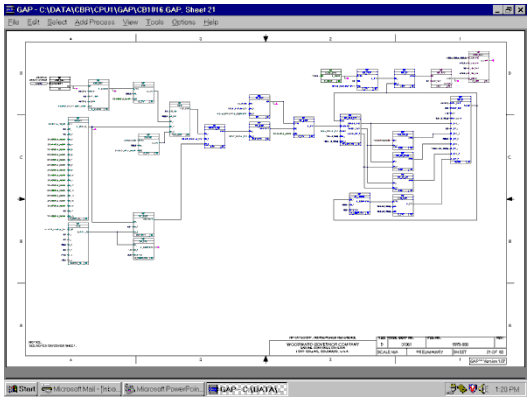
GAP™

Graphical Application Programmer (versions 1.x & 2.x)

Application



Woodward's Graphical Application Programmer (GAP) utilizes a fourth-generation language developed by Woodward. The self-documenting GAP software allows control engineers to custom design a control system for any application. Woodward has used GAP-created code since the mid 1980s. This high level machine language is in a block format, putting many lines of code in each block to achieve a desired function. The advantage of using a block format is that individual lines of code do not need to be debugged, since each program block has already been debugged during development, testing, and field applications.

Description

Woodward's Graphical Application Programmer (GAP™) is a Windows based software program that allows controls engineers to create block format application programs for a number of Woodward control systems. Once the control logic is entered using the program's graphical programming environment, the GAP software compiler function generates code that runs in the control.

GAP software blocks are written in C code, which is a transportable language, allowing it to be used on many different hardware platforms. Because of this feature, the same control logic software that has been used and proven in thousands of controls is available for use in the creation of future controllers, without introducing system bugs.

For ease of learning and use, the GAP software diagram entry screen has the look and feel of several software simulator tools familiar to many control engineers. Blocks much like the blocks used to enter models into Matlab or Matrix XTM are entered into the GAP software.

The GAP software package is a mature programming package, providing revision control, security, and code import/export tools to facilitate program management and code re-use. "Multi-GAP" functionality allows multiple sections of control logic and algorithms to be created and controlled by different entities, then compiled into a single program for unit control.

While GAP block language has proven to be an extremely efficient means of programming a turbine controller, some users still require that their existing sequencing logic run on their new control. For these users, Woodward's Ladder Logic Programming tool can be used with GAP to create sequence logic via industry-standard ladder logic programming. This tool gives the user the capability to perform sequencing in Ladder Logic program language and to perform complex control algorithms in GAP. Like most industry standard ladder logic programs, this program also allows the user to make sequence changes while the engine is on or off line.

GAP allows users to design their control system logic with an integrated drawing package in a standard PC Windows environment, using code that has decades of experience on thousands of engines and turbines.

- Designed for engine or turbine control engineers
- Windows® based, block-oriented
- Minimal programming skills needed
- Self-generating C code
- Self-documenting
- GAP™ blocks are proven, tested software routines
- Quick, simple, intuitive implementation
- Allows Matlab / Simulink code importing
- Compatible with NetSim™ simulation tools

The block format allows a control engineer to re-use a library of known control blocks, string them together with other known control blocks, and develop a software program to match customer needs. Through years of use, GAP has proven to be an extremely successful software package that can be field-modified, even after installation or based on changing customer needs.

An additional advantage incorporated into GAP is the unique rate group structure on which it operates. Variable execution time blocks have been avoided. This means that under an unanticipated set of conditions (system upset), the processor will not become overburdened, and will have time to complete its assigned tasks. Only with predictable execution times can a multitasking operating system have a guaranteed update rate. Software tasks may be programmed to run every 5 milliseconds, 10 ms, 20 ms, etc., and all tasks seem to run simultaneously. Woodward software guarantees that a task programmed to run every 5 ms (or whatever the assigned rate group), will run exactly every 5 ms, no matter what other functions the control system is presently exercising. Most systems based on PLCs cannot do this.

This versatile software program is structured to allow OEMs or other engineering companies the capability of creating or controlling their own fuel control algorithms. GAP also has the capability to import fuel control algorithms created in Matlab, thus allowing OEM engineering departments a method of creating and controlling their own control algorithms in a familiar (industry standard) environment.

When designing and commissioning an engine control system, much time is often spent rewriting, retesting, and redrawing documentation for software that has already been proven in a similar application. GAP allows a control systems engineer to efficiently reuse existing software and minimize repetitive documentation time.

Run-Time Performance

When many complex algorithms are implemented into a single controller, the result can sometimes be a less than robust controller. One of the main advantages of the GAP software is that all the blocks have been written as efficiently as possible without jeopardizing code stability. Blocks are written in one standard language, and the interface between blocks is well defined. Unwanted interactions between unrelated pieces of software, while common in many other control systems, have virtually been eliminated with the GAP software.

The timing requirements of the code generated by the GAP software have also been rigidly defined and adhered to. Variable execution time blocks have been avoided. The non-variability in the execution times of the blocks means that a control which runs efficiently on a test stand is currently exercising the entire control program. The user does not have to worry that under an unanticipated set of conditions, the processor will become overburdened and will not have enough time to complete its assigned tasks. Only with predictable execution times can a multitasking operating system have a guaranteed update rate.

Because of Woodward's longstanding relationship with turbine OEMs, special software algorithms have been written and verified through years of use in the field. A sample list of these special blocks includes:

- A unique PID algorithm with special provisions for disturbance handling and noise rejection
- Anti-reset wind-up low-signal-select and high-signal-select blocks
- Extraction/de-coupling algorithms
- Special acceleration controls for precise turbine start-ups
- Special algorithms that allow parallel or series valve transfers with minimal bumping
- Compressor surge control algorithms
- Several modes available for bumpless transfer between control modes
- Special disturbance rejection algorithms for step load changes
- Model reference controls for special turbine system problems

A tested collection of blocks used to control a given turbine can be used to control a different turbine, with a different control platform, in a consistent, repeatable fashion.

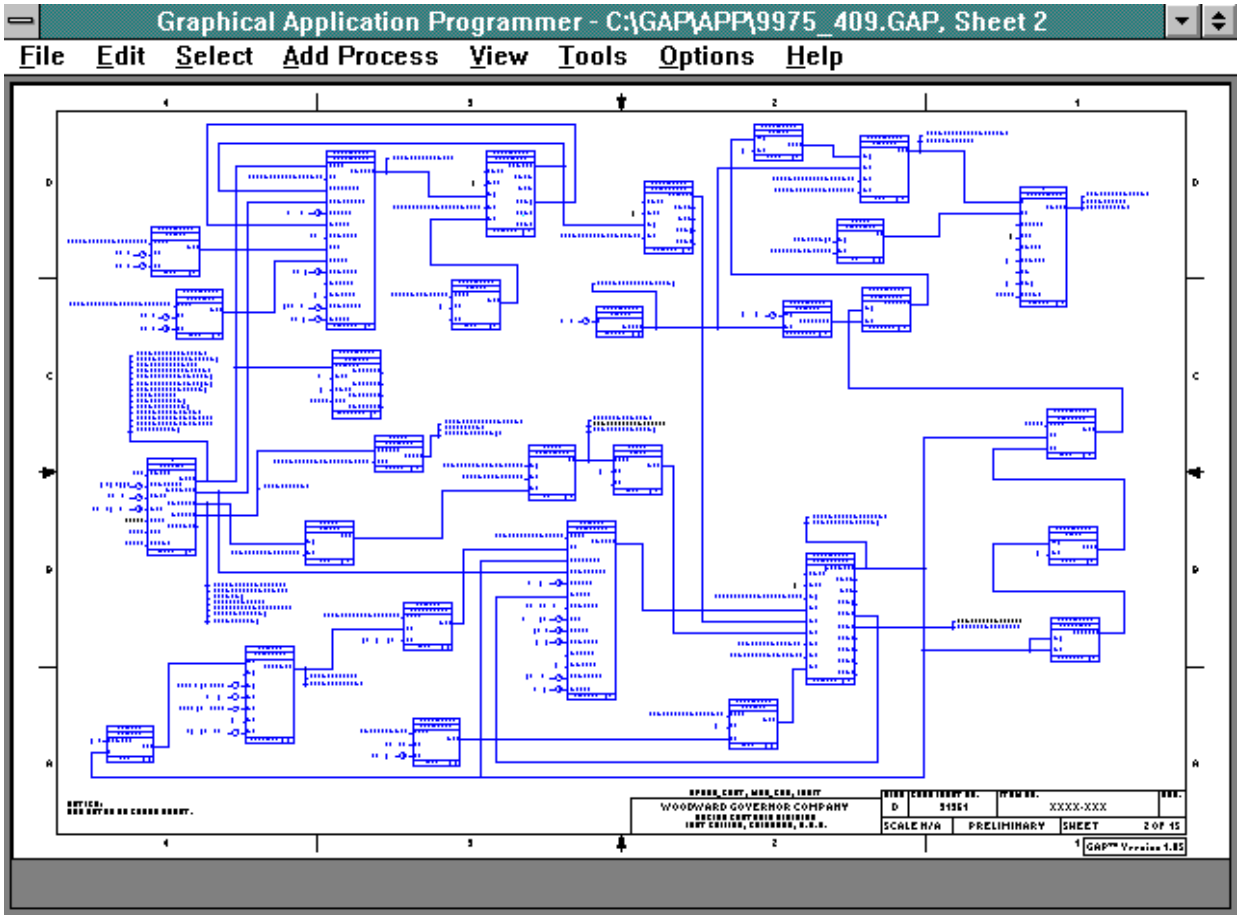
A final run-time advantage of the application generator is that the same code produced by the GAP software is used in a wide variety of Woodward platforms from low-end low-cost controls to high-end fault-tolerant systems. Here are just some of the Woodward products that are programmed using GAP:

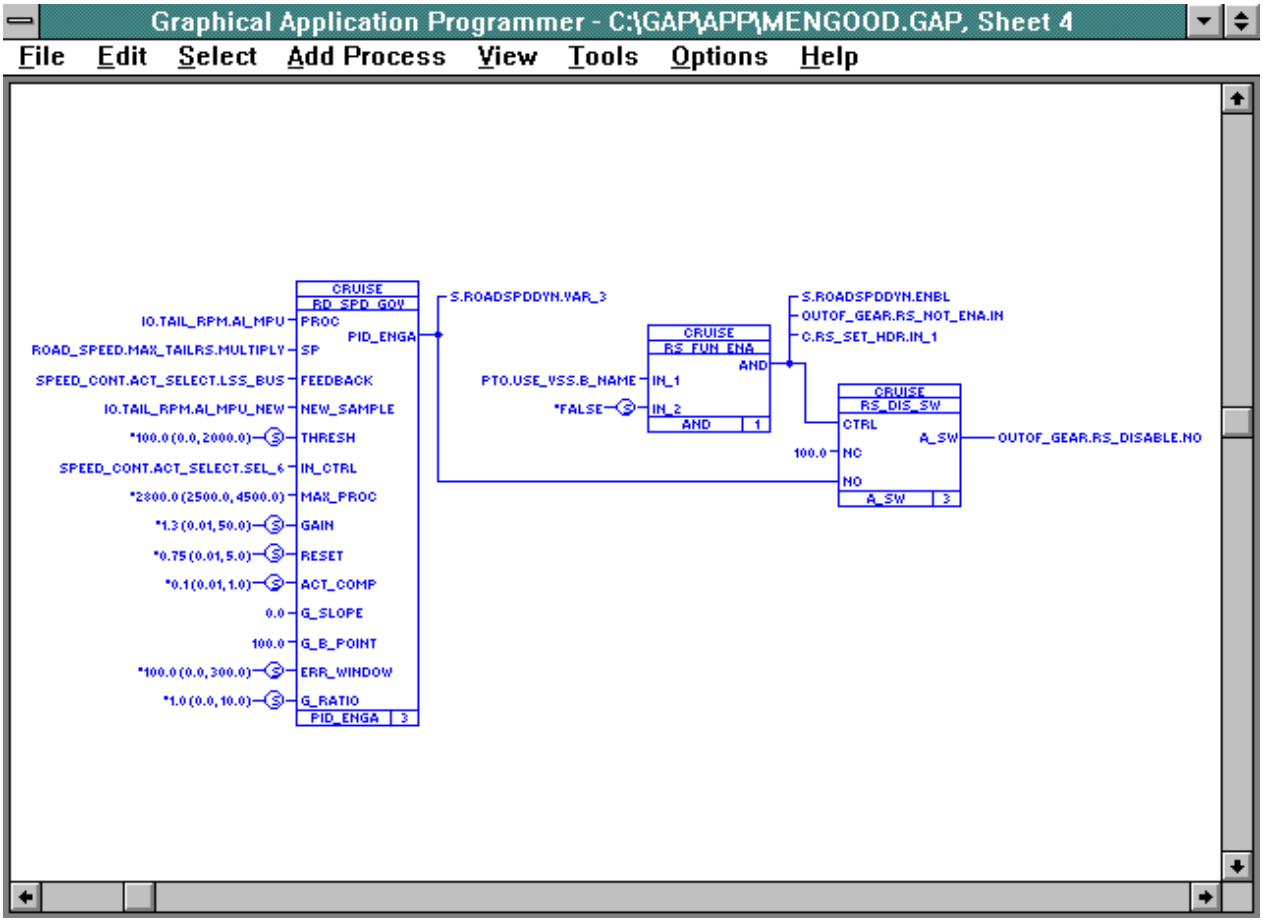
- MicroNet™
- NetCon®
- 500 Series Controllers
- Atlas Series Controllers
- In-Pulse™
- EGCP Series Controllers
- 723, 828 Controllers, and other standard products such as Peak® 150 and 5009

GAP Versions

Three levels (versions) of GAP are available to meet a variety of user needs:

- **Basic GAP**—Contains all of the standard GAP blocks that are needed to create application software for most control applications.
- **Advanced GAP**—Contains all of the functionality of Basic GAP plus a set of more specialized and sophisticated functions that are required in some applications. Examples of this added capability include Fortran, a flexible 'C_Function' block, gas flow computation blocks, 'Calculate' blocks, and other advanced functions.
- **Mathworks GAP**—Integrates the Mathworks Matlab/Simulink tools into the GAP environment, allowing users to synthesize new function blocks by using the code generation features of Simulink.


Graphical Application Programmer - C:\GAP\APP\9975_409.GAP, Sheet 2
File Edit Select Add Process View Tools Options Help
WOODWARD GOVERNOR COMPANY
SCALE N/A
PRELIMINARY
SHEET 2 OF 15


Graphical Application Programmer - C:\GAP\APP\MENGOOD.GAP, Sheet 4
File Edit Select Add Process View Tools Options Help
CRUISE
RD_SPD_GOV
IO.TAIL_RPM.AI_MPU
PROC
PID_ENGA
ROAD_SPEED.MAX_TAILRS.MULTIPLY
SP
SPEED_CONT.ACT_SELECT.LSS_BUS
FEEDBACK
IO.TAIL_RPM.AI_MPU_NEW
NEW_SAMPLE
*100.0 (0.0, 2000.0)
THRESH
SPEED_CONT.ACT_SELECT.SEL_6
IN_CTRL
*2800.0 (2500.0, 4500.0)
MAX_PROC
*1.3 (0.01, 50.0)
GAIN
*0.75 (0.01, 5.0)
RESET
*0.1 (0.01, 1.0)
ACT_COMP
0.0
G_SLOPE
100.0
G_B_POINT
*100.0 (0.0, 300.0)
ERR_WINDOW
*1.0 (0.0, 10.0)
G_RATIO
PID_ENGA 3
S.ROADSPDDYN.VAR_3
CRUISE
RS_FUN_ENA
PTO.USE_VSS.B_NAME
IN_1
*FALSE
IN_2
AND
AND 1
S.ROADSPDDYN.ENBL
OUTOF_GEAR.RS_NOT_ENA.IN
C.RS_SET_HDR.IN_1
CRUISE
RS_DIS_SW
CTRL
A_SW
OUTOF_GEAR.RS_DISABLE.NO
100.0
NC
NO
A_SW 3

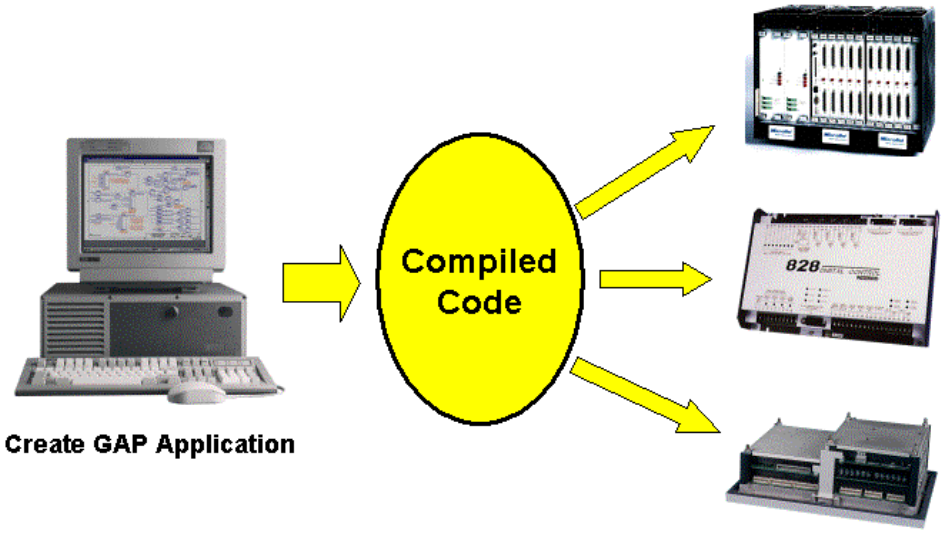
System Support and Modification

During system support and modification, the advantages of the GAP software become readily apparent. Because of the structure of the GAP software, and because it generates its own documentation, user changes to the control software can be supported by field service personnel. If the control system software were written with only the aid of a software compiler like C or Pascal, any errors introduced in this code by users might be almost impossible to identify and repair. The GAP system generates its own documentation that is guaranteed to match the control code generated. Only with a tool like this can user-modified software be supported by anyone other than that user.

Another important control attribute of the code generated by the GAP software is system security. All operator interfaces are password protected with multiple levels of user access so that unwanted tampering with the control is prohibited.

Because of its rigidly defined nature, the GAP system allows Woodward to exercise the same engineering change and revision control system procedures that are used to control the manufacture of any control system part. This ISO 9001 approved engineering change system guarantees software compatibility, as well as documentation that makes it possible to reproduce, diagnose, and repair control logic bugs. If any logic bugs are entered, they will be well documented, and the affected systems will be easily located and corrected.

Additionally, GAP is completely compatible with NetSim™, Woodward's simulation software that provides an environment in which engineers can accurately and easily test the application program against a virtual model of the target hardware. This allows safe and very cost effective trouble shooting of the application without hardware in the loop.



WOODWARD

PO Box 1519, Fort Collins CO, USA 80522-1519
1000 East Drake Road, Fort Collins CO 80525
Tel.: +1 (970) 482-5811 ◆ Fax: +1 (970) 498-3058
www.woodward.com

Distributors & Service

Woodward has an international network of distributors and service facilities. For your nearest representative, call the Fort Collins plant or see the Worldwide Directory on our website.

For more information contact: