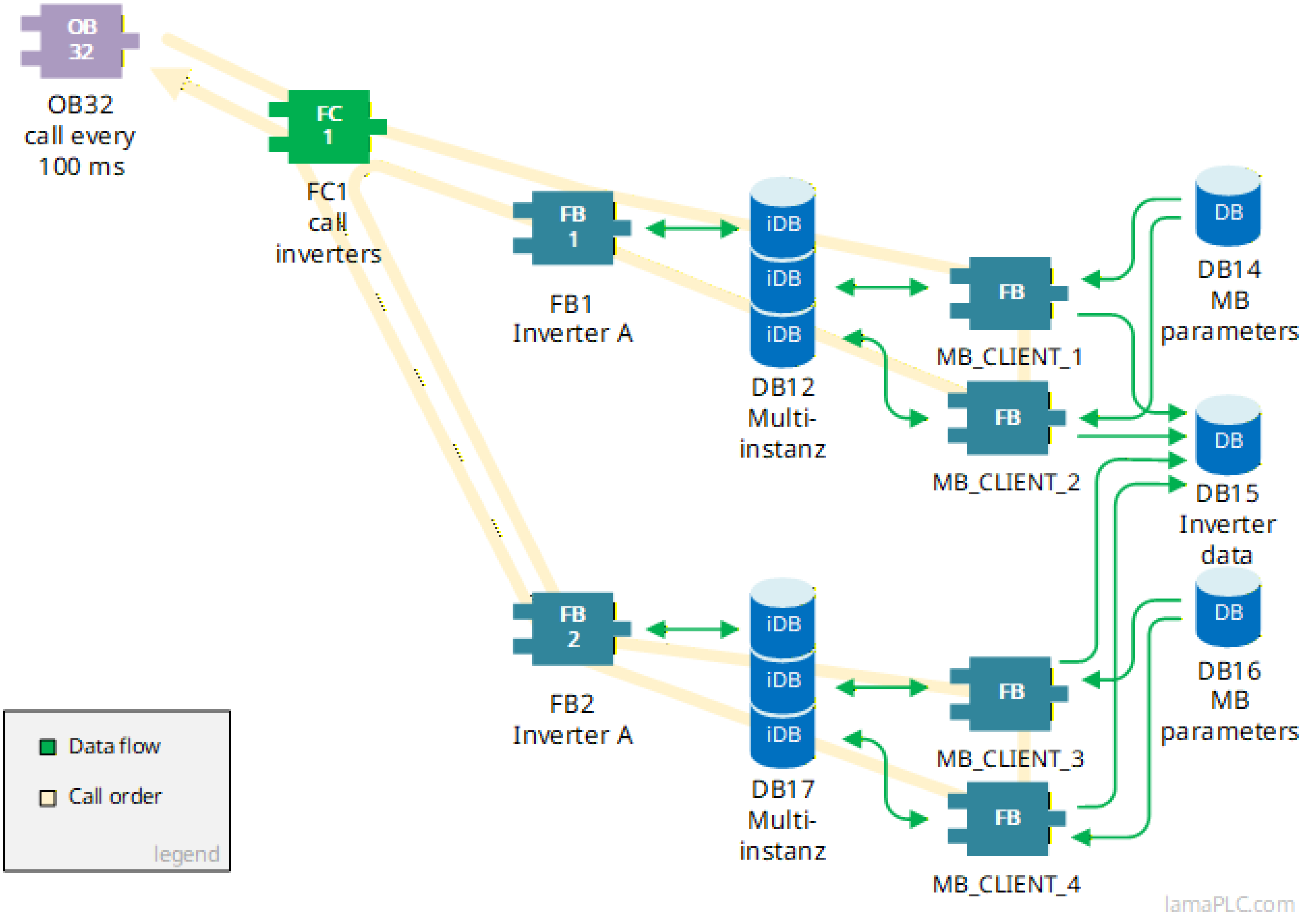
OB, DB, FB, FC

PLCs differ from PCs in several ways. Their structure and programming architecture are simpler and more straightforward. A key difference is that they lack a traditional file system; instead, they consist of four main components:

Symbol	Description
OB	OB – organization block: OBs play a special role in running programs, as they essentially start programs based on various criteria. The simplest is OB1, which runs 'continuously'. As soon as a complete execution cycle ends — that is, all programs started from the OB have run — it restarts them after the signals are output and read. There are scheduled, error-related, and interrupt OBs; more details are available in the OB section.
DB	A DB – data block: A DB organizes data based on various aspects and has two primary types. The global DB is created directly by the programmer, whereas the instant DB is a storage block assigned to an FB. For more information, refer to the DB chapter.
FC – FB: Throughout the program, modules that are repeated multiple times or are structurally different can be organized into FBs or FCs. This approach is crucial during commissioning and later corrections to ensure the code remains clear and well-structured. For instance, unorganized code 'dumped' into OB1 raises concerns for experienced programmers reviewing the system. Several other red flags exist, where acceptance only occurs if the code is entirely rewritten.	
FC	FC – Function. These are basic program blocks that do not pass variables to subsequent cycles. Examples include a module that creates colors in an HMI or a segment that performs mathematical operations. Further details are available in the FC chapter.
FB	FB – function block. Each FB is always linked to a Data Block (DB) that holds some of its data. This DB operates independently of cycles. Sometimes, it isn't a unique DB but a multi-instance DB. More information about these can be found in the FB chapter.
iDB	iDB - instant DB: Although it appears as a standard DB in the TIA Portal without a distinct symbol, I am explicitly identifying this DB type in this document. It is always linked to a specific FB and stores the non-temporary data of that FB. For more information, refer to the DB chapter.

An example of program calls



1. The OB32 calls the FC1 cyclically every 100 ms.
2. The FC1 first calls the first inverter, then the second inverter.
3. The FB1, whose instant data block is DB12, is called. The FB1 initiates a Modbus Client call (MB_CLIENT_1), which reads its parameters (IP, unitID, etc.) from DB14. Because this FB is embedded within FB1, it receives a multi-instant block in DB12. The results of the Modbus read are written to DB15.
4. The second Modbus read operates similarly, with its parameters also in DB14 and its multi-instant block in DB12.
5. FC1 is called again, which then calls inverter2, following a similar call sequence as the first case.
6. Control returns to OB32, which waits for the next 100 ms cycle and then calls FC1 again.