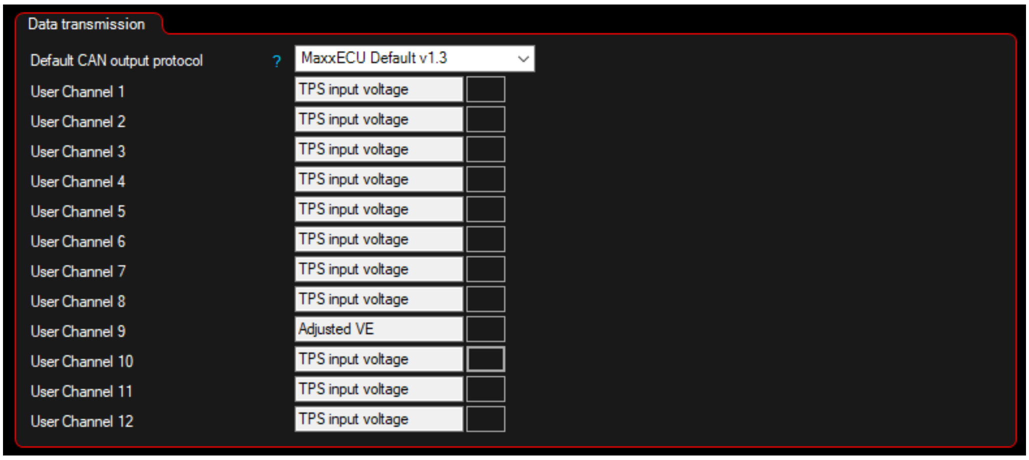
CANTCU - MaxxECU integration v1.0

- **CAN Speed** is configurable
- CAN2.0B, Standard 11bit identifiers
- All 16 bit values are **Little Endian**

MaxxECU Configuration

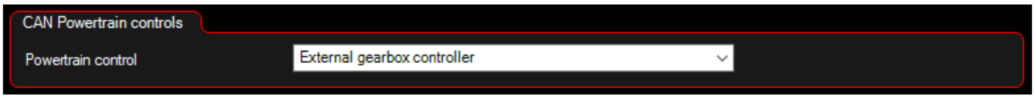
- **Default CAN output protocol v1.3** needs to be activated, and **User Channel 9** needs to be configured to send **Adjusted VE**.



Data transmission

Default CAN output protocol	MaxxECU Default v1.3
User Channel 1	TPS input voltage
User Channel 2	TPS input voltage
User Channel 3	TPS input voltage
User Channel 4	TPS input voltage
User Channel 5	TPS input voltage
User Channel 6	TPS input voltage
User Channel 7	TPS input voltage
User Channel 8	TPS input voltage
User Channel 9	Adjusted VE
User Channel 10	TPS input voltage
User Channel 11	TPS input voltage
User Channel 12	TPS input voltage

- **External gearbox controller** needs to be selected in the **CAN Powertrain controls**.



CAN Powertrain controls

Powertrain control: External gearbox controller

Available Realtime-values in CANTCU (sent from MaxxECU)

- *Engine RPM*
- *TPS Value*
- *Engine MAP*
- *Wheel Speeds*
- *Brake Switch*
- *Coolant Temperature*
- *Engine Oil Tempetature*

Available Realtime-values in MaxxECU (sent from CANTCU)

- *VSS Gear*
- *Gearbox Mode*
- *Gearbox Oil Temp*
- *Gearbox Calc Engine Torque*
- *Gearbox Engine Target Torque*
- *Shiftcut 0/1*
- *Blip 0/1*
- *Clutch Slip %*

Extra values sent on the CAN-bus from CANTCU to aid configuration:

Rate	***CAN ID***	***Type***	***Offset***	***Realtime variable***	***Unit***	***Factor***
50 Hz	0x606	UINT16	0	TCU RPM Target	RPM	1
50 Hz	0x606	UINT8	16	TCU Cut %	%	1
50 Hz	0x606	UINT8	24	TCU Blip %	%	1
50 Hz	0x606	INT16	32	TCU Delta TQ	Nm	1
50 Hz	0x606	UINT16	48	TCU Delta RPM	RPM	1
10 Hz	0x607	UINT8	0	TCU Auto/Manual		1
10 Hz	0x607	UINT8	8	TCU DL/DriveMode		1
10 Hz	0x607		16-63	RESERVED		

CAN-ID 0x606h

<table>
<tr><th></th><th><i>Bit 7</i></th><th><i>Bit 6</i></th><th><i>Bit 5</i></th><th><i>Bit 4</i></th><th><i>Bit 3</i></th><th><i>Bit 2</i></th><th><i>Bit 1</i></th><th><i>Bit 0</i></th></tr>
<tr><td><b><i>Byte 7</i></b></td><td colspan="8" rowspan="2">TCU Delta RPM</td></tr>
<tr><td><b><i>Byte 6</i></b></td></tr>
<tr><td><b><i>Byte 5</i></b></td><td colspan="8" rowspan="2">TCU Delta TQ</td></tr>
<tr><td><b><i>Byte 4</i></b></td></tr>
<tr><td><b><i>Byte 3</i></b></td><td colspan="8">TCU Blip %</td></tr>
<tr><td><b><i>Byte 2</i></b></td><td colspan="8">TCU Cut %</td></tr>
<tr><td><b><i>Byte 1</i></b></td><td colspan="8" rowspan="2">TCU RPM Target</td></tr>
<tr><td><b><i>Byte 0</i></b></td></tr>
</table>

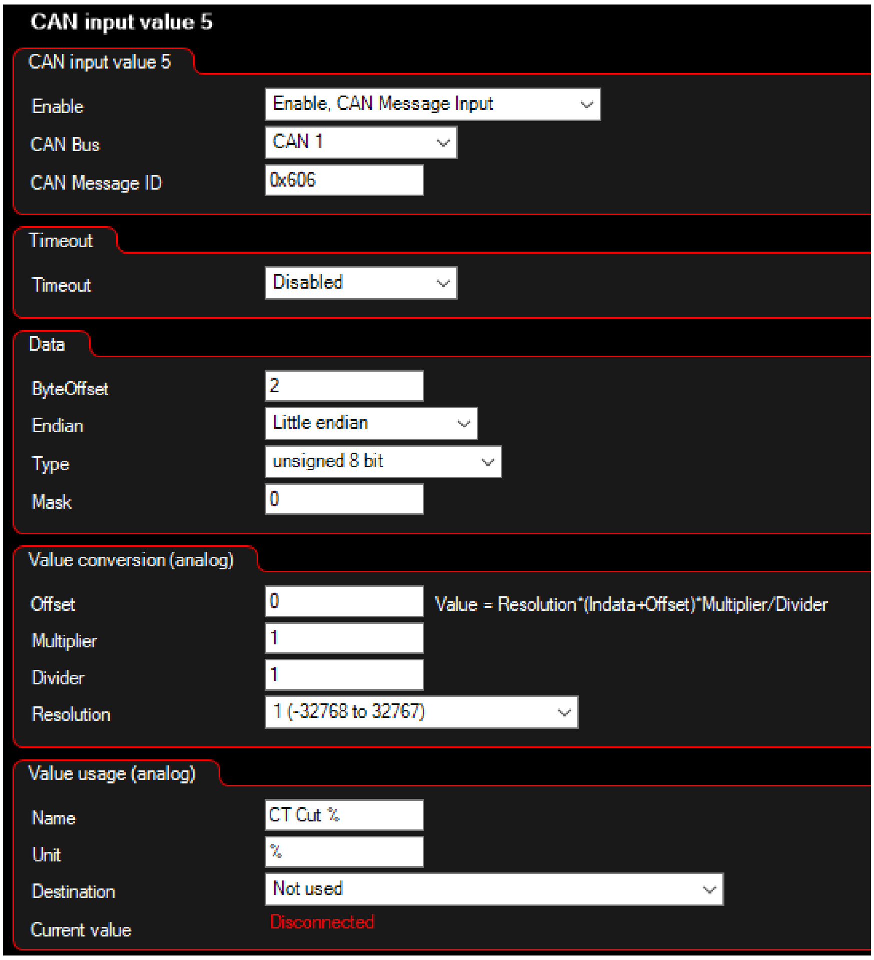
CAN-ID 0x607h

<table>
<tr><th></th><th><i>Bit 7</i></th><th><i>Bit 6</i></th><th><i>Bit 5</i></th><th><i>Bit 4</i></th><th><i>Bit 3</i></th><th><i>Bit 2</i></th><th><i>Bit 1</i></th><th><i>Bit 0</i></th></tr>
<tr><td><b><i>Byte 7</i></b></td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td><b><i>Byte 6</i></b></td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td><b><i>Byte 5</i></b></td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td><b><i>Byte 4</i></b></td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td><b><i>Byte 3</i></b></td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td><b><i>Byte 2</i></b></td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td><b><i>Byte 1</i></b></td><td colspan="8">TCU DL/DriveMode</td></tr>
<tr><td><b><i>Byte 0</i></b></td><td colspan="8">TCU Auto/Manual</td></tr>
</table>

Configuring Extra Values (CAN Inputs)

Extra values sent from CANTCU can be imported into MaxxECU by using the **CAN Inputs**. All the parameters needed for successful configuration of the extra values can be found in the tables in this document.

An example of configuring the extra value **Cut %** on a **CAN Input**:



CAN input value 5

CAN input value 5

Enable	Enable, CAN Message Input
CAN Bus	CAN 1
CAN Message ID	0x606

Timeout

Timeout	Disabled

Data

ByteOffset	2
Endian	Little endian
Type	unsigned 8 bit
Mask	0

Value conversion (analog)

Offset	0
Multiplier	1
Divider	1
Resolution	1 (-32768 to 32767)

Value = Resolution*(Indata+Offset)*Multiplier/Divider

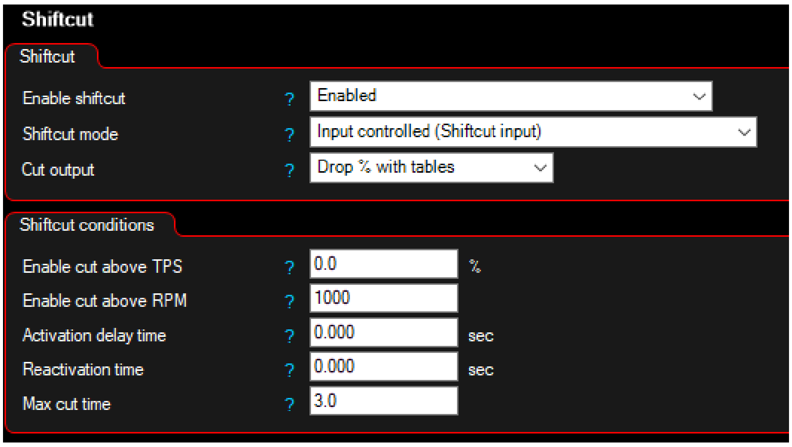
Value usage (analog)

Name	CT Cut %
Unit	%
Destination	Not used
Current value	Disconnected

Tuning - Shiftcut

There are several ways of achieving a proper shiftcut. By default, CANTCU sends a shiftcut request (0/1) to MaxxECU **Shiftcut Input**. This request can be combined with the various extra values available like **Cut %**, **Delta TQ**, **Delta RPM**, **Drive Mode** etc. to get a nice and versatile cut that works optimally in every situation.

- MaxxECU **Shiftcut Mode** must be configured as "*Input Controlled*" to let CANTCU decide the duration of the cuts.
- Cut should be enabled also without throttle input and without any delays



Shiftcut

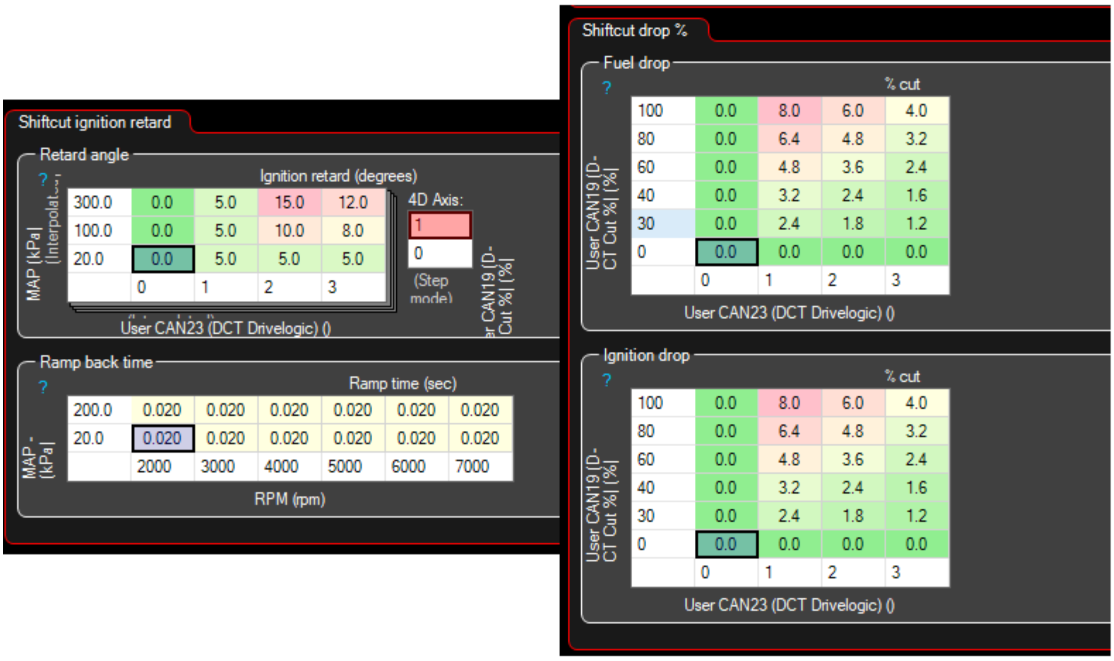
Shiftcut

Enable shiftcut	Enabled
Shiftcut mode	Input controlled (Shiftcut input)
Cut output	Drop % with tables

Shiftcut conditions

Enable cut above TPS	0.0	%
Enable cut above RPM	1000	
Activation delay time	0.000	sec
Reactivation time	0.000	sec
Max cut time	3.0	

Example configuration using F-Series DCT with cuts based on **Cut %** and **DriveLogic Level** from **CAN Inputs**



Shiftcut ignition retard

Retard angle

Ignition retard (degrees)

MAP [kPa] (Interpolat...)				
300.0	0.0	5.0	15.0	12.0
100.0	0.0	5.0	10.0	8.0
20.0	0.0	5.0	5.0	5.0
	0	1	2	3

User CAN23 (DCT Drivelogic) ()

4D Axis: 1, 0 (Step mode) — ...er CAN19 (D-... Cut %) [%]

Ramp back time

Ramp time (sec)

MAP - [kPa]						
200.0	0.020	0.020	0.020	0.020	0.020	0.020
20.0	0.020	0.020	0.020	0.020	0.020	0.020
	2000	3000	4000	5000	6000	7000

RPM (rpm)

Shiftcut drop %

Fuel drop

% cut

User CAN19 (D-CT Cut %) [%]				
100	0.0	8.0	6.0	4.0
80	0.0	6.4	4.8	3.2
60	0.0	4.8	3.6	2.4
40	0.0	3.2	2.4	1.6
30	0.0	2.4	1.8	1.2
0	0.0	0.0	0.0	0.0
	0	1	2	3

User CAN23 (DCT Drivelogic) ()

Ignition drop

% cut

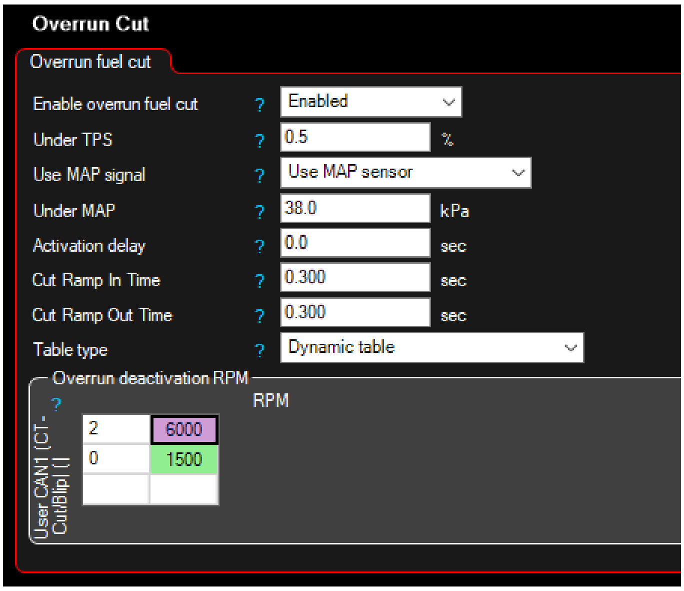
User CAN19 (D-CT Cut %) [%]				
100	0.0	8.0	6.0	4.0
80	0.0	6.4	4.8	3.2
60	0.0	4.8	3.6	2.4
40	0.0	3.2	2.4	1.6
30	0.0	2.4	1.8	1.2
0	0.0	0.0	0.0	0.0
	0	1	2	3

User CAN23 (DCT Drivelogic) ()

Tuning - Blips

There are many strategies available for blipping the throttle. The best way to achieve sophisticated and optimally targeted blips is to continuously vary the throttle position throughout the blip process. This functionality can be achieved in MaxxECU by using a custom blip process instead of the default built-in function (Motorsport->Throttle Blip). For the custom blip process, a **User Table** and some extra values from CANTCU assigned to **CAN inputs** are needed.

To be able to do blips without the MaxxECU default blip function, the **Overrun Cut** (if enabled) needs to be configured to deactivate overrun fuel cut during CANTCU blip requests. This can be done by adding a table to the function, and defining the **Overrun deactivation RPM** to be higher than the maximum blip RPM (e.g. 6000RPM) when CANTCU blip request (CAN input value=2) is active. The other parameters like **Activation delay** and **Cut Ramp In/Out Times** may need some tweaking, but a configuration similar to the one below should provide a good starting point.



Overrun Cut

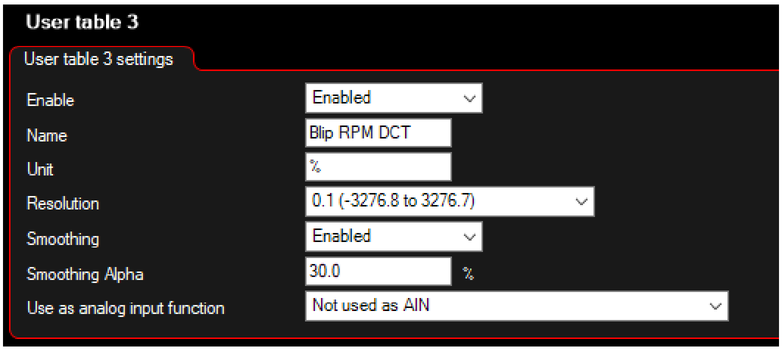
Overrun fuel cut

Enable overrun fuel cut	Enabled	
Under TPS	0.5	%
Use MAP signal	Use MAP sensor	
Under MAP	38.0	kPa
Activation delay	0.0	sec
Cut Ramp In Time	0.300	sec
Cut Ramp Out Time	0.300	sec
Table type	Dynamic table	

Overrun deactivation RPM

User CAN1 [CT - Cut/Blip] [ ]	RPM
2	6000
0	1500

A free **User Table** needs to be enabled. The **User Table** settings can be configured freely. Using smoothing can help calm down throttle control oscillations if needed.

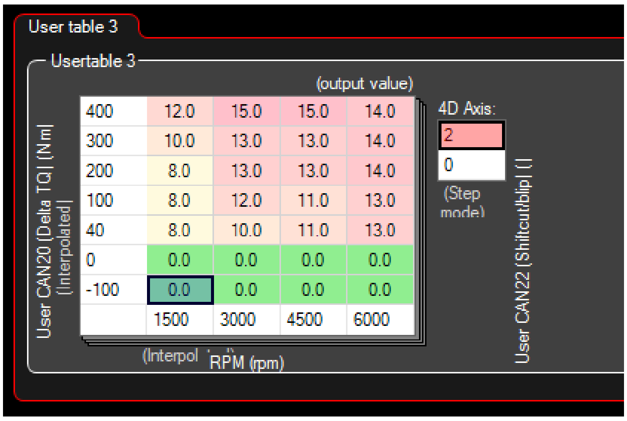


User table 3

User table 3 settings

Enable	Enabled	
Name	Blip RPM DCT	
Unit	%	
Resolution	0.1 (-3276.8 to 3276.7)	
Smoothing	Enabled	
Smoothing Alpha	30.0	%
Use as analog input function	Not used as AIN	

The **User Table** output is a percentage added to throttle position depending on various user-definable inputs. **Delta TQ** or **Delta RPM** combined with **Engine RPM** have been proven to work nicely, but basically any combination can be used, thanks to the freely configurable axes.

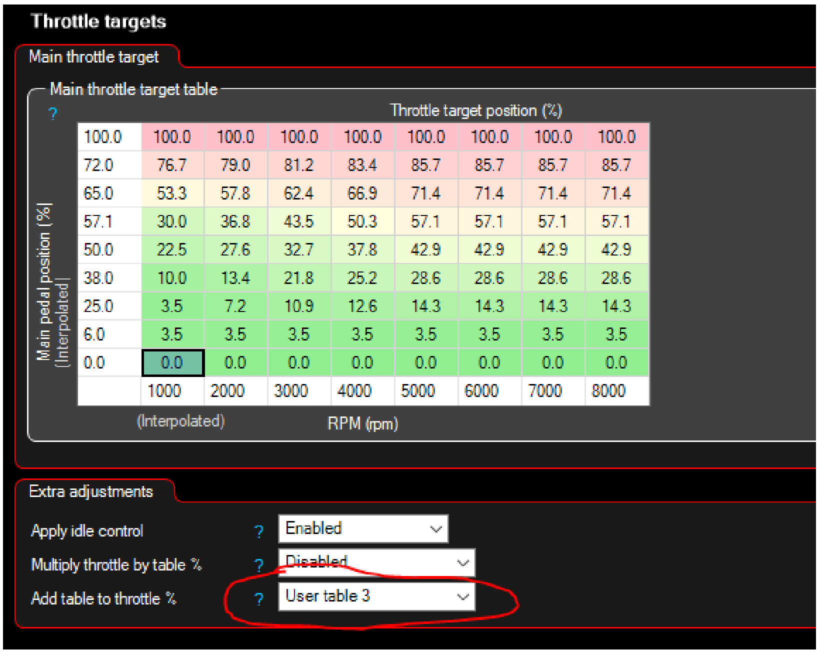

User table 3
Usertable 3
User CAN20 [Delta TQ] [Nm] [Interpolated]
User CAN22 [Shiftcut/blip] [ ]
(Interpol
RPM (rpm)

(output value)

400	12.0	15.0	15.0	14.0
300	10.0	13.0	13.0	14.0
200	8.0	13.0	13.0	14.0
100	8.0	12.0	11.0	13.0
40	8.0	10.0	11.0	13.0
0	0.0	0.0	0.0	0.0
-100	0.0	0.0	0.0	0.0
	1500	3000	4500	6000

4D Axis:
2
0
(Step mode)

To activate the blip function, the **User Table** created earlier is added to **Main Throttle Target** (E-Throttle -> Throttle Targets).



Throttle target position (%)

100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0
72.0	76.7	79.0	81.2	83.4	85.7	85.7	85.7	85.7
65.0	53.3	57.8	62.4	66.9	71.4	71.4	71.4	71.4
57.1	30.0	36.8	43.5	50.3	57.1	57.1	57.1	57.1
50.0	22.5	27.6	32.7	37.8	42.9	42.9	42.9	42.9
38.0	10.0	13.4	21.8	25.2	28.6	28.6	28.6	28.6
25.0	3.5	7.2	10.9	12.6	14.3	14.3	14.3	14.3
6.0	3.5	3.5	3.5	3.5	3.5	3.5	3.5	3.5
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	1000	2000	3000	4000	5000	6000	7000	8000

Apply idle control	?	Enabled
Multiply throttle by table %	?	Disabled
Add table to throttle %	?	User table 3

NOTE!

All tuning should always be done by a professional in safe environment (track/dyno)

Before activating the blip function, it's recommended to test downshifting and verify (realtime or logging) that the user table is behaving correctly during the blip. Starting values for tuning the blip should be low and gradually increased to avoid overrevving and undesired behavior/acceleration during the shift.