Autopilot gauge APGA documentation v1.16

Introduction

The PMS50 autopilot gauge (APGA) is a hidden gauge to be installed in the cockpit by aircraft builders.
It's purpose is to simplify the work of building an autopilot panel by externalizing the autopilot logic.

The idea is to define the panel on a buttons perspective: an autopilot panel is just a set of buttons and/or knobs and eventually a screen to display the autopilot state.

The APGA directly manages the logic behind that. The work of the autopilot panel builder is then limited to send the H event to the APGA corresponding to the buttons or knobs and to get back some L variables to display the autopilot state.

The APGA has been tested with the Asobo stock and Working Title autopilot and flight plan managers. Using the APGA will bring the WTT mode compatibility for the autopilot gauge.

The APGA is a standalone component completely independent of the GTN750 or any other gauge.

The APGA code is open source.
You can use it, modify it and deliver it in your aircraft package without any authorization from PMS50.

Installing the APGA gauge in your aircraft

Unique aircraft ident

You must define an IDENT of the form:

"MyCompany-MyAircraft"

where MyCompany and MyAircraft must be replaced by your real names. Be sure to not use any space nor special characters, only letters a to z, digits 0 to 9 and the '-' character.

This is to be sure to identify uniquely your files inside the virtual file system.

Copying the files

From the root of your aircraft package, create a folder named:

```
html_ui\Pages\VCockpit\Instruments\NavSystems\YOUR-IDENT
```

where YOUR-IDENT is your unique aircraft ident as defined above.

Copy the files named PMS_APGA.html, PMS50_APGA.js and PMS50_APGA_custom.js to this newly created folder.

You will find these files in the Integration/Files/autopilot folder of the PMS50 main package.

Modifying the PMS50_APGA.html file

In your installed copy of the PMS50_APGA.html file, replace "MyCompany-MyAircraft" with your own IDENT in the last 2 lines.

Modifying the PMS50_APGA_custom.js file

The PMS50_APGA_custom.js file can be modified to suit your needs. Usually, it's only necessary to change the parameters but you may need to change the behavior too.
Please modify only the PMS50_APGA_custom.js file and keep unchanged the PMS50_APGA.js file.

Installing the gauge in your panel.cfg

Your panel.cfg file must include the code to call the APGA hidden gauge.
Place the following code in your panel.cfg file:

```
[VCockpitxx]
size_mm=0,0
pixel_size=0,0
texture=$NONE
background_color=42,42,40
htmlgauge00=NavSystems/YOUR-IDENT/PMS50_APGA.html,0,0,0,0
```

Where YOUR-IDENT is your unique aircraft ident as defined above.

And xx is to be replaced by the gauge number.

The texture name is a fake one because this gauge is hidden. You don't need to do anything with your 3D tools.

Working with the APGA

This section gives the necessary events and variables for your autopilot panel to communicate with the APGA.

Here is an example of such an autopilot panel:

This is the internal GTN750 autopilot panel page.
The code of this panel is open source. It can be found in the official GTN750 package at:

```
html_ui\Pms50\gtn750\Pages\VCockpit\Instruments\NavSystems\GPS\GTN750\js\autopilot.js
```

Building your gauge

Your autopilot gauge will call the APGA events when pressing a button.

You can build your gauge in any language that allows sending H events and reading L variables.

> It is strongly recommended to build your autopilot panel as an html gauge even if there is no screen and to implement the standard H events as described below.
> If your autopilot is an XML gauge, you can implement a hidden HTML gauge just to manage the standard H events. Such an example can be found with the KMC321 in the Integration/Files/autopilot/Examples folder of the PMS50 main package.

Using Html/Javascript for building your gauge is probably the simplest way. That requires just a few lines of code.
In Javascript, your gauge will extend the BaseInstrument class.

It is recommended to set an instrument alias to "APGA". This will allow to use some "standard" events to control the gauge from external application or hardware (ex SPAD.next or Stream Deck).

To set the instrument alias, add the following function to your autopilot class:

```
get instrumentAlias() { return "APGA"; }
```

Then check for the events in the onInteractionEvent method. Here is an example for the KAP140:

```
onInteractionEvent(_args) {
    if (this.isElectricityAvailable()) {
        let apOn = this.isAutopilotEngaged();
        switch (_args[0]) {
            case "KAP140_Push_HDG":
            case "APGA_HDG":
                if (apOn)
                    postHEvent("PMS50_APGA_AP_HDG");
                break;
            case "KAP140_Push_NAV":
            case "APGA_NAV":
                if (apOn)
                    postHEvent("PMS50_APGA_AP_NAV");
                break;
            case "KAP140_Push_APR":
            case "APGA_APR":
                if (apOn)
                    postHEvent("PMS50_APGA_AP_APR");
                break;
......
```

Please use the postHEvent function to post any H event. It fixes a bug in the sim that may send bad H events after a flight restart;

Here is the standard H event list:

> The standard events are the ones to be used for your external application or hardware (ex SPAD.next or Stream Deck).

Event	Description
H:APGA_AP	Toggles the autopilot master. To be called when pressing the AP button.
H:APGA_FD	Toggles the flight director. To be called when pressing the FD button.
H:APGA_YD	Toggles the yaw damper. To be called when pressing the YD button.
H:APGA_HDG	Toggles the HDG mode. To be called when pressing the HDG button.
H:APGA_NAV	Toggles the NAV mode. To be called when pressing the NAV button.

Event	Description
H:APGA_APR	Toggles the Approach mode. To be called when pressing the APP button.
H:APGA_REV	Toggles the back course mode. To be called when pressing the REV or BC button.
H:APGA_ALT	Toggles the altitude hold mode. To be called when pressing the ALT button.
H:APGA_VS	Toggles the vertical speed hold mode. To be called when pressing the VS button.
H:APGA_FLC	Toggles the flight level change mode. To be called when pressing the FLC button.
H:APGA_VNAV	Toggles the VNAV vertical mode. To be called when pressing the VNAV button. Works only with the PMS50 WTT mode
H:APGA_LVL	Toggles the wing leveler hold mode.
H:APGA_CGA	Toggles the Go Around mode.
H:APGA_CLVL	Toggles the combined LVL mode.
H:APGA_ARM	Toggles the arm mode for VS or FLC. To be called when pressing the ARM button.
H:APGA_UP	Nose up for VS, FLC and PITCH vertical modes. To be called when pressing the UP button.
H:APGA_DOWN	Nose down for VS, FLC and PITCH vertical modes. To be called when pressing the DN button.
H:APGA_SEL_ALT_SMALL_INC	Increments the selected altitude by a small value. To be called when rotating the inner altitude selection knob.
H:APGA_SEL_ALT_LARGE_INC	Increments the selected altitude by a large value. To be called when rotating the outer altitude selection knob.
H:APGA_SEL_ALT_VERY_LARGE_INC	Increments the selected altitude by a very large value.
H:APGA_SEL_ALT_SMALL_DEC	Decrements the selected altitude by a small value. To be called when rotating the inner altitude selection knob.

Event	Description
H:APGA_SEL_ALT_LARGE_DEC	Decrements the selected altitude by a large value. To be called when rotating the outer altitude selection knob.
H:APGA_SEL_ALT_VERY_LARGE_DEC	Decrements the selected altitude by a very large value.
H:APGA_SEL_VS_SMALL_INC	Increments the vertical speed by 100 feet/mn. To be called when rotating the inner VS select knob.
H:APGA_SEL_VS_SMALL_DEC	Decrements the vertical speed by 100 feet/mn. To be called when rotating the inner VS select knob.
H:APGA_SEL_VS_LARGE_INC	Increments the vertical speed by 1000 feet/mn. To be called when rotating the outer VS select knob.
H:APGA_SEL_VS_LARGE_DEC	Decrements the vertical speed by 1000 feet/mn. To be called when rotating the outer VS select knob.

call example "H:APGA_HDG" to engage the HDG mode.

Implementing the standard events in your gauge is a recommended option.

There are some APGA autopilot examples in the Integration/Files/autopilot/Examples of the PMS50 main package.

Parameters

The APGA provides some few parameters that can be changed by modifying them directly in your delivered copy of the PMS50_APGA_custom.js file.
This is done in the constructor of the MY_APGA class.
Please do not change anything inside the PMS50_APGA.js file.

Parameter description

Parameter	Description	Value / Unit
updateFrequency	The frequency at which the APGA will update its state. Default value: 10	1 to 10. 10 is the highest frequency.

Parameter	Description	Value / Unit
highAltPrecision	The altitude selection precision. The high precision is 10 feet. The normal precision is 100 feet. Default value: false	true or false
altPreselMode	The altitude preselection mode. Default value: altPreselMode.auto	See values below
noUpdatableAltitudeSelector	Set it to true if mechanical selector that can't be updated from APGA. Default value: false	true or false
vsEngageMode	The way we set the initial vertical speed when engaging the VS mode. Default value: vsEngageMode.current	See values below
vsEngageModeDefaultValue	The VS value in feet when the vsEngageMode is vsEngageMode.default. Default value: 500	feet
gaVsValue	The VS value to be used in the go around combined mode. Default value: 700	feet per minute
defaultLateralMode	The default lateral mode. Can be lateralMode.roll or lateralMode.lvl. Default value: lateralMode.roll	number
manualArmDefaultVerticalMode	The default vertical mode to engage when arming in manual arm mode. Can be verticalMode.none, verticalMode.vs or verticalMode.flc. Default value: verticalMode.none	number
useVsForArmedVerticalCheck	Use the current vertical speed to check the arming to the selected altitude. Default value: true	true or false
vsThresholdForArmedVerticalCheck	Vs threshold to use when the useVsForArmedVerticalCheck parameter is true. Keep it between 10 and 50. Default value: 50	number. Between 10 and 50

Parameter	Description	Value / Unit
targetAltitudeAlertThreshold	Target altitude threshold for the alert phase. Default value: 1000	number
targetAltitudeNearThreshold	Target altitude threshold for the near phase. Default value: 300	number
targetAltitudeReachedThreshold	Target altitude threshold for the reached phase. Default value: 20	number
targetAltitudeReachedTimeOut	Time out in second for clearing the reached phase to none. Default value: 5	number
unarmedAltituteLockValue	Altitude locked value for unarmed state when in manual arm mode. This value is sent to the sim when the vertical mode is not armed allowing to use vertical mode without arming. Default value: 99000	feet
altModeUpDownIncrementValue	Up/Down increment when in ALT mode and pressing the UP/DN buttons. This allows a precise adjustment of the alt mode altitude. Default value: 20	feet

altPreselMode

altPreselMode parameter values.

Mode	Description	Value
altPreselMode.none	No altitude preselection	0
altPreselMode.auto	Altitude preselection with an automatic arming. This is the default mode. The autopilot panel doesn't have an arm button.	1
altPreselMode.manual	Altitude preselection with a manual arming. The autopilot panel must have an arm button to arm the VS or FLC modes	2

vsEngageMode

vsEngageMode parameter values.

Mode	Description	Value
vsEngageMode.none	No specifc VS value. The VS engage value is set by the autopilot manager (stock or WTT).	0
vsEngageMode.reset	VS is set to 0.	1
vsEngageMode.default	VS is set to the value defined in the vsEngageModeDefaultValue parameter. The sign depends of your current and target altitude.	2
vsEngageMode.current	VS is set to the current vertical speed.	3

Events

The events are H events of the form "H:PMS50_APGA_EventName" where EventName is the event name described below.

> These events are used to communicate from your cockpit autopilot gauge to the hidden APGA gauge. If you need control events from an external application or hardware (ex SPAD.next or Stream Deck), it's better to use the standard events described in the chapter "Building your gauge" above. Otherwise you may skip some part of the autopilot logic.

Example: "H:PMS50_APGA_AP_ALT" is to be sent when pressing the ALT button.

Event	Description
AP_AP	Toggles the autopilot master. To be called when pressing the AP button.
AP_FD	Toggles the flight director. To be called when pressing the FD button.
AP_YD	Toggles the yaw damper. To be called when pressing the YD button.
AP_HDG	Toggles the HDG mode. To be called when pressing the HDG button.
AP_NAV	Toggles the NAV mode. To be called when pressing the NAV button.

Event	Description
AP_APR	Toggles the Approach mode. To be called when pressing the APP button.
AP_REV	Toggles the back course mode. To be called when pressing the REV or BC button.
AP_ALT	Toggles the altitude hold mode. To be called when pressing the ALT button.
AP_VS	Toggles the vertical speed hold mode. To be called when pressing the VS button.
AP_FLC	Toggles the flight level change mode. To be called when pressing the FLC button.
AP_VNAV	Toggles the VNAV vertical mode. To be called when pressing the VNAV button. Works only with the PMS50 WTT mode
AP_LVL	Toggles the wing leveler hold mode.
AP_CGA	Toggles the Go Around mode. The Go Around mode is a combined roll and pitch mode where the lateral mode is maintained to a zero bank angle and vertical mode is set to a VS value defined in the parameters. The Go Around mode can be used for takeoff, for missed approach and for climbing. If there is an automatic arming to a selected altitude, this one is activated in Go Around mode.
AP_CLVL	Toggles the combined LVL mode. The combined LVL mode maintains a zero bank angle and a zero vertical speed. It can be used as an alternative to the ALT mode in case of turbulences. You can find this mode in the Garmin 605 autopilot.
AP_ARM	Toggles the arm mode for VS or FLC. To be called when pressing the ARM button. The Arm management is an option.
AP_ARM_PRESEL	Toggles the arm mode for VS or FLC with a given selected altitude. The selected altitude must be set in the L:PMS50_APGA_ALTITUDE_PRESEL_VALUE variable (in feet) before sending the event. To be called when pressing the ARM button and if you aircraft doesn't provide a knob for altitude selection (ex mechanical selector). The Arm management is an option.

Event	Description
AP_UP	Nose up for VS, FLC and PITCH vertical modes. To be called when pressing the UP button. In VS mode this increases the VS by 100 feet/mn. In FLC mode this decreases the speed by 1 knot. In PITCH mode, the current pitch is decreased by 0.2 degrees.
AP_DOWN	Nose down for VS, FLC and PITCH vertical modes. To be called when pressing the DN button. In VS mode this decreases the VS by 100 feet/mn. In FLC mode this increases the speed by 1 knot. In PITCH mode, the current pitch is increased by 0.2 degrees.
SEL_ALT_SMALL_INC	Increments the selected altitude by a small value (10 or 100 depending of the precision parameter). To be called when rotating the inner altitude selection knob.
SEL_ALT_LARGE_INC	Increments the selected altitude by a large value (100 or 1000 depending of the precision parameter). To be called when rotating the outer altitude selection knob.
SEL_ALT_VERY_LARGE_INC	Increments the selected altitude by a very large value (1000 or 10000 depending of the precision parameter).
SEL_ALT_SMALL_DEC	Decrements the selected altitude by a small value (10 or 100 depending of the precision parameter). To be called when rotating the inner altitude selection knob.
SEL_ALT_LARGE_DEC	Decrements the selected altitude by a large value (100 or 1000 depending of the precision parameter). To be called when rotating the outer altitude selection knob.
SEL_ALT_VERY_LARGE_DEC	Decrements the selected altitude by a very large value (1000 or 10000 depending of the precision parameter).
SEL_VS_SMALL_INC	Increments the vertical speed by 100 feet/mn. To be called when rotating the inner VS select knob. Don't use this event for the UP button. Use AP_UP instead.
SEL_VS_SMALL_DEC	Decrements the vertical speed by 100 feet/mn. To be called when rotating the inner VS select knob. Don't use this event for the DN button. Use AP_DOWN instead.
SEL_VS_LARGE_INC	Increments the vertical speed by 1000 feet/mn. To be called when rotating the outer VS select knob. Don't use this event for the UP button. Use AP_UP instead.

Event	Description
SEL_VS_LARGE_DEC	Decrements the vertical speed by 1000 feet/mn. To be called when rotating the outer VS select knob. Don't use this event for the DN button. Use AP_DOWN instead.

Status

The autopilot status is returned by the APGA in L variables for display purpose.

The L variables has the form "L:PMS50_APGA_VariableName" where VariableName is the name given below.

Example: "L:PMS50_APGA_HDG_BUTTON_STATE" returns the state of the HDG button.

Variable	Description	Type
RUNNING	The APGA module is installed an is running.	boolean
AP	Autopilot master state.	boolean
FD	Flight director state.	boolean
YD	Yaw damper state.	boolean
AP_BUTTON_STATE	Autopilot button state. May be used to display the button active state.	boolean
FD_BUTTON_STATE	Flight director button state. May be used to display the button active state.	boolean
YD_BUTTON_STATE	Yaw damper button state. May be used to display the button active state.	boolean
HDG_BUTTON_STATE	HDG button state. May be used to display the button active state.	boolean
NAV_BUTTON_STATE	NAV button state. May be used to display the button active state. Can be also used for displaying a global NAV mode annunciator.	boolean
APR_BUTTON_STATE	APR button state. May be used to display the button active state. Can be also used for displaying a global APR mode annunciator.	boolean

Variable	Description	Type
REV_BUTTON_STATE	REV (BC) button state. May be used to display the button active state.	boolean
ALT_BUTTON_STATE	ALT button state. May be used to display the button active state.	boolean
VS_BUTTON_STATE	VS button state. May be used to display the button active state.	boolean
FLC_BUTTON_STATE	FLC button state. May be used to display the button active state.	boolean
VNAV_BUTTON_STATE	VNAV button state. May be used to display the button active state.	boolean
GA_BUTTON_STATE	GA (Go Around) button state. May be used to display the button active state.	boolean
LVL_BUTTON_STATE	LVL (Combined LVL) button state. May be used to display the button active state.	boolean
ARM_BUTTON_STATE	ARM button state. May be used to display the button active state.	boolean
CAN_ARM	Tells if the ARM button can be activated or not.	boolean
UP_DOWN_BUTTON_VALID	Tells if the UP and DN buttons can be activated or not.	boolean
ACTIVE_LATERAL_MODE	The current engaged lateral mode. See the list below.	number
ACTIVE_VERTICAL_MODE	The current engaged vertical mode. See the list below.	number
ARMED_LATERAL_MODE	The current armed lateral mode. See the list below.	number
ARMED_VERTICAL_MODE	The current armed vertical mode. See the list below.	number
ARMED_ALTITUDE_MODE	The current armed vertical mode for altitude. May be used instead of the ARMED_VERTICAL_MODE variable if your GPS has separate status for the altitude and approach armed modes. Can be 0(none) or 3(ALT).	number

Variable	Description	Type
ARMED_APPROACH_MODE	The current armed vertical mode for approach. May be used instead of the ARMED_VERTICAL_MODE variable if your GPS has separate status for the altitude and approach armed modes. Can be 0(none) or 4(GS) or 5(GP).	number
SELECTED_ALTITUDE_PHASE	The selected altitude phase. See the list below.	number
SELECTED_ALTITUDE	The selected altitude.	feet
CAPTURED_ALTITUDE	The captured altitude in ALT mode.	feet
CAPTURED_ALTITUDE_VALID	Tells if the captured altitude is valid for display.	boolean
SELECTED_VS	The selected vertical speed.	feet/minute
SELECTED_AIRSPEED	The selected airspeed.	knots
NEED_TRIM_INDICATOR	The TRIM indicator.	boolean
NEED_TRIM_UP_INDICATOR	The TRIM UP arrow indicator.	boolean
NEED_TRIM_DOWN_INDICATOR	The TRIM DOWN arrow indicator.	boolean
NEED_TRIM_VALUE	The need trim value.	feet per minute

Lateral and Vertical modes

You can use the lateral and vertical modes (current or armed) for updating the autopilot indicator panel.

Lateral modes

Active lateral mode

The current lateral mode is returned by the "L:PMS50_APGA_ACTIVE_LATERAL_MODE" variable.

Mode	Description	Value
None	No mode.	0
ROLL	Roll mode.	1

Mode	Description	Value
HDG	HDG mode.	2
GPS	GPS mode. This is the NAV mode driven by the GPS.	3
VOR	VOR mode. This is the NAV mode driven by VLOC with a VOR frequency.	4
LOC	LOC mode. This is the NAV mode driven by VLOC with a LOCALIZER frequency.	5
BC	Back course mode.	6
APRGPS	Approach GPS mode. This is the Approach mode driven by the GPS (RNAV approach).	7
APRVOR	Approach VOR mode. This is the Approach mode driven by VLOC (VOR approach).	8
APRLOC	Approach LOC mode. This is the Approach mode driven by VLOC (ILS approach).	9
LVL	Wing leveler mode.	10
CGA	Combined Go around mode. The Go Around mode is a combined roll and pitch mode where the lateral mode is maintained to a zero bank angle and vertical mode is set to a VS value defined in the parameters. The Go Around mode can be used for takeoff, for missed approach and for climbing. If there is an automatic arming to a selected altitude, this one is activated in Go Around mode.	11
CLVL	Combined Level mode. The combined LVL mode maintains a zero bank angle and a zero vertical speed. It can be used as an alternative to the ALT mode in case of turbulences. You can find this mode in the Garmin 605 autopilot.	12

> Note: We have separate values for GPS, VOR and LOC. If you have a single NAV indicator, this will be a logical or between GPS, VOR and LOC.
> The same for approach modes and the APR indicator.

Active vertical mode

The current vertical mode is returned by the "L:PMS50_APGA_ACTIVE_VERTICAL_MODE" variable.

Mode	Description	Value
None	No mode.	0
PITCH	Pitch mode.	1

Mode	Description	Value
VS	VS mode.	2
ALT	ALT mode.	3
GS	GS mode. Vertical mode for an ILS approach.	4
GP	GP mode. Vertical mode for an RNAV approach.	5
FLC	Flight level change mode.	6
VNAV	VNAV Mode. Works only with PMS50 WTT mode.	7
CGA	Combined Go around mode. The Go Around mode is a combined roll and pitch mode where the lateral mode is maintained to a zero bank angle and vertical mode is set to a VS value defined in the parameters. The Go Around mode can be used for takeoff, for missed approach and for climbing. If there is an automatic arming to a selected altitude, this one is activated in Go Around mode.	8
CLVL	Combined Level mode. The combined LVL mode maintains a zero bank angle and a zero vertical speed. It can be used as an alternative to the ALT mode in case of turbulences. You can find this mode in the Garmin 605 autopilot.	9

Armed lateral mode

The armed lateral mode is returned by the "L:PMS50_APGA_ARMED_LATERAL_MODE" variable.

Mode	Description	Value
None	No mode.	0
GPS	GPS mode. This is the NAV mode driven by the GPS.	3
VOR	VOR mode. This is the NAV mode driven by VLOC with a VOR frequency.	4
LOC	LOC mode. This is the NAV mode driven by VLOC with a LOCALIZER frequency.	5
BC	Approach back course mode.	6
APRGPS	Approach GPS mode. This is the Approach mode driven by the GPS (RNAV approach).	7
APRVOR	Approach VOR mode. This is the Approach mode driven by VLOC (VOR approach).	8
APRLOC	Approach LOC mode. This is the Approach mode driven by VLOC (ILS approach).	9

> Note: We have separate values for GPS, VOR and LOC. If you have a single NAV indicator, this will be a logical or between GPS, VOR and LOC.
> The same for approach modes and the APR indicator.

Armed vertical mode

The armed vertical mode is returned by the "L:PMS50_APGA_ARMED_VERTICAL_MODE" variable.

Mode	Description	Value
None	No mode.	0
ALT	ALT mode.	3
GS	GS mode. Vertical mode for an ILS approach.	4
GP	GP mode. Vertical mode for an RNAV final approach.	5
VNAV	VNAV Mode. Works only with PMS50 WTT mode.	7

Selected altitude phases (ALTS)

The current selected altitude phase is returned by the "L:PMS50_APGA_SELECTED_ALTITUDE_PHASE" variable.

These phases switch from one to the other in a consecutive order when a target altitude is armed. The thresholds are defined as parameters. See above.

Phase	Description	Value
None	No phase. The ALTS is not armed	0
Engaged	The ALTS is armed and the altitude to the target is greater than the alert threshold.	1
Alert	The ALTS is armed and the altitude to the target is between the alert and near thresholds.	2
Near	The ALTS is armed and the altitude to the target is less than the near threshold.	3
Ready	The vertical mode is ALT and the altitude to the target is greater than the reached threshold.	4
Reached	The vertical mode is ALT and the altitude to the target is less than the reached threshold. The Phase returns to None after a time out.	5

Annunciator panel example

Indicator	Variable and value for activating the annunciator
FLT DIR	L:PMS50_APGA_FD = true
HDG SEL	L:PMS50_APGA_ACTIVE_LATERAL_MODE = 2
ALT HOLD	L:PMS50_APGA_ACTIVE_VERTICAL_MODE = 3
AP	L:PMS50_APGA_AP = true
NAV	L:PMS50_APGA_NAV_BUTTON_STATE = true
ARM	L:PMS50_APGA_ARMED_LATERAL_MODE != 0
ALT ARM	L:PMS50_APGA_SELECTED_ALTITUDE_PHASE = 1, 2 or 3
GO AROUND	L:PMS50_APGA_ACTIVE_VERTICAL_MODE = 8
APPR	L:PMS50_APGA_APR_BUTTON_STATE = true
CPLD	L:PMS50_APGA_ARMED_LATERAL_MODE = 0 and (L:PMS50_APGA_NAV_BUTTON_STATE = true or (L:PMS50_APGA_APR_BUTTON_STATE = true)
GS CPLD	L:PMS50_APGA_ACTIVE_VERTICAL_MODE = 4
BACK CRS	L:PMS50_APGA_ACTIVE_LATERAL_MODE = 6
TRIM WARN	L:PMS50_APGA_NEED_TRIM_INDICATOR = true

Nota: for GS CPLD you can also test "L:PMS50_APGA_ACTIVE_VERTICAL_MODE = 4 or L:PMS50_APGA_ACTIVE_VERTICAL_MODE = 5" if you want to activate the indicator for both GS and GP(RNAV) glide slope.

APGA example

You will find some autopilot APGA gauge examples in the Integration/Files/autopilot folder of the PMS50 main package.

Copyright