صناعية. وأي نظام تحكم مُؤتمت يكون مُؤلفاً من العديد من المكونات بدءاً بتجهيزات التحكم وبرامج المراقبة والتحكم وتطبيقات الاتصال وغيرها. توجد الكثير من الشركات حول العالم التي تقوم بتزويد هذه المكونات الخاصة بعملية التحكم، لكن بعض منتجات هذه الشركات تكون غير متوافقة مع منتجات الشركات الأخرى في كثير من الأحيان. فإذا وجدت مجموعة من المكونات غير المتوافقة مع بعضها في المشروع عندها ستظهر لدينا مشاكل عديدة في عملية تشارك البيانات أو تشغيل منظومة التحكم.

سنعرض فيما يلي هذه المشاكل وكيف تم حلها باستخدام تقنية OPC : [8]

1) البروتوكولات ذات الملكية المسجلة (Proprietary Protocols) :

يستخدم المصنعون عدة بروتوكولات اتصال خاصة بهم ومُسجلة لهم لضمان حقوق المُلكية وتسمح هذه البروتوكولات بالاتصال فيما بين المنتجات المتماثلة التي تستخدم هذه البروتوكولات ، ولكننا نحتاج إلى موائمات خاصة

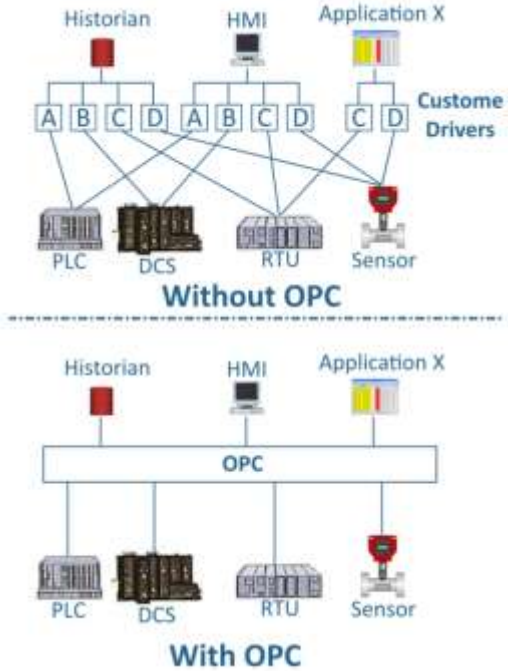
لتأمين الاتصال فيما بين المُنتجات التابعة لشركات مختلفة وأحيانا قد تكون المنتجات التابعة لشركة معينة لا تتصل مع مُنتجات أخرى لهذه الشركة لأسباب تتصل بأوقات التصنيع وتغيير البروتوكولات المُعتمدة وغيرها وبالتالي تحتاج لموائمات لتأمين الاتصال فيما بينها.

تم حل هذه المشكلة بواسطة تقنية "OPC" بحيث أصبح من غير الضروري لوجهة البيانات أن تعرف أي شيء عن مصدر هذه البيانات والكيفية التي يقوم بها هذا المصدر بالاتصال ومعالجة البيانات.

2) المعّرفات (التعاريف) المخصصة (Custom Drivers):

إن كل اتصال بين نقطتين يحتاج إلى معرّفات مُحددة لتأسيس الاتصال بين النقاط المختلفة، على سبيل المثال إذا احتجنا لتأمين الاتصال بين واجهة مستخدم (HMI) وجهاز (PLC) فإننا نحتاج إلى تعريف خاص لواجهة المستخدم ويكون هذا التعريف مُبرمجا من أجل بروتوكول الاتصال الخاص بالمتحكم "PLC"، وإذا كنا بحاجة لتخزين البيانات وأرشفتها في جهاز تأريخ (Historian) فإن هذا الجهاز يحتاج أيضا إلى تعريف خاص به وفق بروتوكول الاتصال المطلوب.

إذا لم توجد هذه المعرّفات فإن تأسيس الاتصال سيكون صعباً ومكلفاً.



الشكل (2) : حل مشكلة المعرّفات الخاصة باستخدام تقنية "OPC".

إن تقنية "OPC" تنهي الحاجة إلى المعرّفات الخاصة بين التطبيقات الجديدة ومصادر البيانات. حيث نستعيض عن الاتصالات كما في الشكل (2) بـ (وصلة OPC) واحدة، حيث يستطيع المتحكم "PLC" مثلا أن يتشارك البيانات مع كل من واجهة المستخدم (HMI) والمؤرخ (Historian) عبر وصلة OPC واحدة مع إضافة فائدة أخرى وهي أن وصلة تقنية "OPC" تحتاج لاتصال واحد مع المتحكم "PLC" وبالتالي تُخفض من الحمل على جهاز "PLC" .

3) التكامل المُعقّد (Complex Integration):

استخدام المعرّفات الخاصة يؤدي إلى أن عدد كبير من المعرّفات سيُستخدم من قبل عدد قليل من الأجهزة والتطبيقات المختلفة لتقوم بالاتصال فيما بينها. فإذا كان العمل على واجهة المستخدم (HMI) يتم عبر عدة حواسيب تتصل جميعها مع نفس المتحكم "PLC"، عندها سنحتاج إلى تهيئة جميع هذه الحواسيب بنفس الإعدادات وتحميلها

بنفس المعرّفات فإذا اتصل برنامج (HMI) مع أجهزة (PLC) إضافية عندها فإن كل واجهة مستخدم تحتاج إلى المعرّفات الخاصة بالأجهزة الجديدة المتصلة معها وهذا يؤدي إلى تعقيد كبير في بنية النظام وكابوس حقيقي عند الحاجة إلى الصيانة.

استخدام تقنية "OPC" يبسط هذا التكامل حيث يتم تهيئة وصلة "OPC" واحدة من أجل كل مصدر للبيانات وكل التطبيقات يمكنها مشاركة البيانات من هذا المصدر عند استخدامها لتقنية "OPC" من دون الحاجة إلى معرّفات خاصة بكل منها.

4) التحميل الزائد على الأجهزة والمُتحكمات (Devices and controllers extra load):

كل برنامج يقوم بإنشاء الاتصال الخاص به إلى الجهاز أو المتحكم المطلوب للقيام بعملية ما. ووجود العديد من البرامج والأجهزة بمعرّفات متخصصة عديدة يؤدي إلى إمطار الأجهزة وإتخامها بالطلبات من أجل إعطاء نفس المعلومات لكل تطبيق من التطبيقات التي تتصل معها، هذا عدا عن أن بعض الأجهزة قد لا تقبل سوى عدد محدود من الاتصالات بنفس الوقت وبالتالي لا بد من إجراءات إضافية عند الحاجة لتأسيس اتصالات إضافية مع هذه الأجهزة.

إن استخدام تقنية "OPC" يؤدي إلى إنقاص الازدحام وبالتالي التخفيف من الحمل على الأجهزة بشكل كبير جدا، بحيث أنّ وصلة OPC واحدة من أجل جهاز محدد تحتاج لاتصال واحد فقط مع مصدر البيانات وبنفس الوقت تقوم بتزويد هذه البيانات إلى العديد من التطبيقات دون وجود أي مشكلة في الاتصال.

5) التقادم في البنية التحتية المتوارثة (Obsolescence of legacy infrastructure) :

عند تجديد وإعادة ترميم بعض المشاريع القديمة يتم الاحتفاظ بجزء من البنية التحتية القديمة بدون تجديد وهذه هي ما تُسمى البنية التحتية المتوارثة.

تقوم الشركات بإطلاق منتجات جديدة وتتوقف عن دعم المنتجات القديمة. وبالتالي عند تجديد المشروع إذا قمنا بتنصيب برنامج واجهة مستخدم (HMI) جديد فإنه قد يحتاج إلى مجموعة جديدة من المعرّفات الخاصة به من أجل الاتصال مع الأجهزة القديمة التي كانت تتصل مع برنامج واجهة المستخدم القديم وقد يكون البرنامج الجديد لا يدعم الاتصال مع هذه الأجهزة.

تقنية "OPC" تساعد في زيادة مدة حياة الأنظمة القديمة لأنه عند تهيئة وصلة OPC لهذه الأنظمة فإنه يمكن لأي برنامج يدعم تقنية "OPC" أن يتصل مع هذه الأنظمة القديمة بغض النظر عما إذا كان هذا البرنامج بالأصل يدعم الاتصال (بشكل مستقل بدون OPC) مع هذه الأنظمة أو لا ، وبالتالي تسمح تقنية "OPC" للتطبيقات الجديدة أن تستمر بالاتصال مع الأنظمة القديمة.

قمنا فيما سبق بتسليط الضوء على مفهوم "OPC" بإظهار الميزات التي يتمتع بها وقدرته على حل مشاكل عدم التوافق بين تجهيزات منظومة التحكم والفوائد التي يحققها لنظم التحكم المتعددة المنتجين وسنرى لاحقا تطبيقاً عملياً لهذا المفهوم. ولكن قبل ذلك سنلقي الضوء بشكل مختصر عن أحد أهم البروتوكولات المُستخدمة في اتصالات الشبكات الصناعية وهو بروتوكول (Modbus) الذي تم تطويره من قبل شركة (Modicon) لصالح الأجهزة المنتجة من قبلها لكنه لم يلبث أن تحول إلى بروتوكول واسع الانتشار بين الشركات المنتجة لأجهزة التحكم الصناعي فهذا البروتوكول يعمل على تنظيم وإدارة عملية الاتصال لنقل البيانات في الشبكة الصناعية بين المتحكمات والأجهزة المتصلة بها ويصف العملية التي يقوم بها المتحكم بطلب البيانات من هذه الأجهزة وكيف يستجيب المتحكم لمتطلبات الأجهزة

وكيف يتم اكتشاف ومعالجة الأخطاء في عملية نقل البيانات. كما يقوم البروتوكول بتحديد عناوين الأجهزة الموجودة على الشبكة وتزويد الرسائل المرسلة عبر الشبكة بعناوين الأجهزة التي ترسلها أو التي يجب أن تصل إليها. ويتم نقل البيانات بشكل حِزم تحتوي العديد من المعلومات مثل عنوان الجهاز المرسل والمتلقي بالإضافة إلى البيانات الأساسية وبيانات التحقق من الأخطاء وغيرها ولن نخوض في تفاصيل البروتوكول أكثر من ذلك. [9]

ما يهمنا في هذا العمل هو أنماط هذا البروتوكول وكيفية عنونة المتغيرات المستخدمة في عملية التحكم. حيث يتم الاتصال عبر بروتوكول (Modbus) باستخدام أحد نمطي اتصال معياريين الأول يسمى (Modbus RTU) والثاني (Modbus ASCII) ويكون الاتصال بواسطة الوصلة التسلسلية (RS232) حيث يتم اختيار النمط المرغوب والبارامترات الأساسية للمنفذ التسلسلي المستخدم مثل معدل نقل البيانات (Baudrate) ونمط (Parity) وغيرها. [9]

بالنسبة للعنونة يستخدم البروتوكول طريقة عنونة مختلفة حسب نوع البيانات وبشكل عام يستخدم الأنواع التالية من البيانات:

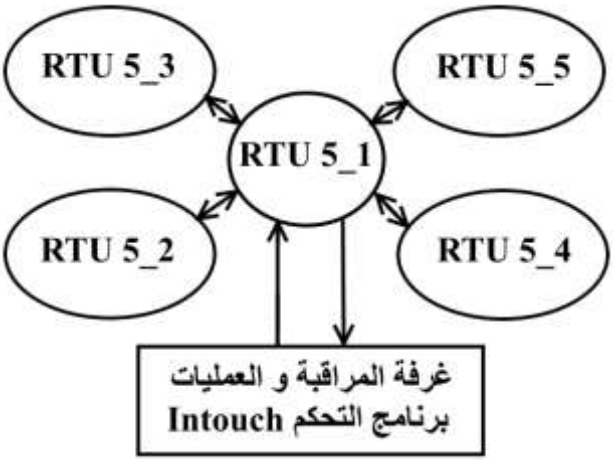
- المخارج الرقمية (Coils) والتي تكون عبارة عن إشارة ON/OFF وهذه الإشارة تتم عنونتها بالشكل (00xxxx) حيث تشير xxxx إلى رقم لا يتكرر لأي مخرج رقمي.
- المداخل الرقمية (Digital Inputs) وهي إشارات ON/OFF خارجية تدخل إلى المتحكم من الحقل وتتم عنونتها بالصيغة (10xxxx) حيث تشير xxxx إلى رقم لا يتكرر لأي مدخل.
- إشارة المداخل التماثلية أو ما تسمى (Input Registers) وتتم عنونتها بالصيغة (30xxxx).
- إشارة (Holding Registers) وهي مسجلات قد تكون رقمية أو تماثلية وتتم عنونتها بالصيغة (40xxxx).

3-7- استخدام تقنية "OPC" لحل مشاكل نظام التحكم في الشركة السورية لنقل النفط:

كتطبيق عملي لتقنية "OPC" سنشرح فيما يلي كيف سنستخدم هذه التقنية في حل مشاكل الاتصال في إحدى منظومات التحكم في الشركة السورية لنقل النفط في بانياس.

تعمل الشركة السورية لنقل النفط على نقل النفط السوري الخام من آبار دير الزور شرقا إلى مرفأ بانياس غربا ومن حقول الرميلان إلى طرطوس وذلك عبر العديد من محطات الضخ المنتشرة على طول خط النفط ومروراً بمدينة حمص. كما تقوم الشركة بنقل المشتقات النفطية من بانياس باتجاه مدينة حمص وكل ذلك عبر أنابيب متعددة الأقطار تحت الأرض.

يتم التحكم بعملية الضخ ومراقبتها في بانياس عبر منظومة تحكم مؤلفة مما يلي : وحدات طرفية للتحكم عن بعد ("RTU's Remote Terminal Units) عددها خمسة موزعة في منطقة الخزانات وتحتوي كل من هذه الوحدات على منظومة تحكم مبرمج "PLC" تقوم بالتحكم بالصمامات الحقلية والمضخات وقراءة البيانات والبارامترات اللازمة للقيام بعملية التحكم هذه. يتم تجميع الإشارات القادمة من هذه الوحدات (RTU's) ونقلها إلى غرفة المراقبة والعمليات التي يتم فيها مراقبة العمل بواسطة برنامج مراقبة وتحكم (SCADA) وهذا البرنامج هو (Intouch7) حيث يتم من خلاله التحكم بفتح وإغلاق الصمامات وتشغيل المضخات ومراقبة عملها وغير ذلك من المهمات اللازمة للقيام بالتحكم. والشكل التالي يبين مخططاً بسيطاً للنظام القديم:



الشكل (3) : مخطط منظومة التحكم القديمة

عملية نقل البيانات تتم عبر شبكة تستخدم مودمات صناعية وباستخدام بروتوكول (Modbus RTU).

هذه المنظومة أصبحت قديمة حيث تم البدء باستخدامها عام 1996 وتواجه العديد من المشاكل أهمها :

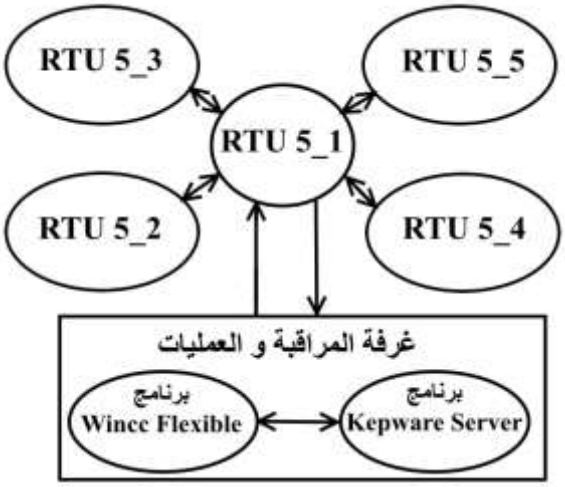
- الكمبيوتر الذي يعمل عليه نظام التحكم والمراقبة يتم تشغيله بواسطة نظام تشغيل Windows NT4 والذي أصبح قديما جدا ولم يعد مستخدماً في العالم وبالتالي تصعب صيانته والتعامل معه.
- المكونات الصلبة لهذا الكمبيوتر لم تعد متوفرة في السوق مما يعني استحالة إصلاحه عند تعطل أحد هذه المكونات ، مثلاً ذاكرة الوصول العشوائي (RAM) المستخدمة هي من النوع القديم جدا والنادر ولم تعد متوفرة في الأسواق أبداً.
- نظام التحكم والمراقبة بحاجة إلى تعديل نظراً لحدوث تغييرات في شبكة الأنابيب والصمامات المستخدمة في عملية الضخ.

إذاً، نحن بحاجة إلى استخدام حاسب حديث من أجل عملية المراقبة ولا بد من برنامج مراقبة حديث يعمل عليه لأن نسخة برنامج (Intouch) الموجودة لا تعمل على أنظمة تشغيل حديثة. ومن جهة أخرى لدينا نظام مراقبة لمنطقة أخرى في الشركة يستخدم منتجات شركة (Siemens) وهو نظام حديث نسبياً. فإذا استطعنا استخدام برنامج واحد لكلا نظامي المراقبة سيكون ذلك أفضل لعمليات التشغيل والصيانة وبالتالي زيادة الموثوقية.

3-8- النظام الجديد المقترح:

تم اقتراح استبدال برنامج (Intouch) ببرنامج (WinCC Flexible) من شركة (Siemens) والذي يمكن تنصيبه على حواسب حديثة تستخدم نظام التشغيل (Windows 7) وهو متوافر في الشركة حيث يتم استخدامه في عملية مراقبة منطقة المرابط البحرية ولكن الصعوبة التي واجهتنا أن برنامج (WinCC Flexible) لا يدعم كل خصائص بروتوكول (Modbus) المستخدم في عملية نقل البيانات في منظومة التحكم بالضخ وبالتالي لا يمكن وصل هذا البرنامج بشكل مباشر مع "RTU's" المستخدمة. ولكن برنامج (WinCC Flexible) يدعم بروتوكول "OPC" فكان الحل هو باستخدام برنامج "OPC" وسيط يؤمن الاتصال بين "WinCC" والوحدات الطرفية "RTU's" التي تستخدم بروتوكول "Modbus" وبذلك يتم حل المشكلة.

تم اقتراح استخدام برنامج (Kepware OPC Server) وذلك لتمتعه بموثوقية عالية عالمياً ولأنه مدعوم من قبل (WinCC Flexible) ولسهولة عملية إعداد الاتصال وإدخال البارامترات. ويصبح النظام الجديد كما هو موضح بالشكل (4):



الشكل (4) : مخطط منظومة التحكم الجديدة

من الناحية الاقتصادية فإن استخدام نظام مراقبة جديد سيوفر الكثير من مصاريف الصيانة الكبيرة التي كانت تُصرف على تأمين مستلزمات النظام القديم ومكونات الحاسب. نذكر مثلاً أنه في عام 2009 بلغت قيمة مناقصة تأمين الحاسب اللازم للنظام مع قطع التبديل اللازمة له من لوحات أم و ذواكر وكروت شاشة أكثر من مليون ليرة وبعدها أصبح تأمين هذه المواد صعباً للغاية بسبب فقدانها من الأسواق. وبالتالي فإن المصاريف اللازمة لتأمين البرامج والمكونات الصلبة Hardware تعتبر مقبولةً بالمقارنة مع مصاريف الصيانة التي كانت موجودة سابقاً. من ناحيةٍ أخرى إذا أردنا استبدال منظومة التحكم الموجودة بمنظومة حديثة سيكلف ذلك عشرات الملايين من الليرات بينما في النظام المقترح قمنا بالإبقاء على معظم أجزاء المنظومة القديمة وقمنا باستبدال برنامج المراقبة فقط ببرنامج موجود لدينا مسبقاً و إضافة برنامج Kepware واستخدام حاسب حديث وهذا لا يُكلّف في أقصى الحالات أكثر من نصف مليون ليرة وقد لا يحتاج إلى صيانة وقطع تبديل لفترة من الزمن وبالتالي على المدى البعيد فإن الجدوى الاقتصادية من المشروع مقبولة.

3-9- خطوات الحل (النظام الجديد):

بعد أن اكتملت لدينا فكرة مكونات النظام الجديد سنبدأ بعملية الإعداد والتصميم وربط هذه المكونات حيث قمنا بتجزئة العملية إلى الخطوات التالية:

1. جمع البارامترات المستخدمة في عملية التحكم وتنظيمها وتسجيل العناوين المستخدمة لكل متغير من المتغيرات ويبين الجدول (1) مثالاً عن بعض البارامترات المستخدمة وعناوينها:

الجدول (1) : نموذج للمتغيرات المستخدمة وطريقة عنونتها

اسم المتغير	نوع المتغير	وصف المتغير	العنوان المستخدم
MOV1613_CS	Input	الصمام 1613 مغلق (Close Status)	100001
MOV1613_OS	Input	الصمام 1613 مفتوح (Open Status)	100002
MOV1613_CC	Coil	أمر الإغلاق للصمام 1613 (Close Command)	000001

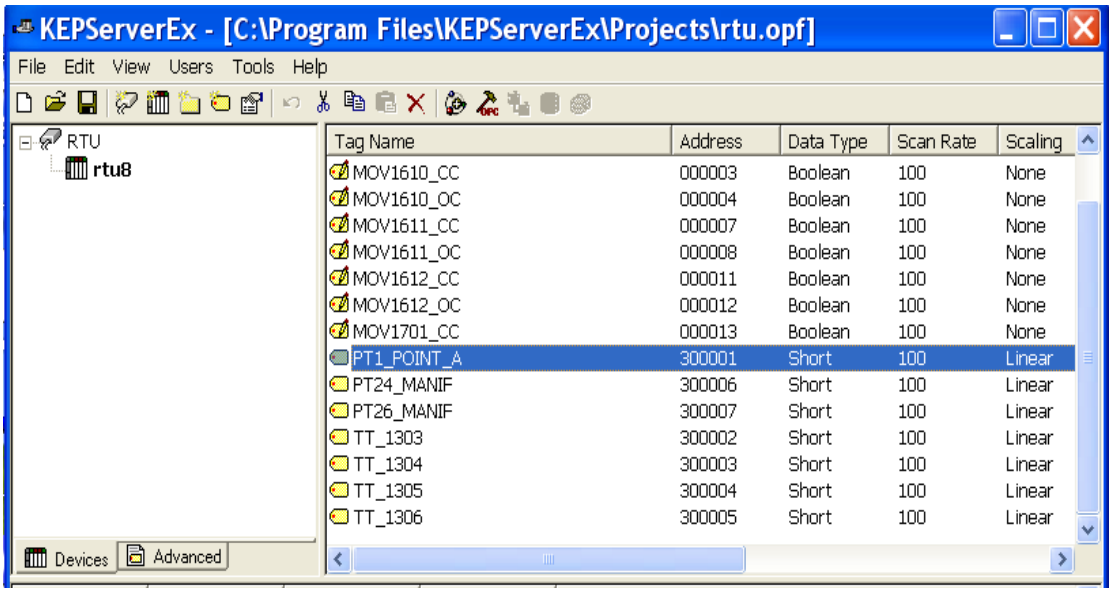
MOV1613_OC	Coil	أمر الفتح للصمام 1613 (Open Command)	000002
PT1_POINT_A	Input Register	قيمة الضغط في النقطة A	300001
TT_1303	Input Register	قيمة مرسل الحرارة 1303	300002

2. إعداد برنامج (Kepware OPC Server) حسب البارامترات المستخدمة في النظام وهي كما في الجدول التالي:

الجدول (2) : إعدادات الاتصال لبرنامج kepware

البروتوكول المستخدم	معدل نقل البيانات Baudrate	Data Bits	Stop Bit	Parity
Modbus RTU	9600	8	1	None

3. إدخال أسماء المتغيرات وعناوينها إلى برنامج Kepware وهي عملية تستغرق وقتا بسبب كثرة المتغيرات وبالنهاية يصبح البرنامج كما في الشكل (5):



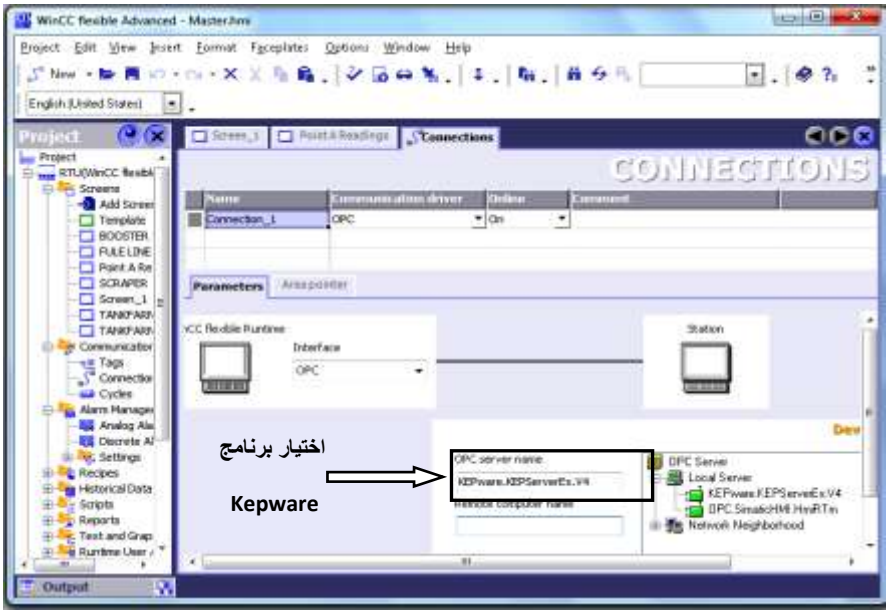
الشكل (5) : واجهة إعداد برنامج kepware servrer ويظهر جزء من البارامترات المستخدمة وعناوينها

4. إعداد برنامج المراقبة والتحكم (WinCC Flexible) وربطه مع برنامج (kepware) وسيتم شرح العملية في الفقرة التالية بشكل منفصل.

5. بعد الانتهاء سيتم تجريب البرنامج بشكله النهائي والتأكد من عمله بدقة تامة.

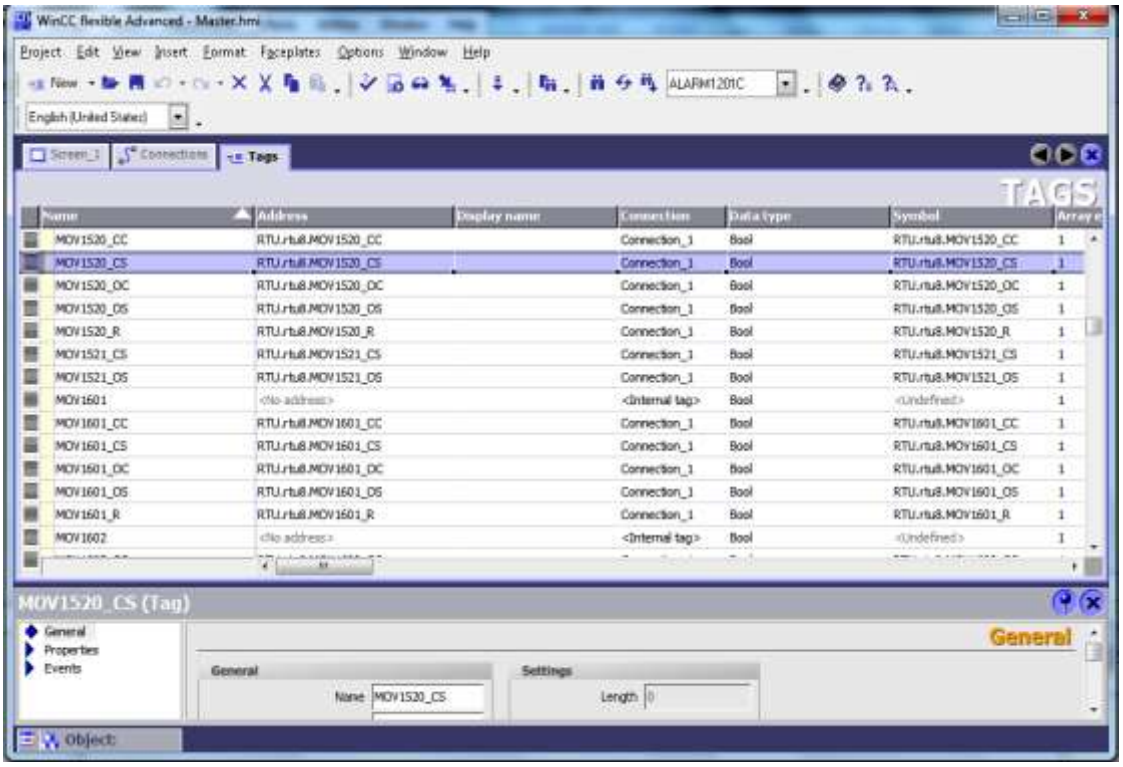
3-10- إعداد برنامج WinCC Flexible : [10]

نبدأ أولا باختيار مشروع جديد فارغ ثم نقوم بإعداد الاتصال المراد استخدامه حيث يحتوي البرنامج على العديد من إمكانيات الاتصال مع أنواع واسعة من أجهزة "PLC" ومن بينها إمكانية استخدامه كـ (OPC Client) حيث نختار من نافذة Connections نوع الاتصال OPC ثم تظهر لدينا قائمة ببرامج "OPC Server" المتاحة حيث نختار برنامج (Kepware) من قائمة البرامج هذه وذلك كما في الشكل (6):



الشكل (6) : واجهة إعداد الاتصال في برنامج WinCC Flexible واختيار OPC كنوع للاتصال

بعد ذلك نقوم بإدخال المتغيرات المستخدمة في نافذة Tags ونعطيها اسماء مناسبة توافق ما فعلناه سابقا عند إعداد برنامج (Kepware) ويتم ربط كل متغير مع المتغير المقابل له في برنامج (Kepware) أي بمعنى آخر نعطي لهذه المتغيرات العناوين المناسبة لها عن طريق عملية الربط هذه كما في الشكل (7):



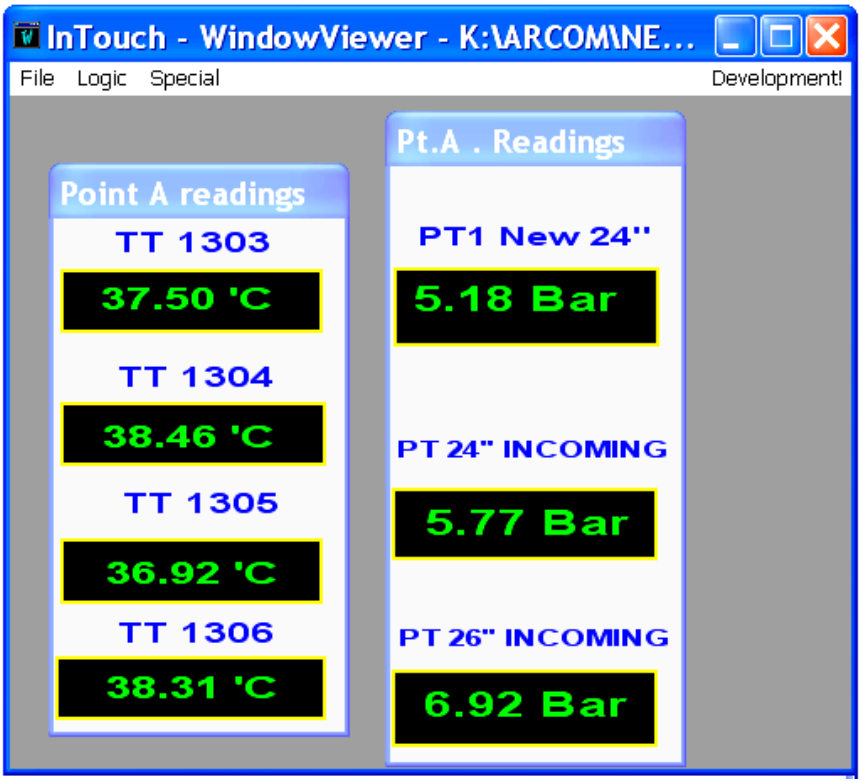
الشكل (7) : واجهة إعداد البارامترات في WinCC Flexible وربطها مع مقابلاتها في برنامج Kepware.

بذلك يكون تم ربط برنامج (WinCC Flexible) مع برنامج (Kepware) وهذا الأخير يؤمن الاتصال مع الأجهزة الحقلية ووحدات الاتصال (RTU's).

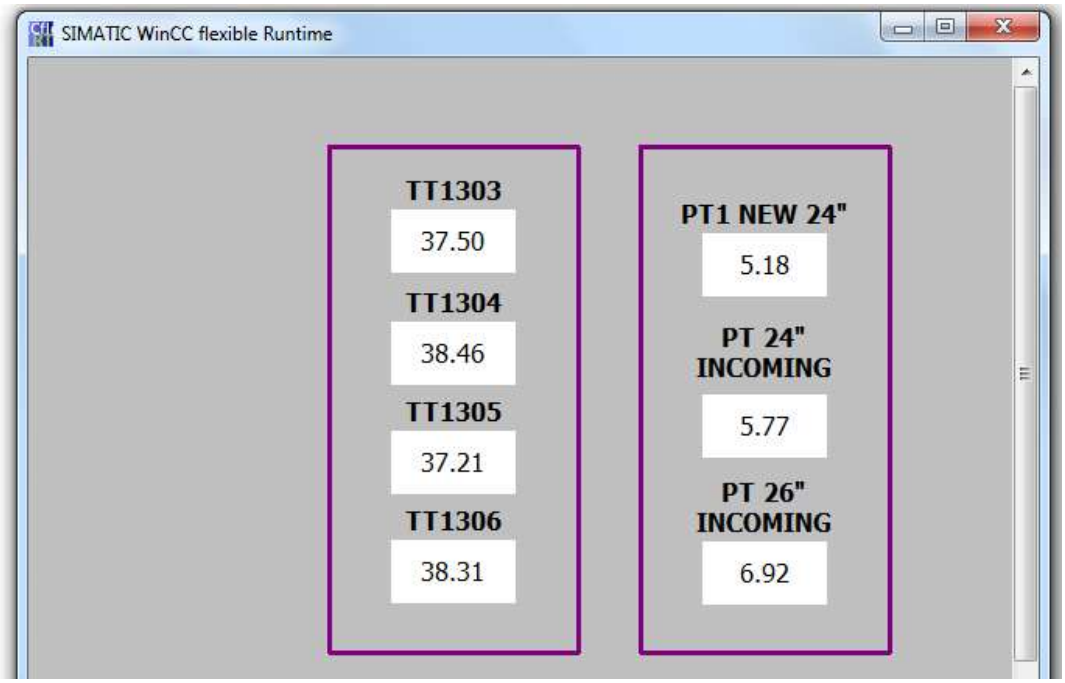
بعد ذلك نقوم بتصميم الشاشات التي ستظهر لدينا في برنامج المراقبة والتحكم والتي سيتم استخدامها من قِبل المشغلين في عملية التحكم بفتح وإغلاق الصمامات وتشغيل المضخات ومراقبة عملها حيث تم إلغاء بعض الشاشات القديمة التي لم تعد مستخدمة ودمج البعض الآخر بشاشة واحدة وإضافة شاشات مراقبة أخرى تناسب العمل بشكل أكثر.

النتائج والمناقشة :

بعد الانتهاء من عملية تصميم برنامج المراقبة والتحكم ننتقل إلى عملية التجريب والتأكد من عمل البرنامج بالشكل المطلوب حيث قمنا بتنصيب البرامج على حاسب حديث يستخدم Windows7 ويعمل بكفاءة عالية وتمت قراءة البيانات المتاحة عن طريق الحاسب القديم ثم تم وصل الحاسب الجديد وقراءة نفس البارامترات وحصلنا على قراءات متطابقة تقريبا ونرى في الشكلين التاليين (8) و (9) مثالاً عن هذه القراءات :



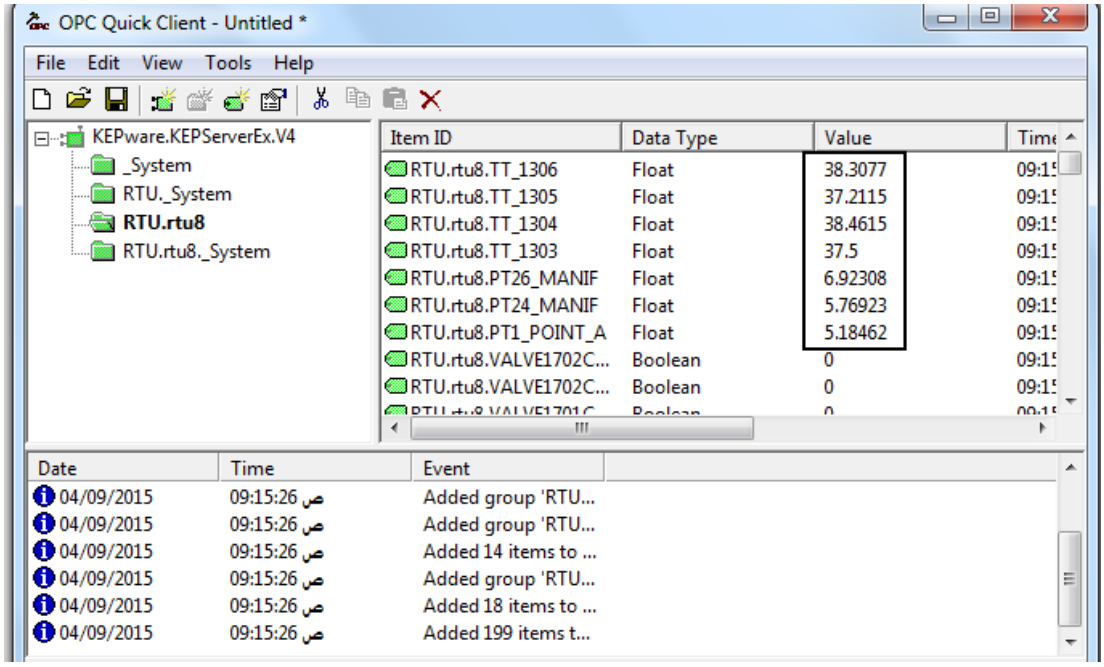
الشكل (8) : قراءة الضغط والحرارة عن طريق برنامج (Intouch) في الحاسب القديم



الشكل (9) : قراءة الضغط والحرارة عن طريق برنامج WinCC Flexible في الحاسب الجديد

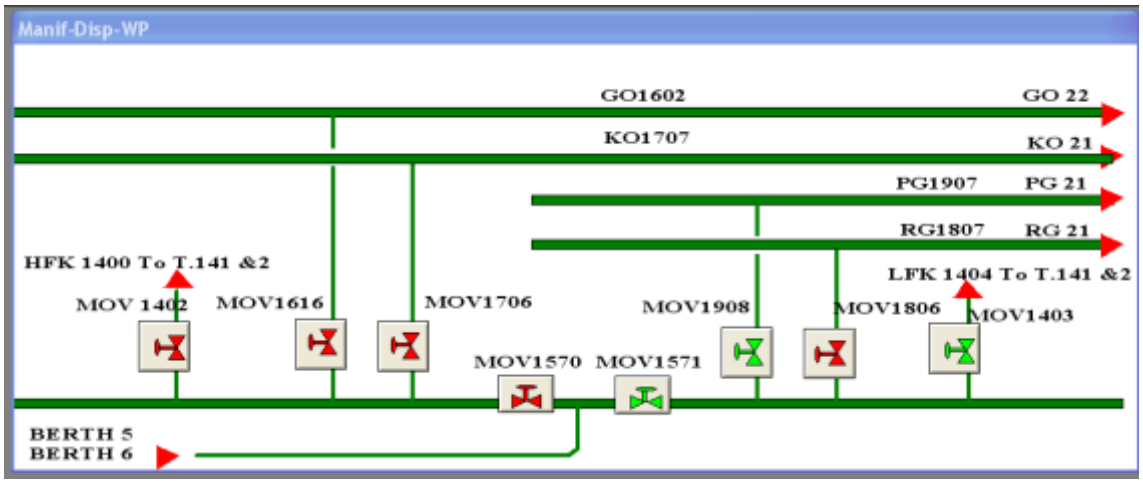
ونلاحظ وجود فرق في قراءة أحد البارامترات وهو درجة الحرارة TT1305 وهذا الفرق أقل من 0.3 درجة مئوية وهذا الخطأ يُمثل 2 بالألف من القيمة العظمى لقراءة الحرارة والبالغة 150 درجة مئوية هذا عدا عن أن القراءتين لم تتما في نفس الوقت تماما وإنما يوجد فرق في عملية فصل الحاسب القديم ووصل الحاسب الجديد ويمكن أن يحدث مثل هذا الفرق البسيط في درجة الحرارة المقاسة وبالتالي فالقيمة تعتبر مقبولة.

كما بإمكاننا أيضا مراقبة قيمة البارامترات التي نريدها عن طريق برنامج Kepware OPC كما في الشكل (10) :

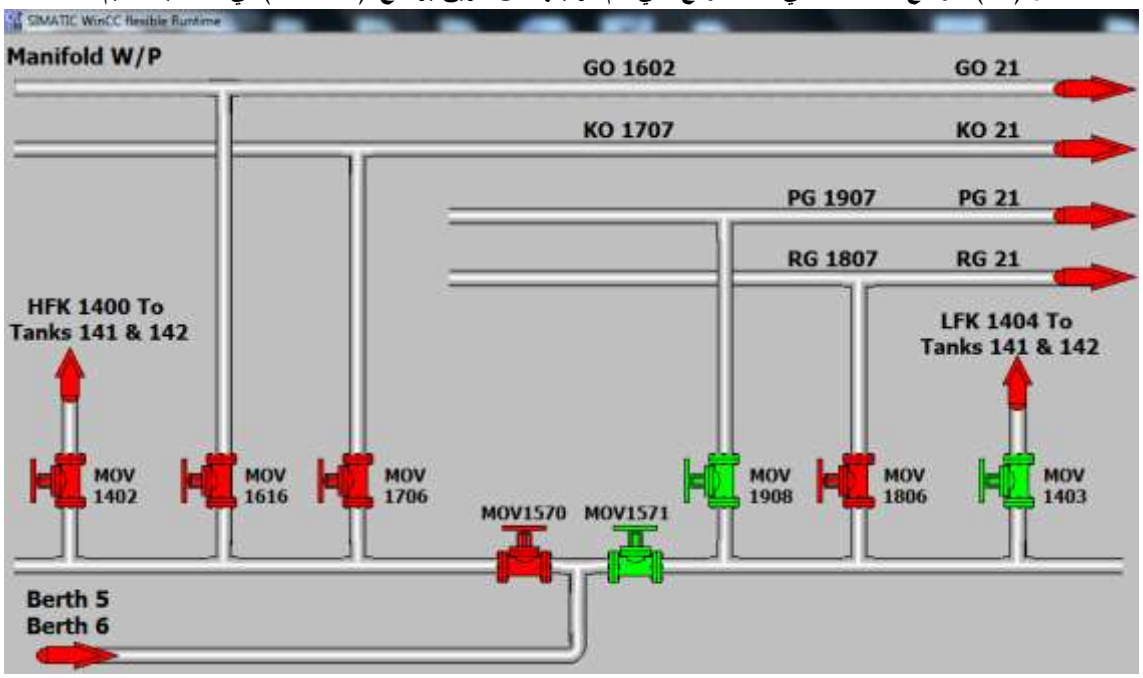


الشكل (10) : قراءة البارامترات في برنامج تقنية "OPC"

إذاً، نجد أن القراءة تمت بشكل صحيح من الحقل كما تم التحكم بفتح الصمامات وإغلاقها ومراقبة عمل المضخات دون وجود أية مشاكل والشكلان (11) و (12) يبينان أحد شاشات التحكم بالصمامات وفيهما تظهر الصمامات المفتوحة بلون أخضر والصمامات المغلقة بلون أحمر وذلك لكلٍّ من برنامجي المراقبة القديم والجديد:



الشكل (11) : وضع الصمامات في أحد المواقع التي تتم مراقبتها عن طريق برنامج (Intouch) في الحاسب القديم



الشكل (12) : وضع الصمامات في أحد المواقع التي تتم مراقبتها عن طريق برنامج WinCC Flexible في الحاسب الجديد

الاستنتاجات والتوصيات :

بعد استخدام نظام OPC لتأمين الاتصال بين منظومات التحكم المختلفة تم تحقيق ما يلي:

- فتح و إغلاق الصمامات عن بعد من غرفة المراقبة والعمليات.
- قراءة الضغوط ودرجات الحرارة في جميع المواقع التي تتطلب ذلك.
- مراقبة عمل المضخات وقراءة البارامترات المتعلقة بها من سرعات وضغوط وغيرها.
- تجريب عملية الإيقاف للمضخات في الحالات الطارئة التي تتطلب ذلك.

- إعطاء إنذار عند ارتفاع قيم أحد البارامترات أو توقف المضخات مع معرفة السبب الذي أدى إلى توقفها.

و كل ما سبق يتم في الوقت (الزمن) الحقيقي ودون وجود أي تأخير يذكر في عملية نقل البيانات.

بالتالي وبالنتيجة التجريبية نجد أن استخدام تقنية "OPC" قدّم لنا حلاً للمشاكل التي كانت موجودة وذلك رغم الوقت والجهد الذي استغرقته عملية إعادة التصميم وكتابة المتغيرات وربطها إلا أننا وصلنا إلى النتيجة النهائية المرجوة وبذلك نضمن عمل النظام لفترة أطول وبشكل أفضل.

ونوصي بتوسيع عملية استخدام هذا النظام ليشمل كل المرافق التي تتطلب حلولاً مماثلة لعملية الاتصال و ذلك لما فيه من توفير للمال من جهة وتمتعه بموثوقية عالية من جهةٍ أخرى.

المراجع:

1. OPC Foundation. *What is OPC* , 13 January 2016. <https://opcfoundation.org/about/what-is-opc/>
2. Kepware Technologies. *KEPServerEX Quick Client Connectivity Guide KTSM-00021 V. 1.04* . Kepware Technologies , 2011, 11.
3. GEOVAP Technologies . *Reliance 4 OPC Tutorial*. GEOVAP, Spol. S R.O, 2015,21.
4. Matrikon OPC. *opc-multimedia-tutorial*, 13 January 2016 . <http://www.matrikonopc.com/training/opc-multimedia-tutorial-new/index.html>
5. Wikipedia. *Component Object Model* , 18 June 2016 . <https://en.wikipedia.org/wiki/Component_Object_Model>
6. Wikipedia. *Distributed Component Object Model* , 18 June 2016 . <https://en.wikipedia.org/wiki/Distributed_Component_Object_Model>
7. Automation dot com . Introduction *to OPC for Factory Automation and Plant Process Control*, 13 January 2016 . <https://www.youtube.com/watch?v=16fM_7vZ_FE>
8. Komenik ,D . *The Every Man's Guide to OPC* . Matrikon OPC , 2009 , 9 .
9. Modicon . *Modbus Protocol Reference Guide PI–MBUS–300 Rev: J* . Modicon Inc , 1996 , 121.
10. Siemens. *WinCC Flexible 2008 Communication Part 1 User's Manual* . Siemens, 2008 , 284 .