How I retrofitted a digital gauge cluster into my Cadillac ATS-V

Varjitt Jeeva - Software Engineer, Car Enthusiast

Overview

1. Introduction of me and the project

2. What points made this project possible

3. How the hell I figured it out

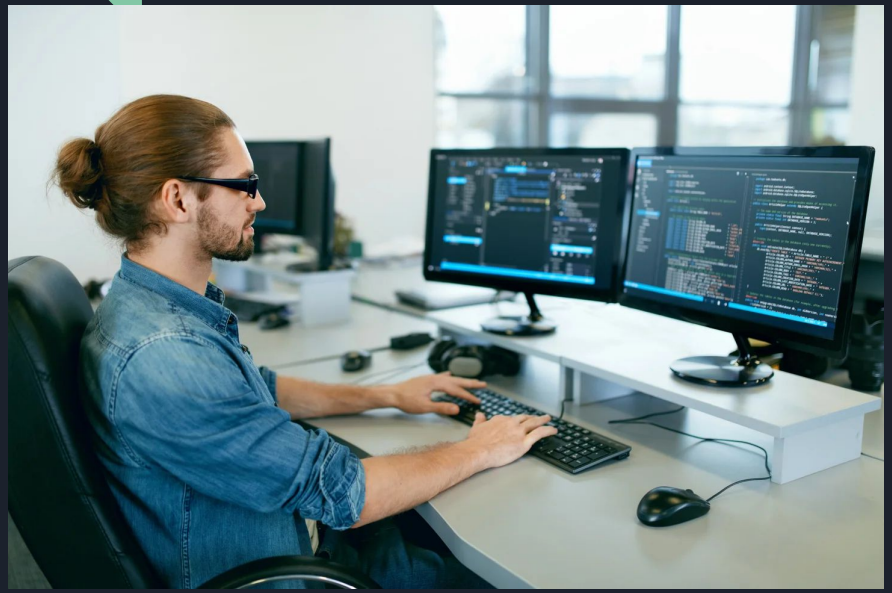
3. High level process

4. Results!

5. Future of GM Programming and Open Source!

Introduction to me and the project

Who Am I?



What came with my car vs what came in other Cadillacs in the same year (not for ATS-V)

GM, why do you make people suffer like this

You can pay people $350 to do the programming

2 people sell this service, one guy sells cool GM programming courses, some people sell sketchy tools and services.

Clearly the needed information is available online, just very hard/obscure to find and understand. That seems like a great challenge to me!

What points made this project possible

What points made this project possible

1. Lots of Cadillac Models from the same year of my car (2017) had **the same 12.3 inch digital cluster** with different views (all controlled with configuration flags)!

2. All those 12.3 inch clusters, **along with my car's cluster,** had the **same connecting plug!**

3. The view I wanted (for the Cadillac CTS V-Sport) was almost the same engine so the gauges make sense with my car (3.6 Twin Turbo V6) with turbo boost gauges showing vacuum, etc.

4. The total set of configuration flags used by my car's cluster and ALL variations of the 12.3 inch digital cluster **is the same! So the cluster supported my car fully without ever being shipped with it!**

The Grand Story of how I figured this out

In the next slides, I also marked off how much money I saved by figuring it out myself.

If I can do it, you can too!

Thought Number 1 - Existing Programming Services

Programming Advertisement:

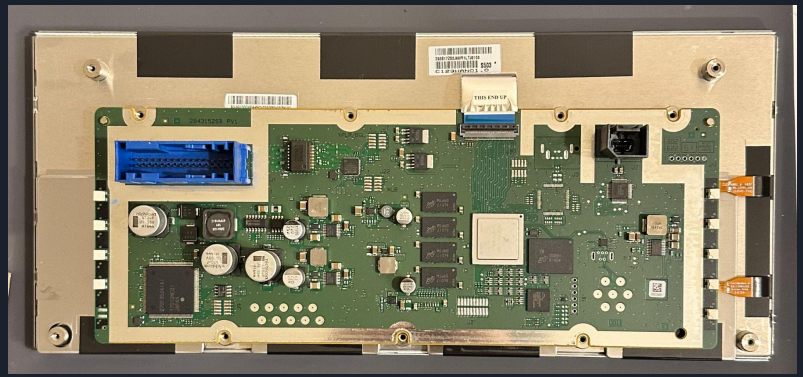
- **Give us any Cadillac 12.3 inch digital cluster (same year as your car)**
- **We'll program it to show the V cluster for your car**
- **We'll also ensure it has the correct redline 8000rpm, 6MT manual active gear, AND boost gauges with negative vacuum!**

Thought Number 2 - Can I retrofit from another car?

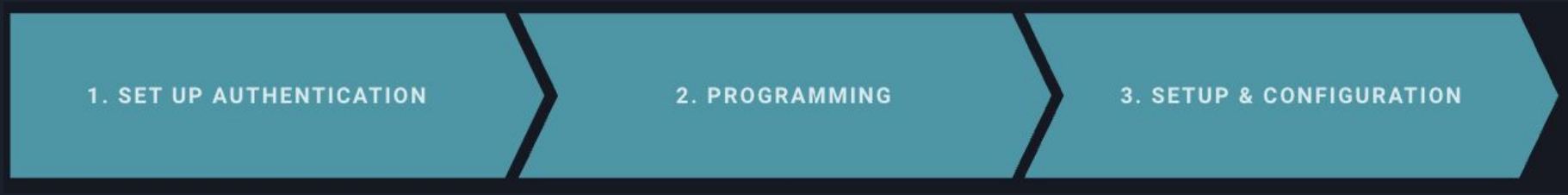
- Is it totally custom somehow? **Unlikely**
- Does that cluster view exist already in a close matching car? **YEAH**
  - My car: Cadillac ATS-V (3.6L TTV6)
  - Car with 12.3 V cluster and similar engine: **Cadillac CTS V-Sport (3.6L TTV6)**

Thought Number 3 - How can I program my cluster to look like the donor car?

- Board had a clear CPU on it, **if I remember correctly, was Texas Instruments ARM**
  - Must have been Linux-based
  - **Therefore had files and animations playing for everything that just had to be on the board!**
- **Next Question 1** - can this connect to my car?
- **Next Question 2** - are the right files on there? Do I need to patch them?
- **Next Question 3** - how do I configure the cluster to work with my car (features, flags, etc)?



Three Steps to Programming GM CAN Bus Modules



1. **Authentication -** Module needs to Auth with the rest of the CAN Bus System.

2. **Programming** - .bin files contain actual code/data/etc to be loaded into modules. **These contain data/code for all variations/types/behaviours of the module.**

3. Configuration - Set flags on the module so it behaves according to your car and its features.

1. SET UP AUTHENTICATION

Thought Number 4 - Can I just plug and play this cluster?

- Nope - GM CAN Bus Modules all need to **authenticate otherwise the module itself won't work properly or at all.**
- How does that work?
  - EEPROM on Cluster stores VIN and "Security PIN" (generated from VIN) to authenticate with the CAN Bus System.
- Cheap & Easy Solution - CH341a EEPROM Programmer
  - Chip is I2C 24C16

Doing this without EEPROM editing = $100
$100/$950

AsProgrammer

IC Options Hardware Scripts Language

SPI I2C Microwire(16bit)

IC: W25Q128BV

Size: 16777216

Page: 256

SPI commands: 25xx

Scripts

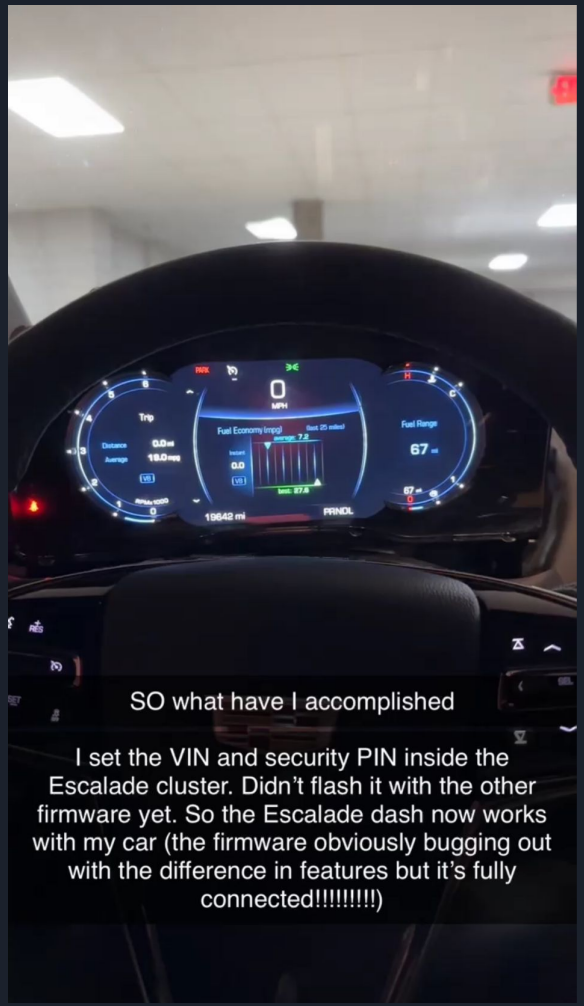
Start address: 0x 000000

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	0123456789ABCDEF
0x00000660	02	80	04	40	0C	00	81	04	03	80	04	40	04	00	84	30	...@.....@...0
0x00000670	09	00	80	10	00	00	02	24	01	00	02	24	02	10	04	40	.......$...$...@
0x00000680	00	18	91	7C	40	3D	92	7C	02	00	04	40	00	38	95	7C	...|@=.|...@.8.|
0x00000690	05	00	00	10	80	1A	94	7C	00	00	11	24	00	00	12	24	.......|...$...$
0x000006A0	00	00	14	24	00	00	15	24	00	78	19	40	00	7A	24	7F	...$...$.x.@.z$▯
0x000006B0	99	01	07	24	07	00	E4	10	9A	01	07	24	05	00	E4	10	...$.......$....
0x000006C0	00	00	00	00	00	00	03	24	00	00	10	24	11	00	00	10	.......$...$....
0x000006D0	00	00	13	24	01	00	03	24	BF	BF	05	3C	00	80	A5	34	...$...$...<...4
0x000006E0	08	00	A4	8C	44	FF	05	7C	04	70	04	7C	02	00	A4	10	....D..|.p.|....
0x000006F0	00	00	00	00	3F	00	00	70	BF	BF	16	3C	00	80	D6	36	....?..p...<...6
0x00000700	28	20	D0	8E	00	00	C4	8E	00	BE	05	3C	0C	00	A4	8C	(.....<....
0x00000710	40	04	93	7C	08	00	E0	03	00	00	00	00	40	00	0B	3C	@..|.....@..<
0x00000720	04	04	6B	35	00	60	8B	40	01	80	0A	40	C0	00	4B	7D	..k5.`.@...@..K}
0x00000730	1F	00	60	11	07	00	0B	24	00	98	8B	40	00	98	0A	40	..`....$...@...@
0x00000740	1B	00	41	05	00	90	80	40	01	98	8B	40	01	98	0A	40	..A....@...@...@
0x00000750	17	00	41	05	01	90	80	40	02	98	8B	40	02	98	0A	40	..A....@...@...@
0x00000760	13	00	41	05	02	90	80	40	03	98	8B	40	03	98	0A	40	..A....@...@...@
0x00000770	0F	00	41	05	03	90	80	40	04	98	8B	40	04	98	0A	40	..A....@...@...@

Size: 16777216 W25Q128BV

Reading memory...
Done
Execution time: 00:02:24
CRC32 = 0x0F342791

	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	Decoded Text
00000000	31	47	59	53	34	48	4B	4A	32	48	52	32	34	32	31	33	1GYS4HKJ2HR24213
00000010	31	50	43	41	52	53	54	4E	23	36	31	20	17	02	15	00	1PCARSTN#61 ....
00000020	00	FF	FF	FF	FF	FF	00	CE	4A	73	00	00	00	00	00	00	........Js......
00000030	00	00	00	00	00	00	00	00	00	00	FF	FF	FF	FF	05	04	....
00000040	7A	13	41	4E	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	00	00	z.AN............
00000050	00	00	00	00	00	00	00	00	FF	FF	FF	FF	FF	FF	FF	FF	....
00000060	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	....
00000070	FF	FF	FF	FF	FF	FF	FF	FF	00	00	39	2C	00	0E	0F	82	..........9,....
00000080	00	CE	44	C0	00	00	00	00	00	03	37	F8	00	CB	8B	58	..D.......7....X
00000090	00	15	11	25	00	CE	44	BC	00	E8	EE	53	00	00	00	00	...%..D....S....
000000A0	00	CB	0D	1A	00	CA	96	36	00	00	38	37	39	31	01	00	........6..8791..
000000B0	02	04	B5	D1	AD	01	00	00	00	00	00	C8	00	39	0F	44	..............9.D
000000C0	CC	00	00	00	00	00	00	00	00	00	0A	28	78	00	00	3C	..........(x...<
000000D0	00	00	00	00	00	00	E6	B7	2D	0E	00	45	BF	57	12	FF	..........-..E.W.

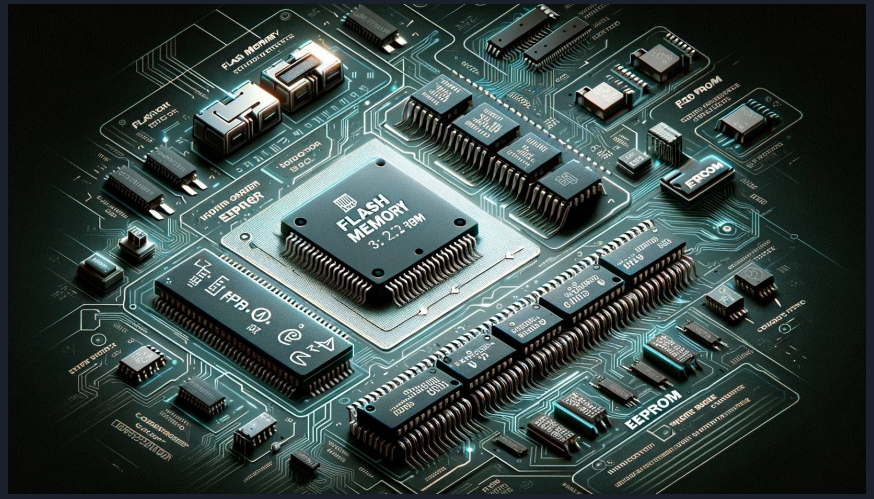


How I knew the cluster authenticated: **my mileage showed up!**

This comes from the BCM (Body Control Module) of the car. The VIN and PIN on the cluster must match the BCM, then the cluster can fetch the mileage from there.

If not matching, **the mileage will show as dashes!**

Thought Number 5, Part 1 - confirming the right files!



How module programming works with GM/ACDelco

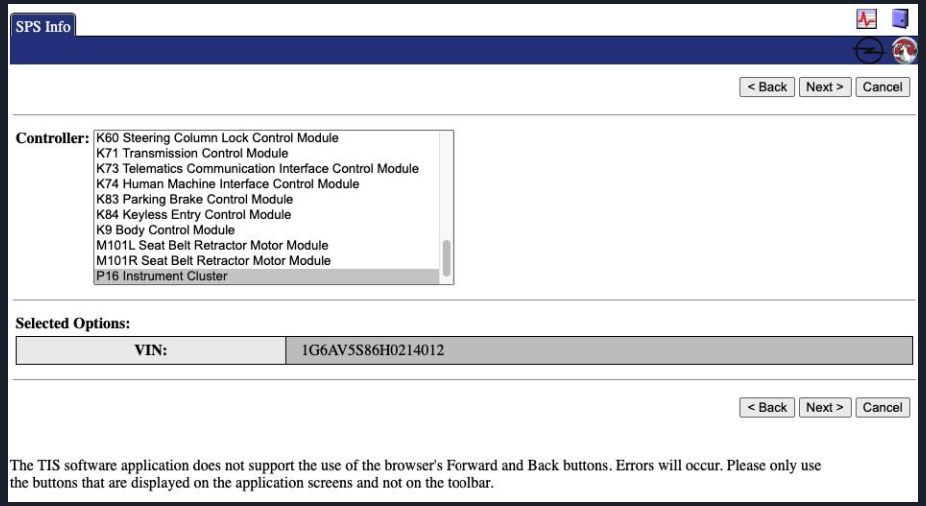
- Dealer software called SPS2 or ACDelco TDS, available to car owners for $45 per VIN 2 years!
- OBD Port Cable - “MDI” or “MDI2” Cable. AliExpress $186

What Programming Does:

- Load .bin files contain actual code/data/etc to be loaded into modules. **These contain data/code for all variations/types/behaviours of the module**

2. PROGRAMMING

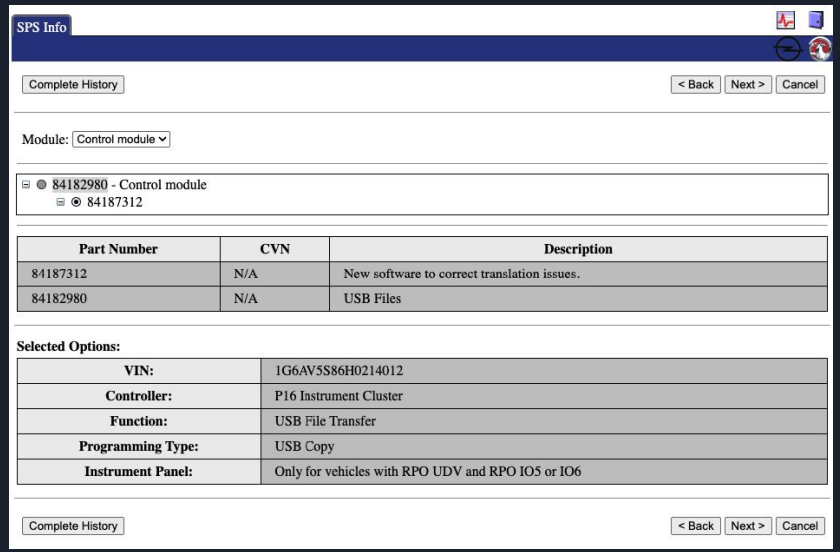
Thought Number 5, Part 2 - confirming the right files!



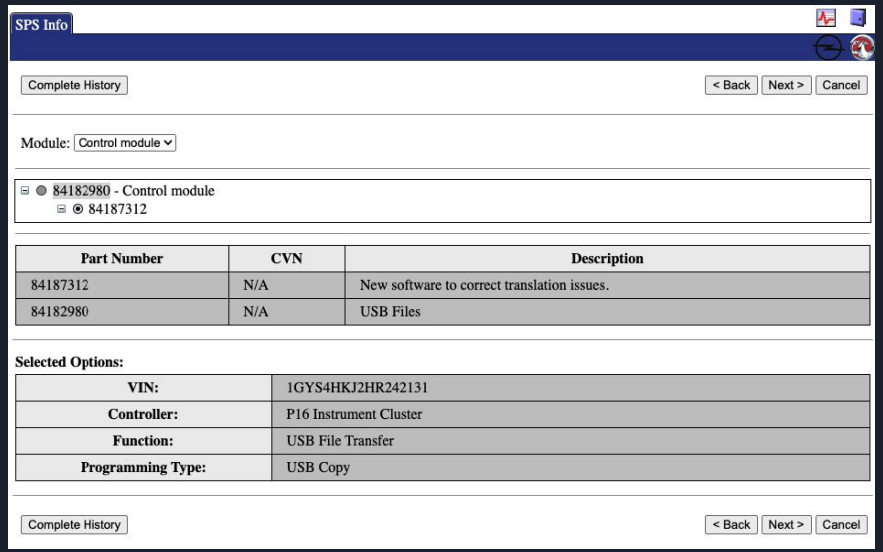
- Does my cluster have the right BIN file? Is it different among the cars? **Do the right animation files exist in my cluster?**
- https://tis2web.service.opel.com/tis2web/
  - Opel SPS Web UI (turns out GM programming is used also in Opel/Vauxhall)
  - This site gives **calibration numbers** for .bin files used in programming without loading full SPS2.
  - Site is used to confirm what BIN files should be used in programming, if updates exist, etc (calibration numbers increment for updates)

Comparing my Escalade Cluster to Donor CTS V-Sport

Note - Clusters are programmed via Function “USB Copy” due to big file sizes (animations, images, etc)! Programming via regular Programming (MDI Cable) seems to be for lower level stuff like Bootloaders.



CTS V-Sport



Escalade

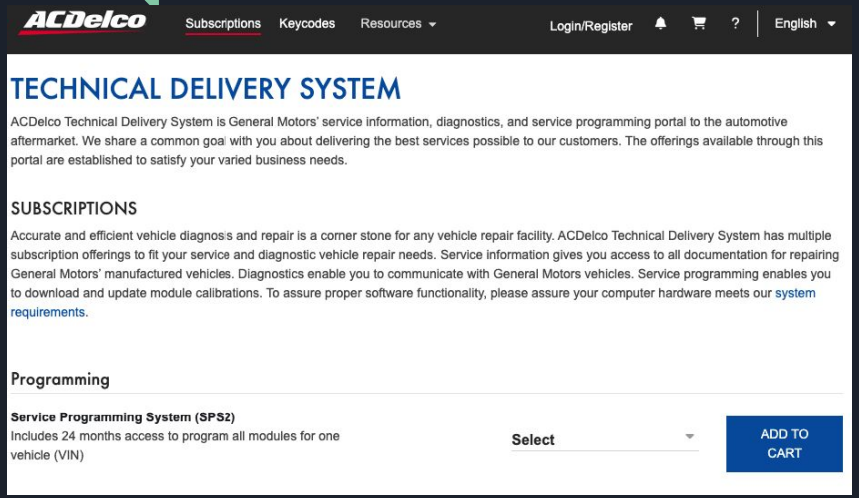
IT’S THE SAME 84187312 BIN!!!

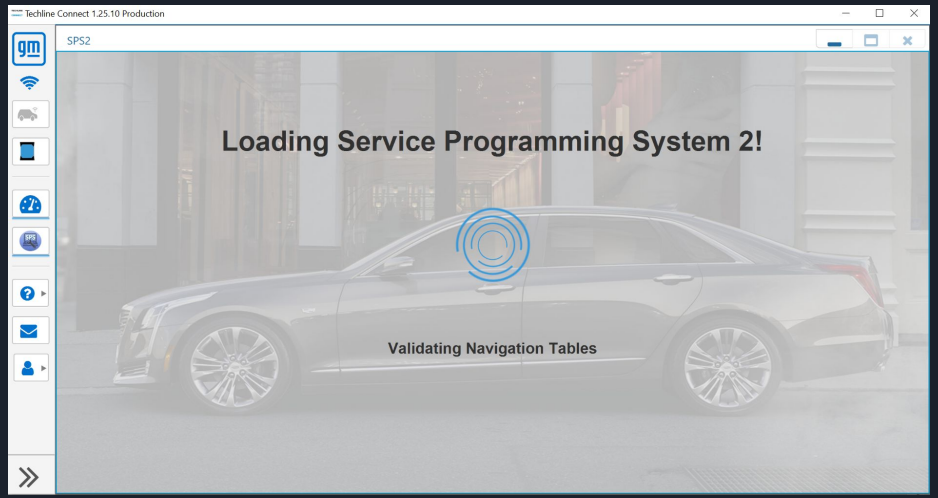
3. SETUP & CONFIGURATION

Thought Number 6 - How can I configure the flags to show the right V gauges?

- Setup & Configuration set flags in a module for it to enable/disable features, etc.
  - We just do this to enable the right V gauges!
- How do they get set?
  - Next Slide!

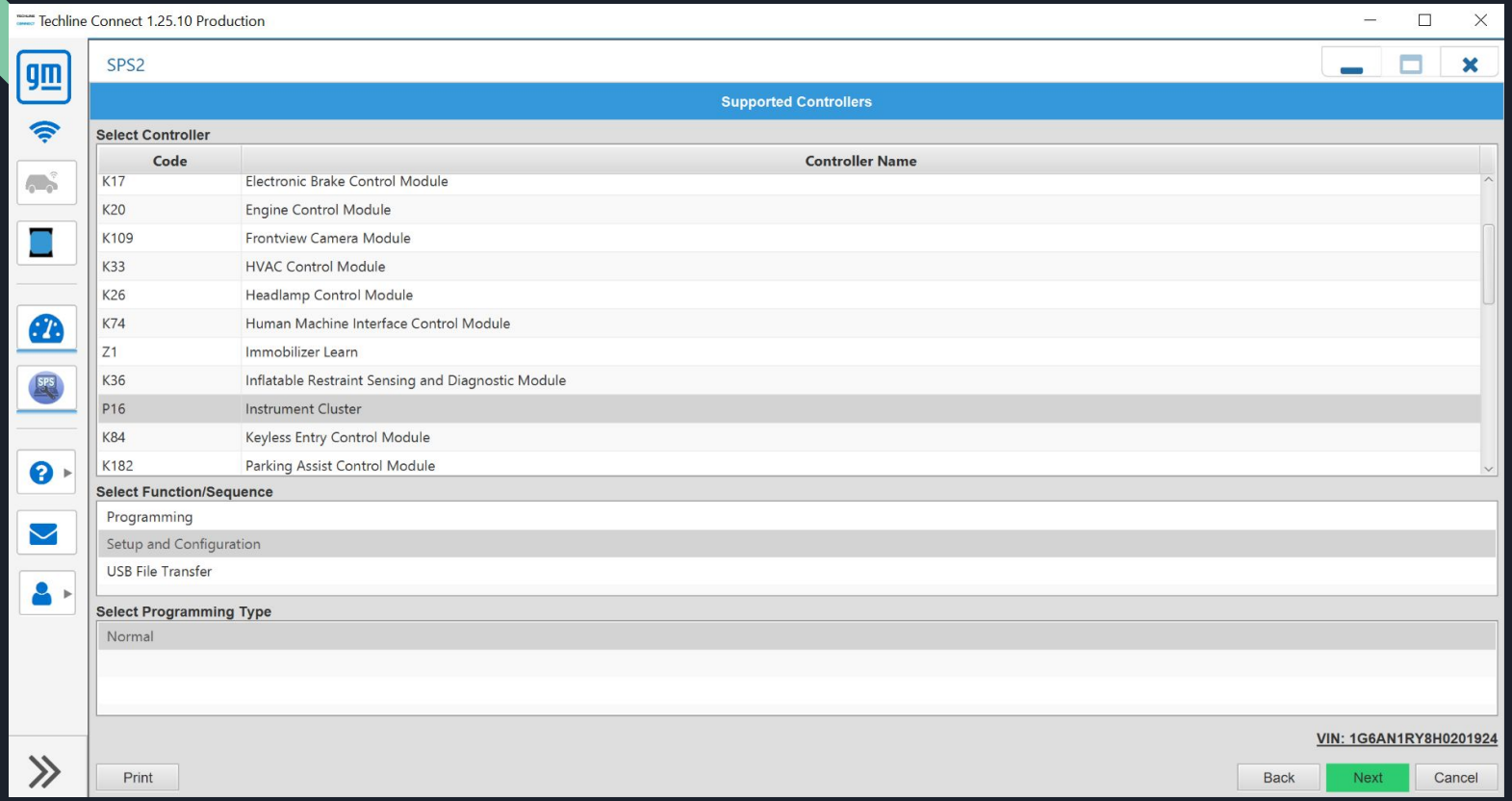
Setup & Configuration with SPS2



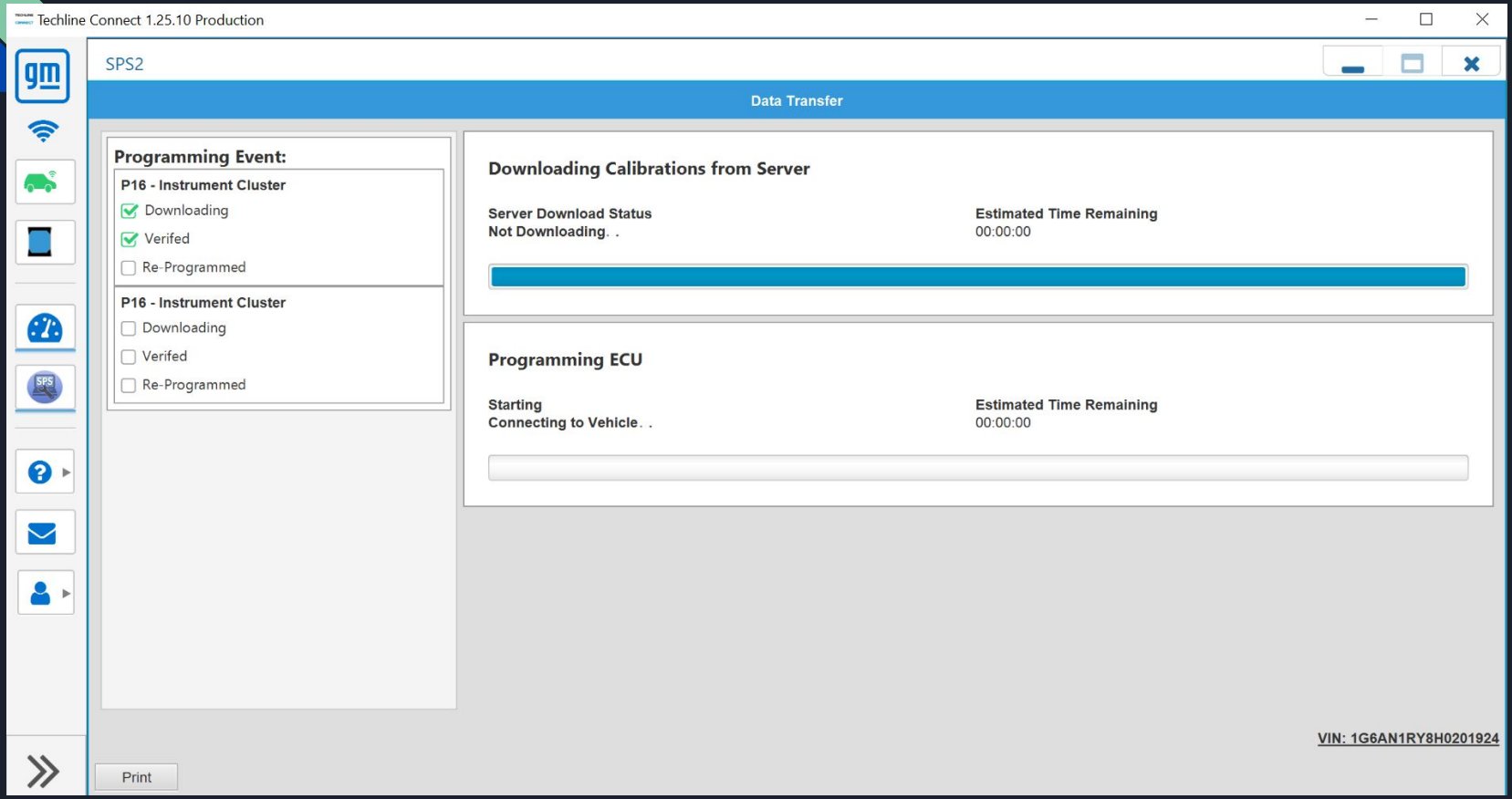


https://www.acdelcotds.com/

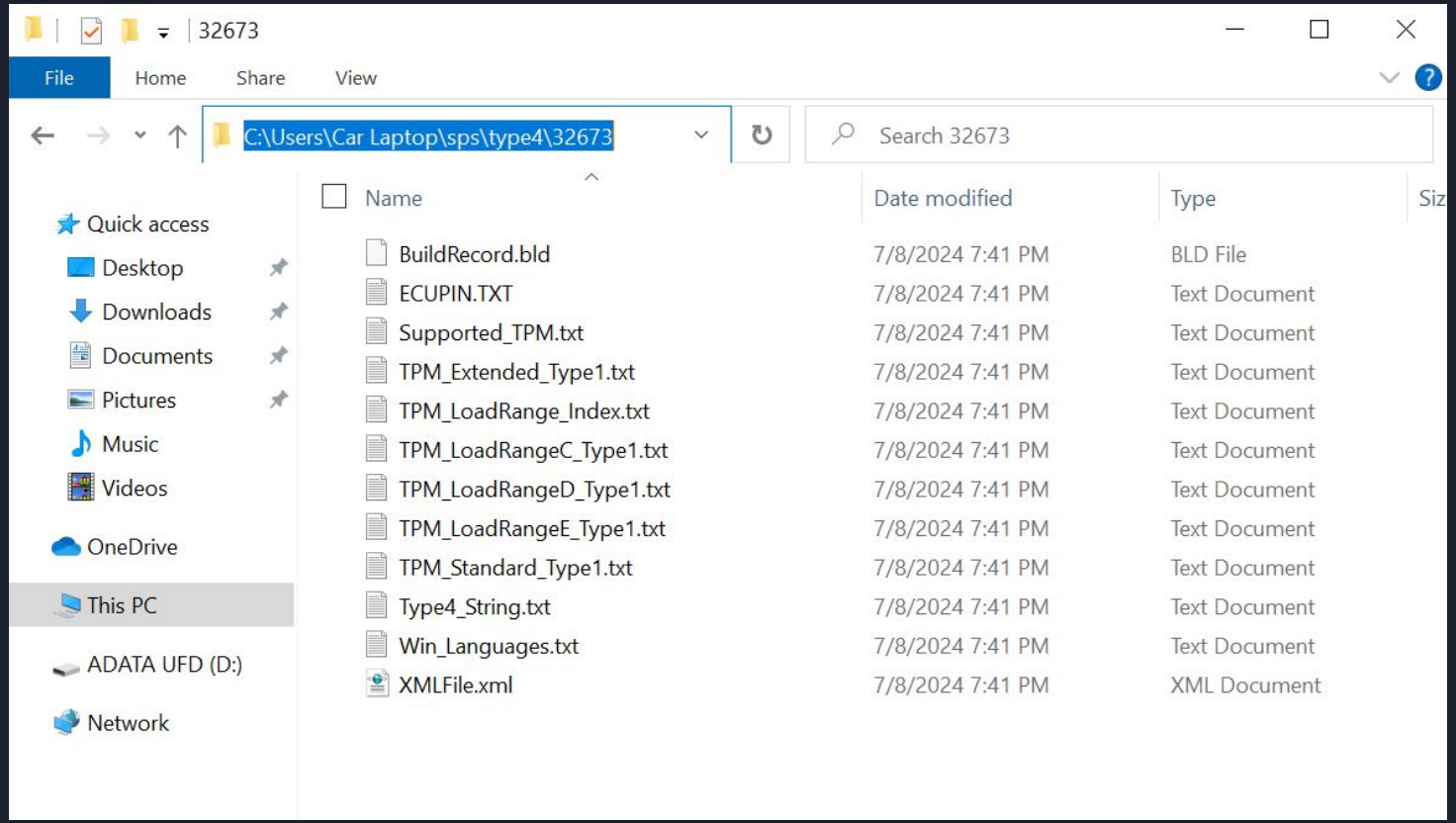
Setup & Configuration with SPS2



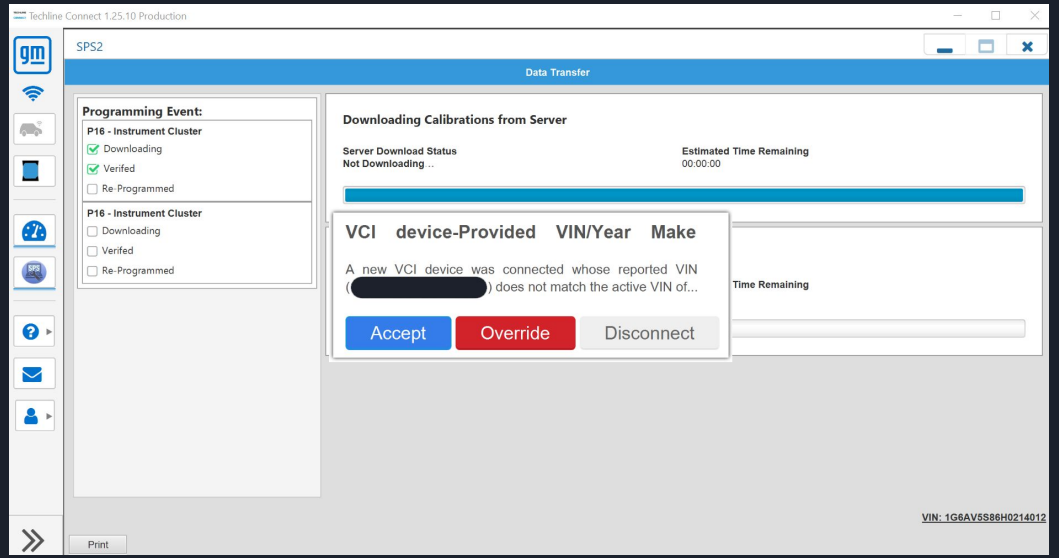
Setup & Configuration with SPS2



Setup & Configuration with SPS2



Bonus Slide 1 - How to Grab BINs and Config Files from SPS2 for ANY car!



- Have an SPS2 Subscription for your Donor VIN **(aka the car you WANT files from)**
  - ACDelco TDS just adds VIN to an allow list on your account
- Hook up your car into SPS2
- **When connecting, click the Donor VIN (not your car)!**
  - SPS2 will pop up saying VIN is mismatched, **click Accept and that's it!**
- You can now "program" for your target modules from the donor VIN, then during the program process, let SPS2 download the files, then **unplug the cable as soon as it tries to program your car!**
  - Check cache directories, files are now there!

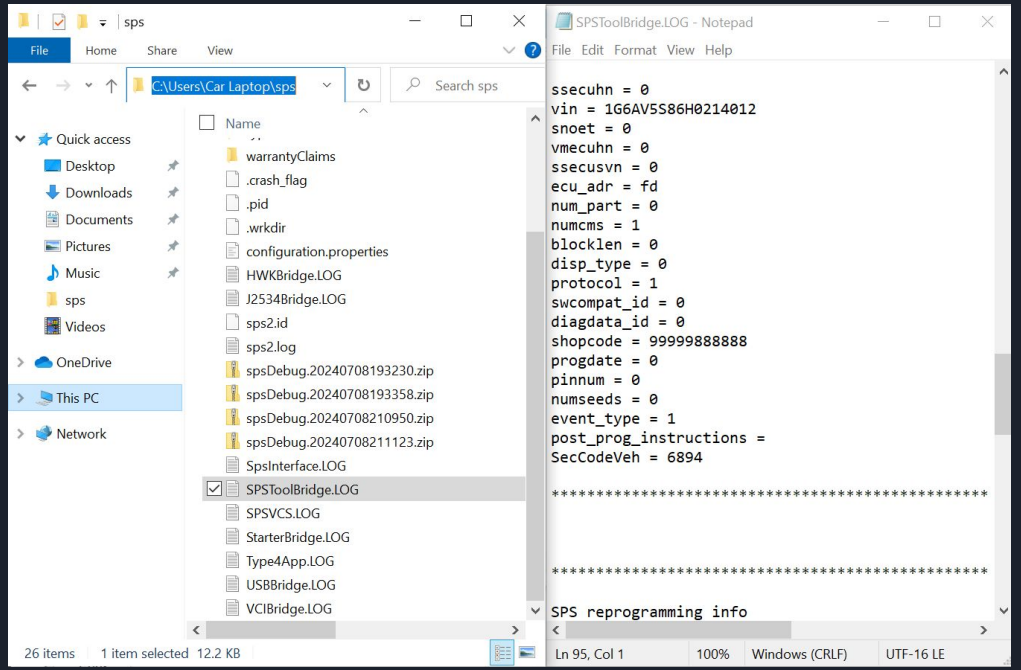
NO NEED TO BUY SKETCHY MDI CAR EMULATORS, THEY ALSO GET YOU BANNED!

Emulator = $450
$550/$950

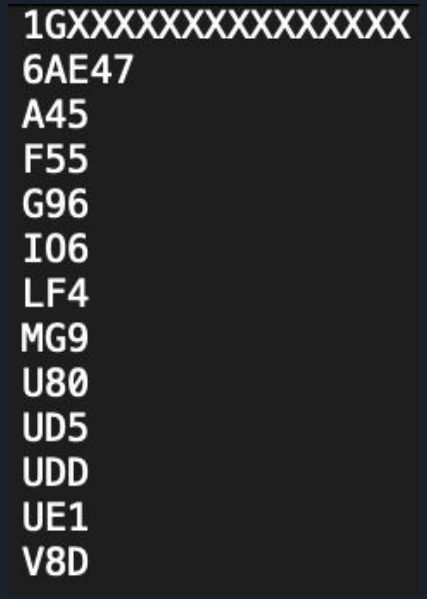
VIN to PIN Service = $50
$600/$950

Bonus Slide 2 - How to Grab the Security PIN for a VIN

- Have an SPS2 Subscription for the VIN you want the PIN from
- Hook up your car into SPS2
- **When connecting, enter the Target VIN (not necessarily your car)!**
  - If not the same, SPS2 will pop up saying VIN is mismatched, **click Accept!**
- Run a Setup & Configuration with any module (let's say P16 Instrument Cluster)
- Let it run. If the entered VIN doesn't match your car's VIN, the configuration won't work. SPS2 will fail to auth with the module since VIN and PIN don't match.
- Check logs, **Target VIN and Security PIN are there!**



Example RPO Code List



BuildRecord.bld from my car.

ACDelco

RPO Codes and Descriptions

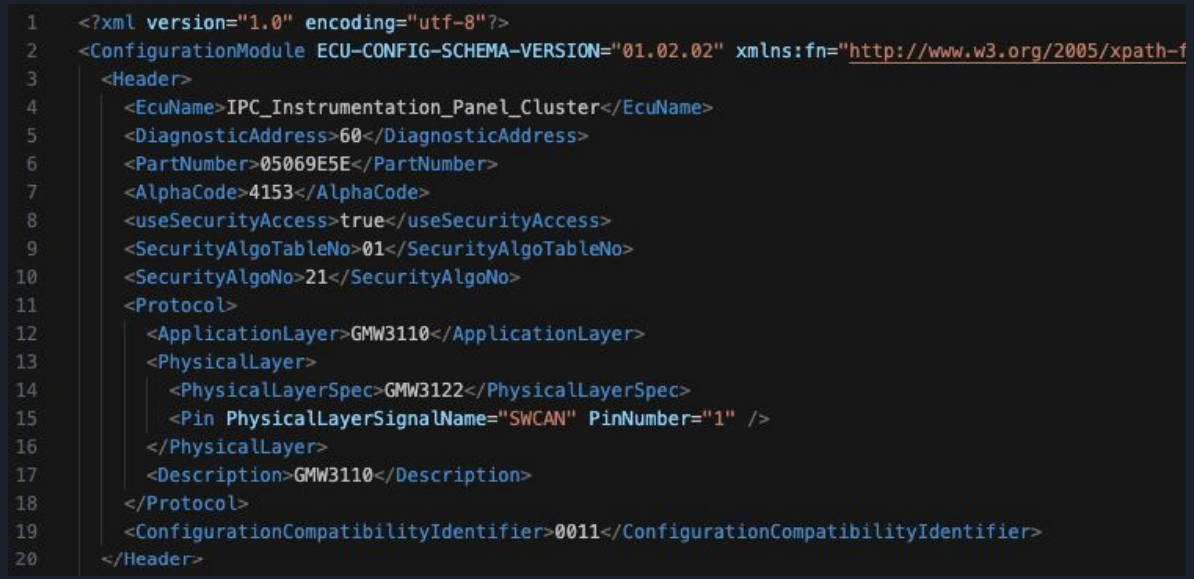
Code	Description
AAB	MEMORY DRIVER CONVENIENCE PACKAGE
AAC	SHIPPED LOOSE PARTS FOR SHIPPING INSTRUCTION
AAD	WINDOW BODY, LEFT SIDE
AAE	INTERIOR TRIM DELETE
AAF	2009 OEM ENGINE PHYSICAL ID AAF & PRODUCTION NUMBER 12603557
AAG	MEMORY PASS CONVENIENCE PACKAGE
AAH	RESTRAINT KNEE, INFLATABLE, LH
AAI	CONTROL A/TRANS, MODE, ECONOMY/POWER
AAK	LOCK CONTROL, ENTRY DOOR, ELEC, KEY ACTIVATED
AAL	RESTRAINT KNEE, BOLSTER, LH/RH
AAM	RESTRAINT,KNEE BOLSTER,DRIVER
AAO	WINDOW ABSORBING GLAZING
AAP	FLEET INCENTIVE US INVESTIGATION SERVICES (D/W/3A/3Z - TRK STUX)
AAP	IDENTIFICATION EFFECTIVE POINT CONTROL, 2012 1/2 M.Y.
AAQ	ADJUSTER PASS ST POWER, 4 WAY
AAR	KNEE BOLSTER, FOAM TYPE
AAV	INTERIOR TRIM CONFIG - DELETE
AAW	INTERIOR TRIM CONFIG #17
AAX	INTERIOR TRIM CONFIG #18
AAY	INTERIOR TRIM CONFIG #16

Publicly Released List by ACDelco: https://rparts-sites.s3.amazonaws.com/dec399b083833699748404 2ed2af1004/design/rpoCodes.pdf

XMLFile.xml - What is it?

Seems to be a blueprint to tell the configuration software what P_XXXX flags to set in a targeted module, and to what values, based on your car's build options.

You can read it saying "Set P_XXXX" to value "Y" if Chassis Code is ZZZZZ and RPO Code PPP exists. (See next slide)



```
<?xml version="1.0" encoding="utf-8"?>
<ConfigurationModule ECU-CONFIG-SCHEMA-VERSION="01.02.02" xmlns:fn="http://www.w3.org/2005/xpath-f
  <Header>
    <EcuName>IPC_Instrumentation_Panel_Cluster</EcuName>
    <DiagnosticAddress>60</DiagnosticAddress>
    <PartNumber>05069E5E</PartNumber>
    <AlphaCode>4153</AlphaCode>
    <useSecurityAccess>true</useSecurityAccess>
    <SecurityAlgoTableNo>01</SecurityAlgoTableNo>
    <SecurityAlgoNo>21</SecurityAlgoNo>
    <Protocol>
      <ApplicationLayer>GMW3110</ApplicationLayer>
      <PhysicalLayer>
        <PhysicalLayerSpec>GMW3122</PhysicalLayerSpec>
        <Pin PhysicalLayerSignalName="SWCAN" PinNumber="1" />
      </PhysicalLayer>
      <Description>GMW3110</Description>
    </Protocol>
    <ConfigurationCompatibilityIdentifier>0011</ConfigurationCompatibilityIdentifier>
  </Header>
```

XMLFile.xml - the secret sauce!

$350 programming service
$950/$950

```
3221      <Parameter ParamDefaultValue="02">
3222        <ParameterName>P_DDH_VARIANT</ParameterName>
3223        <StartByte>7</StartByte>
3224        <StartBit>3</StartBit>
3225        <BitLength>4</BitLength>
3226        <ConfData useConfData="true">
3227          <ConfigData Value="03" AutoGenExpression="(M:6AK69 &amp;amp; UDV) | (M:6AL69 &amp;amp; !LF3 &amp;amp; UDV)">
3228            <Product>
3229              <RpoCodeRef IDREF="OCUDV" hasToBeAvailable="true" />
3230              <ModelDesignatorRef IDREF="MD6AK69" hasToBeAvailable="true" />
3231            </Product>
3232            <Product>
3233              <RpoCodeRef IDREF="OCLF3" hasToBeAvailable="false" />
3234              <RpoCodeRef IDREF="OCUDV" hasToBeAvailable="true" />
3235              <ModelDesignatorRef IDREF="MD6AL69" hasToBeAvailable="true" />
3236            </Product>
3237          </ConfigData>
3238+         <ConfigData Value="06" AutoGenExpression="(M:6AL69 &amp;amp; LF3 &amp;amp; UDV) | (M:6AE47 &amp;amp; LF4 &amp;amp; UDV)">
3239            <Product>
3240              <RpoCodeRef IDREF="OCLF3" hasToBeAvailable="true" />
3241              <RpoCodeRef IDREF="OCUDV" hasToBeAvailable="true" />
3242              <ModelDesignatorRef IDREF="MD6AL69" hasToBeAvailable="true" />
3243            </Product>
3244+           <Product>
3245+             <RpoCodeRef IDREF="OCLF4" hasToBeAvailable="false" />
3246+             <RpoCodeRef IDREF="OCUDV" hasToBeAvailable="true" />
3247+             <ModelDesignatorRef IDREF="MD6AE47" hasToBeAvailable="true" />
3248+           </Product>
3249          </ConfigData>
```

MY XML FILE AND THE TARGET CAR'S XML FILE WERE THE SAME!

You see P_XXXX flags being set different values based on RPO Codes and Chassis Codes!

I just looked for anything related to:

- 6AL69 (CTS V-Sport Chassis Code)
- UDV (RPO code for 12.3 Cluster)
  - Side note, UDD is the RPO code for my old analog cluster

Found P_DDH_VARIANT = 06 likely for V-Sport Gauges and made the left edit!

My car (ATS-V Coupe):

- Chassis: 6AE47
- Engine RPO Code: LF4

Note - I also changed my BuildRecord UDD code to UDV for 12.3 cluster support.

The Final Moments!

- Edited the files in the Cache Directory
- Triggered SPS2 P16 Instrument Cluster Configuration
  - It saw the cache directory files and ran with them!
- BOOM! —---------->

Tying all of this together!
High level process

Requirements

- Windows Laptop
- 12.3 inch cluster from a Cadillac with the same model year (XTS/CTS/CT6/Escalade/etc) $200
- GM MDI interface Cable (AliExpress $186)
- GM SPS2 Subscription $45
- CH341a EEPROM Programmer $5
- AsProgrammer software
- Tools to remove cluster

High Level Programming Steps

1. Setting up Windows Laptop
2. Run an SPS2 cluster configuration to get RPO Codes and XML
3. Back at desk, edit EEPROM on Cluster to set VIN and Security PIN
4. At your desk, edit the config files and place them in the SPS2 Cache Directory
5. Use SPS2 to set the Config Flags in the car!

Future of GM Programming and Open Source!

Future of GM Programming!

See my public Gist on Github!
https://gist.github.com/vjeeva/6a3fedc55af82ccafde59547ecff5348

I posted on the Cadillac Forums and helped a few people!
https://www.cadillacforums.com/threads/howto-program-a-12-3-inch-digital-v-cluster-for-your-cadillac-ats-v.1143115/

The fundamental understanding of this process will help anyone interested perform similar car electronics retrofitting in GM!

- EG: taking other factory options from different cars and putting them on your car

Why didn't GM just support this cluster as an in-dealer upgrade? **They had the resources to do it!**