

# 65 nm SoC design based on an emerging standard: SPIRIT



**Olivier FLORENT, François REMOND**  
STMicroelectronics, Grenoble , France

# 65 nm Set Top Box design – STB7200

**24 Million synthesized gates**

**6 Mbits embedded RAM**

**64 bits CPU core**

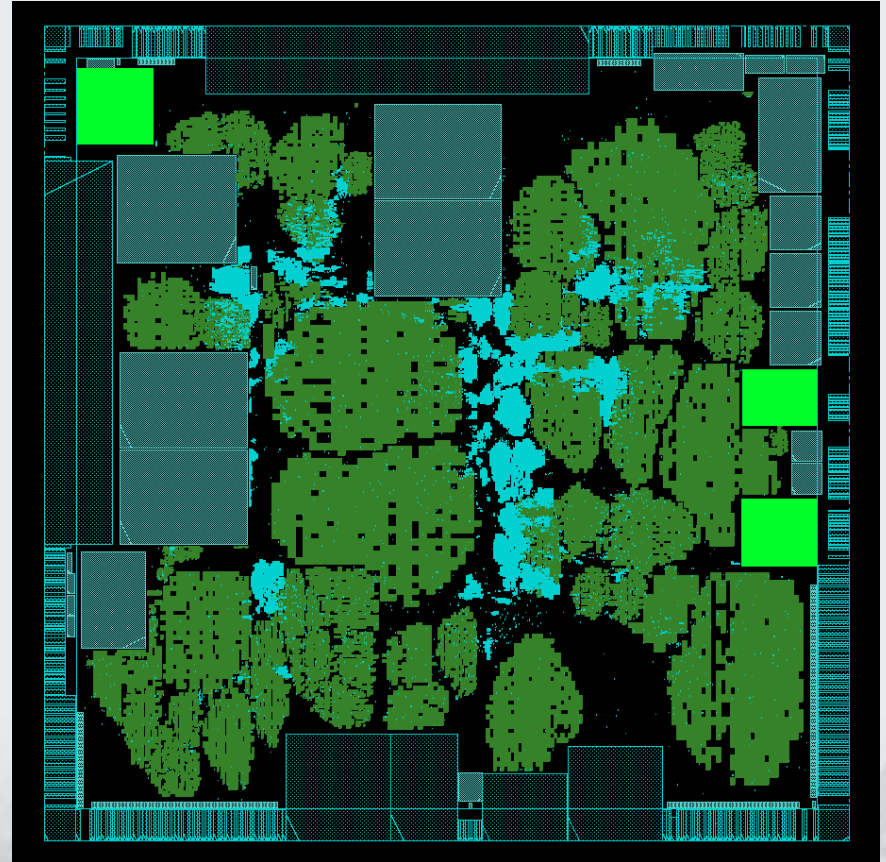
**200MHz On-Chip-Bus**

**Dual audio decode**

**Dual MPEG4HD decode**

**DDR2, SATA, USB interface**

**40 IPs & 30 custom blocks grouped  
in 8 subsystem**



# Drivers for SPIRIT based design flow

- Reduce Front-End design capture by enabling correct by construction assembly
  - thanks to the protocol based design capture approach
- Interface early with SoC verification and software development
  - by mixing different levels of IP models and by taking advantage of memory map consolidation
- Use a defined and open format
  - enable usage of standard CAD tools and development of custom toolset



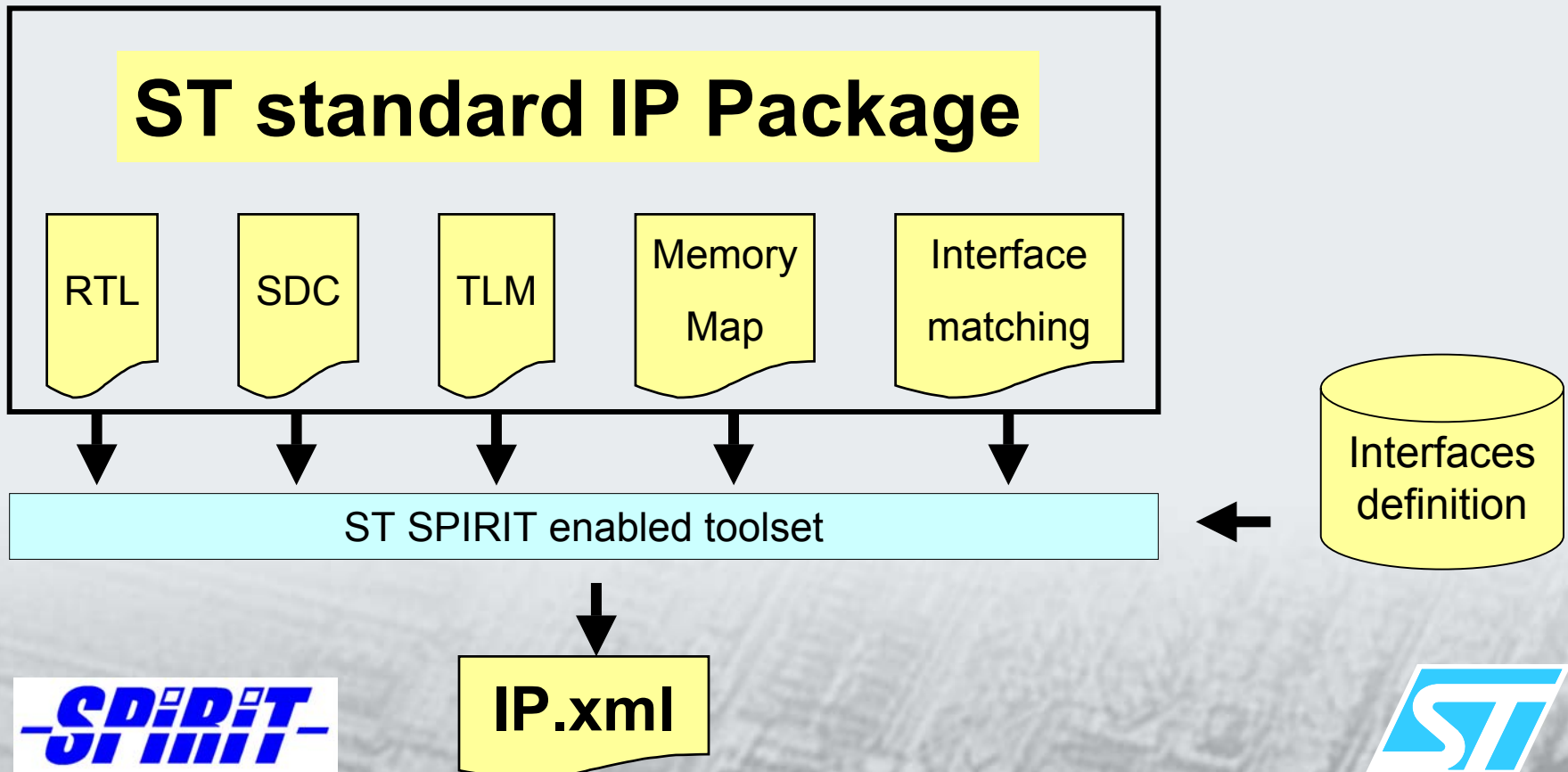
# SPIRIT based assembly flow in 3 steps

1. Generation of SPIRIT meta-datas for each library element
2. Bottom-up design assembly, including On-Chip-Bus configuration, relying on commercial tool (Synopsys coreAssembler)
3. Usage of assembly database (SPIRIT format) to address several areas : logical & physical design, verification, software, using ST SPIRIT enabled toolset.



# Generation of SPIRIT Meta-Data

IP meta-data are automatically built from “ST standard IP Package” using custom ST SPIRIT enabled toolset



# SPIRIT based chip assembly flow

1. Custom tool developed to capture the pad ring, generating an “empty core” module
2. coreAssembler working database populated using reference SPIRIT views through data-management and version control
3. coreAssembler used :
  1. to assemble bottom-up the IP’s into logical subsystems,
  2. to connect them to the On-Chip-Bus, matching the “empty core” pinout.
4. SPIRIT database is generated out of coreAssembler within a standard ST FE-package data-structure

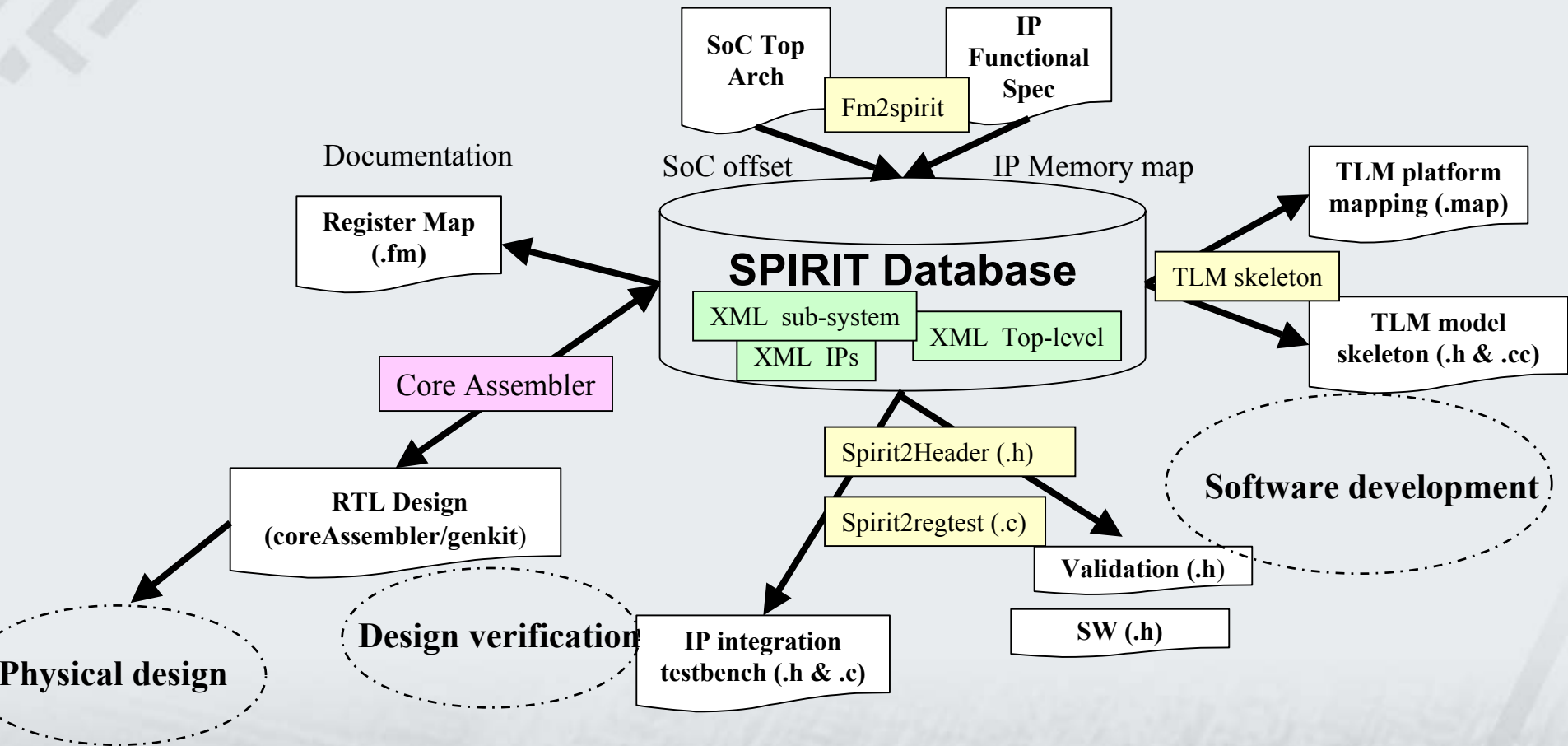


# Details of SPIRIT based IP assembly steps

1. Main IP  $\leftrightarrow$  IP or IP  $\leftrightarrow$  Bus interfaces are automatically connected, based on their protocol definition
2. Remaining wire connectivity is completed by hand
3. Batch script replay is enabled for quick update (spec changes)
4. External signal & interfaces are exported to the next level of hierarchy
5. ST-Bus is instantiated and configured



# Links with Design, Verification & Software



# Conclusion

- SPIRIT cost of adoption is low
- Open standard = access to standard CAD tools and reusable scripting (vendor agnostic)
- Unified database helps to address consistently and concurrently different design areas
- Growing adoption in STMicroelectronics
  - Cellular communication devices
  - Complex IP designs (video-processor)
- Next steps
  - Reuse of commercially available IP SPIRIT views

