PUNCH | Softronix

Experience in optimization of MCS C-Code for hard real-time applications

GTM Tech Days – Stuttgart, Sep. 22nd, 2022

Vittorio Calvia

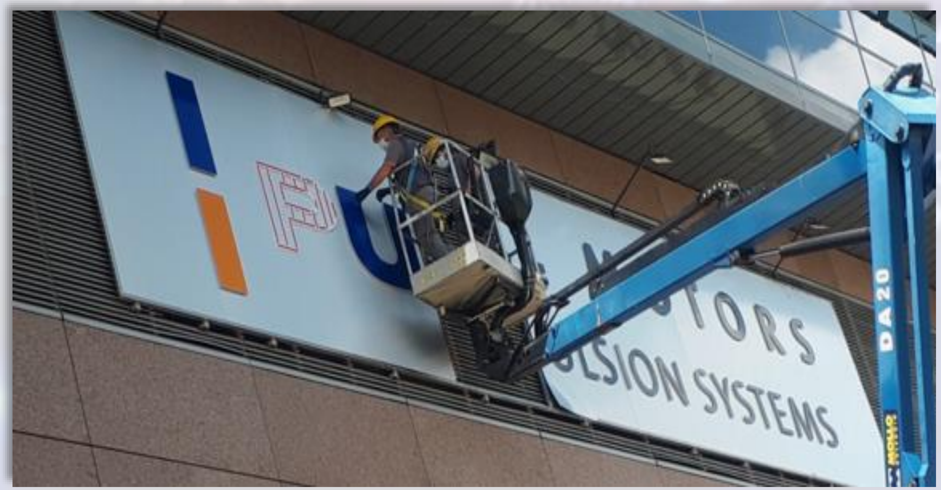
PUNCH Softronix

Home | News | Company | Plants & Facilities | Images | Videos

Punch Group Acquires General Motors Propulsion Engineering Center in Turin, Italy

2020-02-27

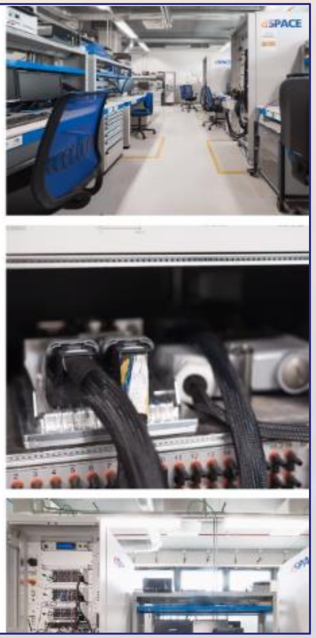
TURIN, Italy — Punch Group and General Motors announced today that Punch Group has acquired GM's propulsion engineering center in Turin. The transaction between the two companies includes an engineering services agreement to support GM's global product



PUNCH | Softronix

PUNCH SOFTRONIX IS BORN

2022-01-01

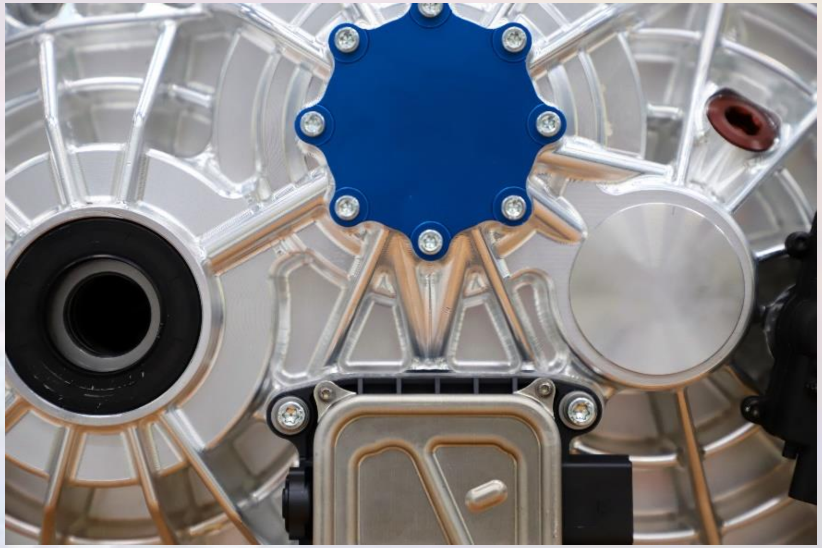


"PUNCH Torino will now create a center focused on the electronics, controls and software giving life to the new PUNCH Softronix business unit"

PUNCH | Softronix

PUNCH Softronix

Softronix has a consolidated experience in designing and implementing Complex Drivers in GTM for gas and Diesel ECU applications. Complex drivers microcode has been developed directly in MCS assembly code.

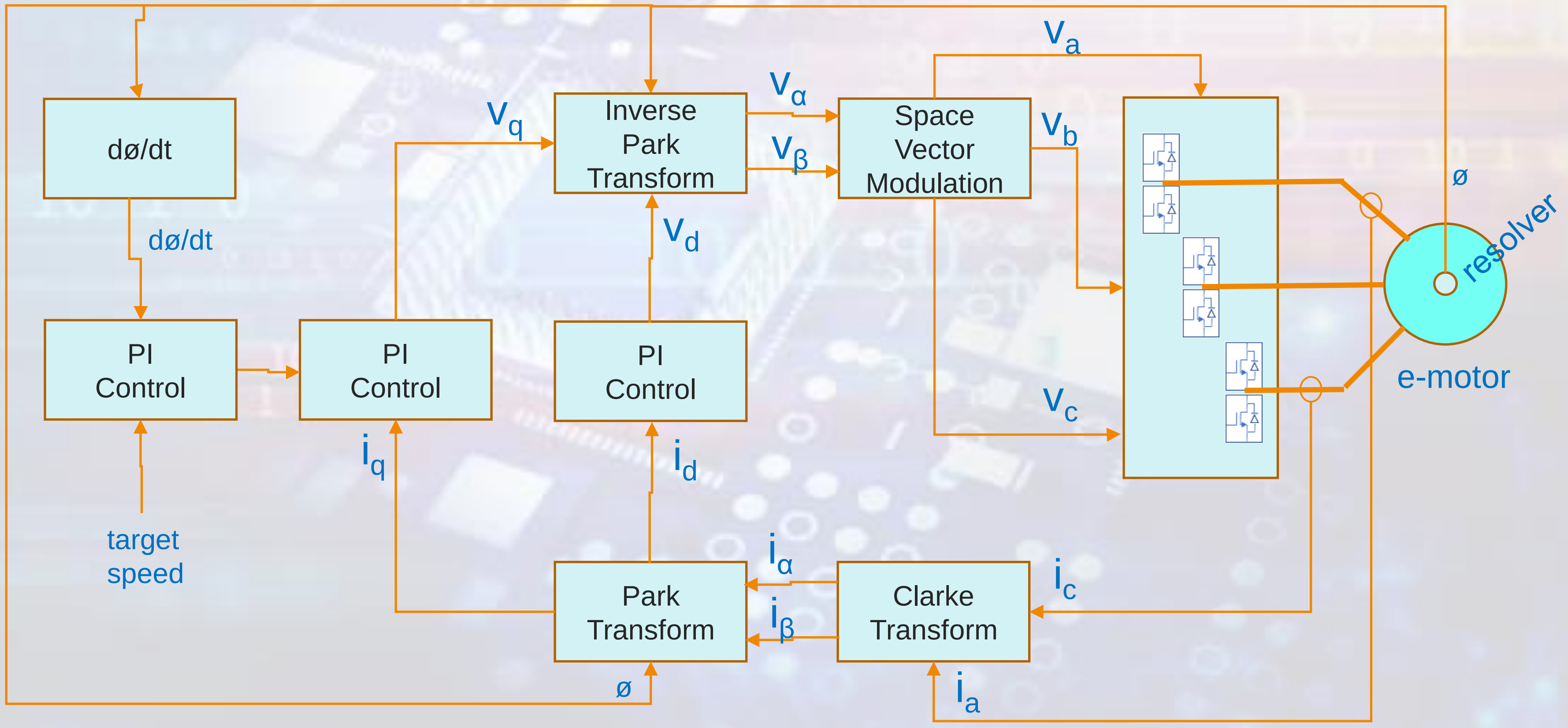


For a customer, Softronix recently completed a software project for driving a traction power inverter controller in the electrical domain.

Context

- As per customer requirements, the low-level control design and implementation included the usage of the Bosch GTM IP in an Infineon TC3xx microcontroller.
- Customer also required microcode developed using a compiler that converts C-code in MCS assembly.
- The implementation resulted in a very fast control loop, from torque demand and motor speed, to the space vector modulation for driving the inverter, without charging the microcontroller cores.
- Target for calculation and actuation: **20 µs**.

Context



dø/dt
dø/dt
PI Control
target speed
PI Control
i_q
V_q
Inverse Park Transform
V_d
PI Control
i_d
V_α
V_β
Space Vector Modulation
V_a
V_b
V_c
ø
resolver
e-motor
Park Transform
i_α
i_β
Clarke Transform
i_c
i_a
ø

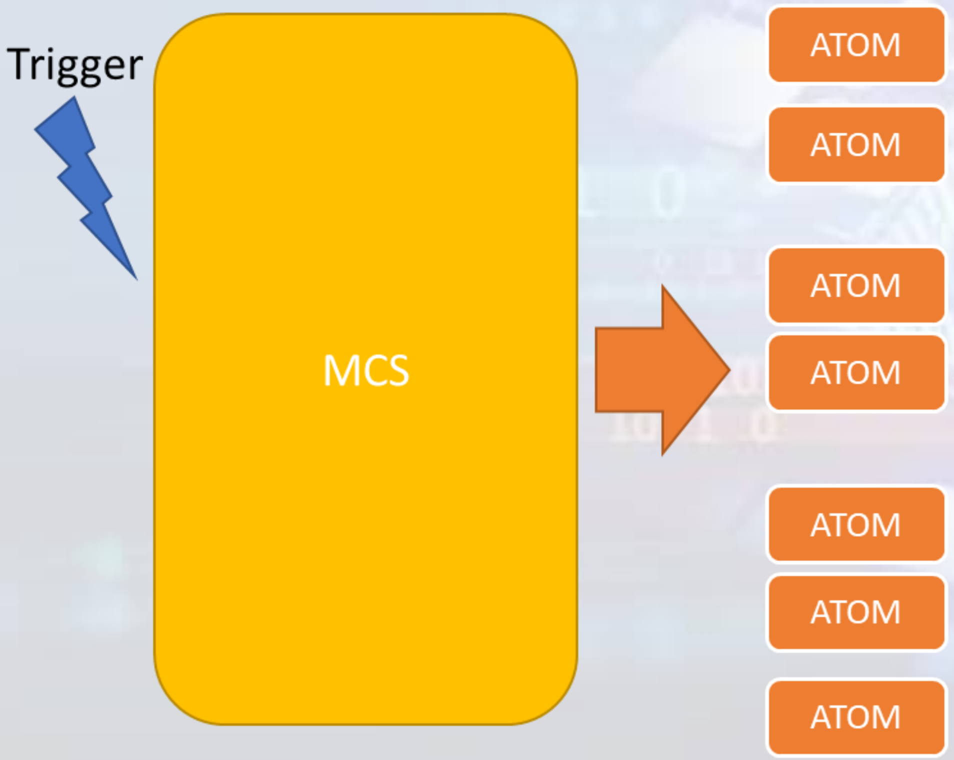
Design and coding considerations

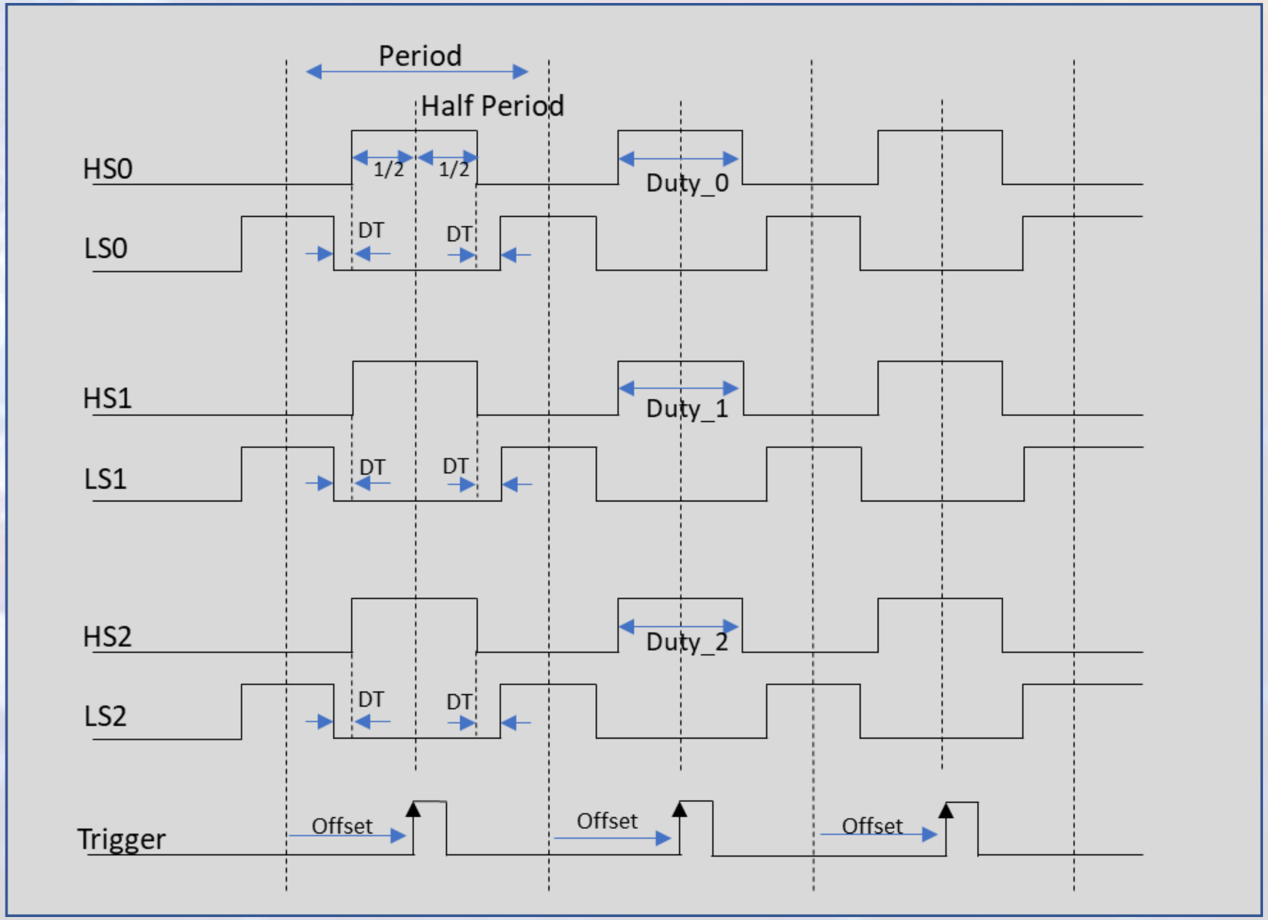
- One MCS channel is used to perform calculations: currents are measured and provided as input. MCS processes data and generates duty cycles and period to command 7 ATOMs in PWM mode.

- MCS channel has been set in accelerated mode.

- Data processing includes trigonometric functions and square root calculation. They have been adapted to run in fixed point at 24 bits.

Design and coding considerations





First results

A preliminary estimation has been made to understand if given target could be satisfied. Pseudo code has been generated to estimate execution time

Virtual model of GTM has been used to measure execution time of first come implementation

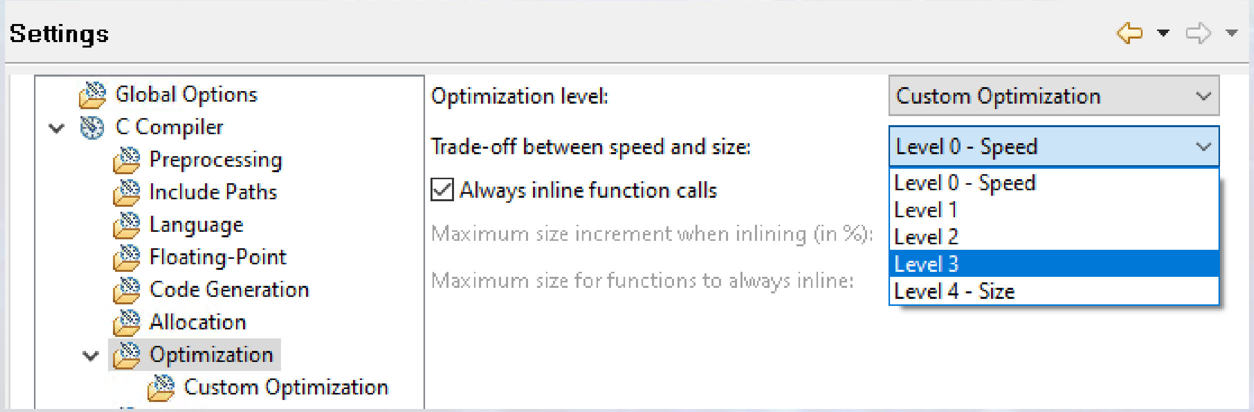
Code has been executed on a real evaluation board with Infineon microprocessor

	Code	Execution time [µs]
Estimation	Pseudo Code	16.0
Virtual	Compiled C Code	15.6
EVB		41.1

Compiler optimizations

Compiler offers several options to optimize generated microcode. User can decide if priority should be given to size of generated code or speed.

Speed optimization have been activated. As side effect, size reached the limit of MCS memory.



Compiler optimizations

Compiler offers several options to optimize generated microcode. User can decide if priority should be given to size of generated code or speed.

Speed optimization have been activated. As side effect, size reached the limit of MCS memory.

	Execution time [µs]
Old EVB measurement	41.1
New EVB measurement	26.7

Grouping of ATOM commands

To reduce the number of memory accesses, some ATOM commands have been grouped together: instead of activating 7 ATOMs one by one, a single command is sent as the logical “or” of the channels.

```
for (Index = 0; Index < (NumOfPhases*2); Index++)
{
  ChannelEnable(Index);
}

void ChannelEnable(uint8 Index)
{
  Addrs = Addr_h_ATOM_AGC_GLB_CTRL;
  Data = (Enable_Update << ((UPEN_OFFSET + (Index*2))));
  __mcs_bwr(Addrs, Data);
}
```

```
Addrs = Addr_h_ATOM_AGC_GLB_CTRL;
for (Index = 0; Index < (NumOfPhases*2); Index++)
{
  Data = Data | (Enable_Update << ((UPEN_OFFSET + (Index*2))));
}
__mcs_bwr(Addrs, Data);
```

Grouping of ATOM commands

To reduce the number of memory accesses, some ATOM commands have been grouped together: instead of activating 7 ATOMs one by one, a single command is sent as the logical “or” of the channels.

	Execution time [µs]
Old EVB measurement	26.7
New EVB measurement	25.6

Usage of local variables and extended register set

Another method to reduce the number of memory accesses is to increase the usage of local variables.

- Extended register set has been enabled. The number of stack read/write operations decreased by 30%.
- Some subfunctions have been collapsed into a single one. This allowed the reuse of some local parameter/variable.
- ACB was an unused register since no ARU access was needed. It has been used to store a frequently used variable. This reduced related read/write operations.

Usage of local variables and extended register set

Another method to reduce the number of memory accesses is to increase the usage of local variables.

- Extended register set has been enabled. The number of stack read/write operations decreased by 30%.
- Some subfunctions have been collapsed into a single one. This allowed the reuse of some local parameter/variable.
- ACB was an unused register since no ARU access was needed. It has been used to store a frequently used variable. This reduced related read/write operations.

	Execution time [µs]
Old EVB measurement	25.6
New EVB measurement	21.4

Removal of “for” loops

Examining code, it contained some for loop where the index was pointing to an array of a structure. On every iteration, several microcode instructions were generated to point to the proper address. The loop has been converted in a sequence of “if” statements.

```
Addrs = Addr_h_ATOM_AGC_GLB_CTRL;
for (Index = 0; Index < (NumOfPhases*2); Index++)
{
  Data = Data | (Enable_Update << ((UPEN_OFFSET + (Index*2))));
}
__mcs_bwr(Addrs, Data);
```

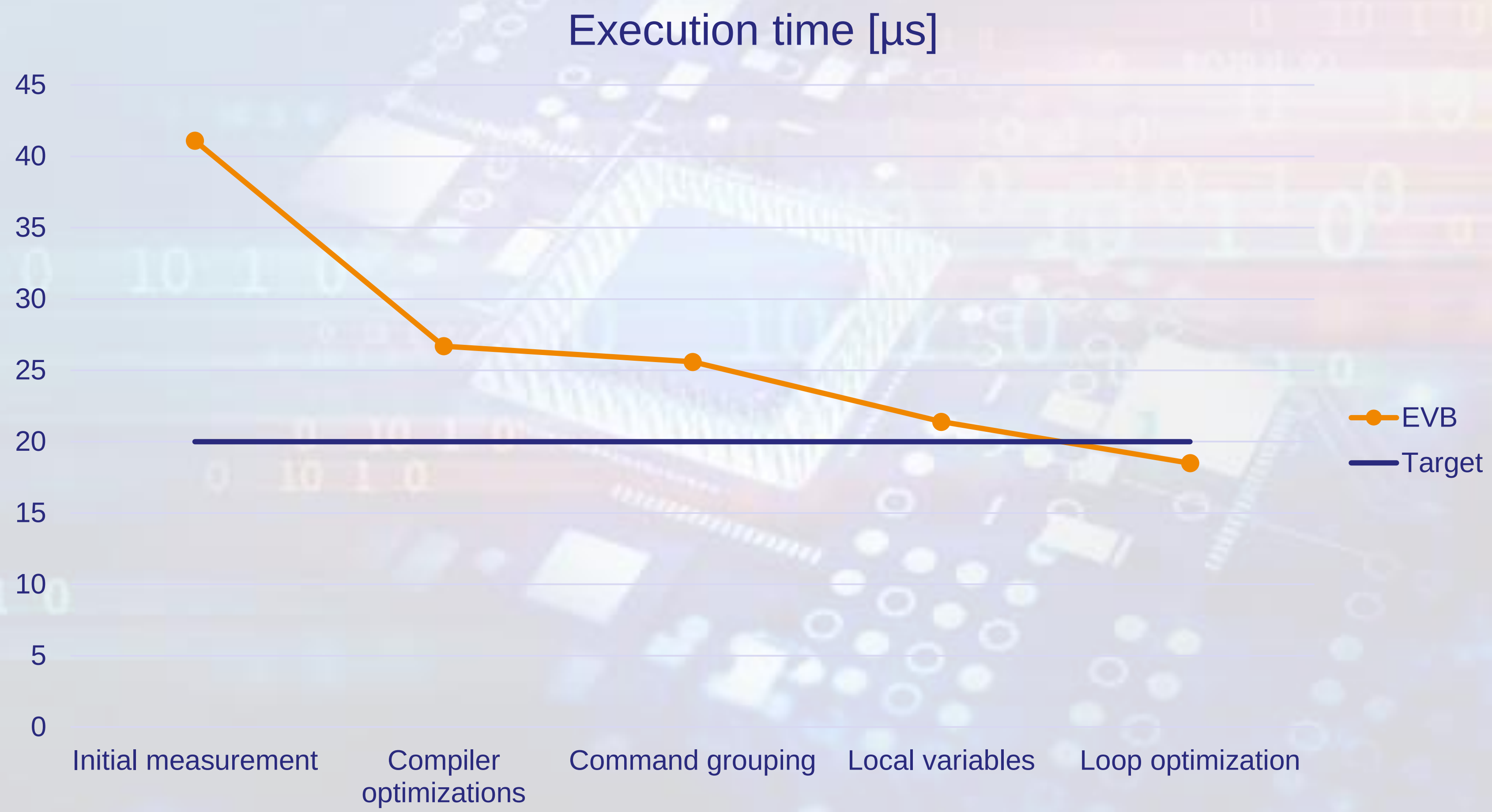
```
Addrs = Addr_h_ATOM_AGC_GLB_CTRL;
if(NumOfPhases > 0)
{
  Data = (Data | (Enable_Update<<UPEN_OFFSET) | (Enable_Update<<(2+UPEN_OFFSET)));
}
if(NumOfPhases > 1)
{
  Data = (Data | (Enable_Update<<(4+UPEN_OFFSET)) | (Enable_Update<<(6+UPEN_OFFSET)));
}
if(NumOfPhases > 2)
{
  Data = (Data | (Enable_Update<<(8+UPEN_OFFSET)) | (Enable_Update<<(10+UPEN_OFFSET)));
}
Addrs = Addr_h_ATOM_AGC_GLB_CTRL;
__mcs_bwr(Addrs, Data);
```

Removal of “for” loops

Examining code, it contained some for loop where the index was pointing to an array of a structure. On every iteration, several microcode instructions were generated to point to the proper address. The loop has been converted in a sequence of “if” statements.

	Execution time [µs]
Old EVB measurement	21.4
New EVB measurement	18.5

Optimization summary



PUNCH | Softronix

Conclusion

C-code compiler for MCS is very good in generating assembly code. If the user specifies its target (speed or code size), a lot of optimizations are very effective.

Manual code still has slightly better performances since developer can manage registers and memory in an optimal way.

Virtual environment is useful. Even if real board is not completely emulated, it helped a lot during development phase.

Having a good knowledge of GTM allows optimizations that a C developer could not consider. At the same time, the experience in GTM coding can be used to use the module at its best and reach customer target.

COMPLEX DRIVERS USING GTM BOSCH IP – OUR EXPERIENCE

Drivers (all with GTM 3.1.5)	Served Applications					In Production
	Diesel ICE	Gas ICE	Fuel Cells	Punch El. Platform (H2 ICE/Diesel ICE/..)	EV	
Direct Injection Fueling Output	✓	✓		✓		2023-Diesel, Gas
Fuel Pulse Monitoring	✓	✓		✓		2023-Diesel, Gas
Voltage & Current Synchronous Samping	✓	✓		✓		2023-Diesel, Gas
Port Fuel Injection Outputs		✓		✓		2024-Gas
High Pressure Fuel Pump Driver		✓				2023-Gas
Knock Inputs		✓		✓		2023-Gas
Spark Outputs		✓		✓		2023-Gas
Engine Position System Inputs	✓	✓		✓		2023-Gas
Step Cam Outputs		✓				2023-Gas
Cylinder Deactivation Inputs/Outputs		✓				2023-Gas
Complex pulse Output with Synchronous Diagnostic				✓		-
Time base Injector Outputs			Planned in 2023			-
Inverter Control					✓	-