S7-1200 and STEP 7 Basic
- The S7-1200 is the successor of the S7-200 and is available in June 2009. It is designed for and will be sold in the world wide market place.
- The S7-200 will remain an active Siemens product.
- STEP 7 Basic v10.5 programming package (ordered separately) is used with the S7-1200
  - Supports LAD and FBD. STL is not supported.
  - Includes WinCC Basic for configuring HMI Basic panels.
  - No separate USB license stick is required. The software is automatically activated when installed.
- An export-import function for an S7-1200 project is not available in the initial release. To move project files from one PC to another PC, use the Windows explorer and PKZIP to copy the project files directory structure.

S7-1200 Approvals
- S7-1200 hardware has the necessary approvals for the US and European market.
- The S7-1200 has FM approval for hazardous location:
The Factory Mutual Research (FM): Approval Standard Class Number 3600 and 3611 Approved for use in:
  - Class I, Division 2, Gas Group A, B, C, D, Temperature Class 40° C
  - Class I, Zone 2, IIC, Temperature Class T4 Ta = 40° C
- The S7-1200 hardware has UL and CE approvals.
## System Expandability

### Hardware
- S7-200
- S7-1200

### Communication
- EM (Expansion Module)
- CP (Communication Processor)

### HMI
- CPU
- EM (Expansion Module)
- CP (Communication Processor)

### Memory
- SM (Signal Module)

### Block concept
- Instruction set
- Resources

### Timers
- New data types

### Counters
- CPU

### Technology
- Resources

### Resources
- CPU 1214C DC/DC/DC
- CPU 1212C DC/DC/DC
- CPU 1211C DC/DC/DC

### System Expandability

#### S7-1200 CPU
- CPU 1211C
  - 6 IN - 4 OUT
  - 2 IN (0-10V)
  - CPU 1211C DC/DC/DC
  - CPU 1211C AC/DC/Relay
  - CPU 1211C DC/DC/Relay
- CPU 1212C
  - 8 IN – 6 OUT
  - 2 IN (0-10V)
  - CPU 1212C DC/DC/DC
  - CPU 1212C AC/DC/Relay
  - CPU 1212C DC/DC/Relay
- CPU 1214C
  - 14 IN -10 OUT
  - 2 IN (0-10V)
  - CPU 1214C DC/DC/DC
  - CPU 1214C AC/DC/Relay
  - CPU 1214C DC/DC/Relay

#### S7-1200 Signal Modules and Signal Boards
- SM 1221 8 x 24 VDC Input
- SM 1221 16 x 24 VDC Input
- SM 1222 8 x 24 VDC Output
- SM 1222 16 x 24 VDC Output
- SM 1222 8 x Relay Output
- SM 1222 16 x Relay Output
- SM 1223 8 x 24 VDC Input / 8 x 24 VDC Output
- SM 1223 16 x 24 VDC Input / 16 x 24 VDC Output
- SM 1223 8 x 24 VDC Input / 8 x Relay Output
- SM 1223 16 x 24 VDC Input / 16 x Relay Output
- SM 1231 4 x Analog Input
- SM 1232 2 x Analog Output
- SM 1234 4 x Analog Input / 2 x Analog Output
- SB 1223 2 x 24 VDC Input / 2 x 24 VDC Output
- SB 1232 1 Analog Output

#### S7-1200 Communication Modules
- CM 1241 RS232
- CM 1241 RS485
### CPU on-board I/O and SB (Signal Board) I/O

<table>
<thead>
<tr>
<th>Hardware</th>
<th>Communications</th>
<th>Memory</th>
<th>Technology</th>
<th>Resources</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

#### S7-200 CPU 224XP vs. S7-1200 CPU 1214C

<table>
<thead>
<tr>
<th>Feature</th>
<th>S7-200 CPU 224XP</th>
<th>S7-1200 CPU 1214C</th>
</tr>
</thead>
<tbody>
<tr>
<td>DI Digital Input</td>
<td>14</td>
<td>14+ (2 on SB)</td>
</tr>
<tr>
<td>DO Digital Output</td>
<td>10</td>
<td>10+ (2 on SB)</td>
</tr>
<tr>
<td>AI Analog Input</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>AO Analog Output</td>
<td>1</td>
<td>1 on SB</td>
</tr>
<tr>
<td>PWM / PTO</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pulse Width Modulation</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pulse Train Output</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HSC High Speed Counter</td>
<td></td>
<td>6</td>
</tr>
<tr>
<td>PID Closed loop controller</td>
<td>8</td>
<td>16</td>
</tr>
</tbody>
</table>

The Signal Board expands the S7-1200 CPU I/O without increasing the CPU footprint size.

The initial release on the STEP 7 Basic V10.5 software and S7-1200 hardware set does not include the S7-200 expansion module types shown below. To use the S7-1200 CPU for an S7-200 application that uses these modules, you must use a different method to replace the function of these I/O modules. For example, there is no S7-1200 RTD or Thermocouple module. However, you can use standard analog modules with temperature sensors. The user program can perform a linearization of the sensor output.

#### S7-200 module types with no equivalent S7-1200 module (for initial S7-1200 hardware release)

- EM 221 Digital 8 AC inputs (8 x 120/230 VAC)
- EM 222 Digital 8 AC Outputs (8 x 120/230 VAC)
- EM 222 Digital Output (4 x Relays 10A)
- EM231 Analog Input, 8 Inputs
- EM231 Analog Output, 4 Outputs
- EM 231 Analog Input RTD, 2 Inputs
- EM 231 Analog Input RTD, 4 Inputs
- EM 231 Analog Input RTD, 8 Inputs
- EM 231 Analog Input Thermocouple, 4 Inputs
- EM 231 Analog Input Thermocouple, 8 Inputs
- EM 241 Modem module
- EM 253 Position module
- EM 277 PROFIBUS DP module
- SIWAREX MS Micro Scale module
- CP 243-2 ASi master module
- CP 243-1 IT Internet module
- SIMATIC TD (RS485 connection Text Display) TD 100C, TD 200, TD 200C, TD400C, OP73micro, TP177micro
Hardware configuration S7-200

- Expansion modules are automatically recognized when connected and powered
- STEP 7-Micro/WIN "SET PG/PC Interface" window configures the communication driver
- STEP 7-Micro/WIN "Communications" uses the selected driver to scan and connect CPU stations
- STEP 7-Micro/WIN "System Block" window configures CPU parameters
- A System Block download operation transfers the new hardware configuration into the target CPU
Hardware configuration S7-1200

- STEP 7 Basic uses a visual configuration where you create an image of your actual hardware set
- Hardware modules are selected from a hardware catalog tree and dragged into a rack image
- After your system hardware image is assembled, use a mouse click on an item in the system image to set the configuration “Properties” page for the selected hardware item
- Select the CPU PROFINET connector on the CPU image to set the IP address properties
- Use the Download command to transfer the new hardware configuration to the target CPU. When prompted, select a network interface and CPU.

Password protection

- S7-1200 Off-line project block password provides Know-How protection to prevent unauthorized access to one or more of your code blocks (OB, FB, FC, or DB).
- S7-1200 On-line CPU password protection provides 3 levels of security for restricting access to CPU functions.

I/O address assignment

- S7-200: I/O addresses automatically fixed by CPU operating system according to module location.
- S7-1200: Default I/O assignment can be modified by Device configuration properties.
Serial communication for S7-200 and S7-1200

S7-1200 CPU communication via RS232 and RS485 connections
- ASCII protocol (character based serial communication) uses STEP 7 Basic point-to-point (PtP) instructions
- USS Drive protocol is programmed with STEP 7 Basic USS Library instructions
- MODBUS protocol is programmed with STEP 7 Basic MODBUS Library instructions

S7-200 CPUs have 1 or 2 on-board RS485 serial connections
S7-1200 CPUs have 1 on-board PROFINET (ETHERNET) connection. Use the RS232 and RS485 modules for PtP communication

- PROFIBUS master (RS485 connection): The PROFIBUS master/slave functionality is not available in the initial S7-1200 product release.
- MODBUS RTU is possible on both the RS485 and RS232 signal modules.
- USS library is supported on the RS485 port. The libraries are included with STEP 7 Basic.
- S7-1200 SINAULT: You can create an S7-1200 RTU application using the RS232 module, PtP communication, and the existing SINAUT MD720-3 GSM/GPRS wireless modem. A new SINAUT solution and teleservice adapter are planned for the next S7-1200 hardware release.
- The RS232 module supports handshaking.
- The S7-1200 RS232 and RS485 modules have electrically isolated ports.

**S7-1200 Communication Modules**

| CM 1241 RS232 |
| CM 1241 RS485 |
The PROFINET port on the CPU supports simultaneous communication connections:

- 3 connections for HMI to CPU communication
- 1 connection for programming device (PG) to CPU communication
- 8 connections for S7-1200 program communication using the T-block instructions (TSEND_C, TRCV_C, TCON, TDISCON, TSEND TRCV)
- 3 connections for a passive S7-1200 CPU communicating with an active S7 CPU. The active S7 CPU uses GET and PUT instructions (S7-300 and S7-400) or ETHx_XFER instructions (S7-200). An active S7-1200 communication connection is only possible with the T-block instructions.

MODBUS-TCP is not available for the initial S7-1200 release. However, the S7-1200 has "Native" Ethernet TCP/IP protocol available (T-block instructions) for custom development.

OPC server (Object Linking and Embedding – OLE) for Process Control server functionality is possible using the SIMATIC NET OPC Server.

PROFINET functionality including controller, device, and CBA is not available for the initial release of the S7-1200.

The S7-1200 Ethernet interfaces are designated as PROFINET. The S7-1200 will not support PROFINET I/O in the initial release, but this functionality is planned for a future release.

Communication with Omron and Mitsubishi controllers via the Ethernet. The S7-1200 has "Native" Ethernet TCP/IP protocol available ("FreePort" for Ethernet) for custom development of this functionality. It is possible to communicate with 3rd party PLCs as long as they support the same open Ethernet connectivity as the S7-1200.
S7-1200 CPUs use a PROFINET connection to STEP 7 Basic, S7-1200 CPUs, and HMI panels

HMI general
- The HMI Basic Panel in the context of the S7-1200 can communicate with up to 4 CPUs.
- Text Displays for the S7-1200 are planned in a future release.
- Compatibility with current HMI Ethernet devices
  At present, only the Basic Panels have been system-tested and released in conjunction with WinCC Basic and the S7-1200. However, other panels can also be connected to the S7-1200 in the context of WinCC flexible.
- MP277 and 377 panels will talk with the S7-1200. Use WinCC flexible for programming and select the S7-300 communication channel (Rack 0 - Slot 0) at the start.

HMI update rate
- S7-200: HMI data update occurs at end of the program scan and is scan rate limited.
- S7-1200: HMI data update occurs asynchronously during program scan. Therefore, insure that data variables are buffered from change, during a program scan.
Totally Integrated Automation Portal integrates control logic and HMI configuration programming

- HMI Basic panels require a PC for download. They do not have memory cards.
- Variable tags on HMI basic panels. The number of variable tags cannot be increased at this time. There are plans to increase the number of default tags available on these units.
- WinCC Flexible Micro will be available as long as HMI Micro panels (for the S7-200) are sold. At this time there are no plans for discontinuing the micro panel.
- Library graphics are created in WinCC Flexible.
- It is not possible to migrate any library items directly from WinCC Flexible to WinCC Basic. However, you can copy all the elements from the library to an HMI screen in a project and then migrate the resulting project into WinCC Basic.
- Changing from WinCC flexible to WinCC Basic. A firmware update is not necessary in the case of WinCC flexible 2008 and WinCC Basic.
- Sm@RtAccess/Sm@rtService
- The S7-1200 does not support Sm@RtAccess or Sm@rtService

The Totally Integrated Automation Portal provides the tools for managing and configuring all of the devices in your project, such as PLCs and HMI devices. As a component of the TIA Portal, STEP 7 Basic provides two programming languages (LAD and FBD). The TIA Portal also provides the tools for creating and configuring the HMI devices in your project.

<table>
<thead>
<tr>
<th>S7-1200 HMI devices</th>
</tr>
</thead>
<tbody>
<tr>
<td>KTP400 Basic mono PN, 3.8&quot; STN Gray scale Touch-Display, 4 Function keys, Ethernet interface</td>
</tr>
<tr>
<td>KTP600 Basic mono PN, 5.7&quot; STN Gray scale Touch-Display, 6 Function keys, Ethernet interface</td>
</tr>
<tr>
<td>KTP600 Basic color DP and Basic color PN, 5.7&quot; TFT color Touch-Display, 6 Function keys, PROFIBUS-DP / MPI-interface or Ethernet interface</td>
</tr>
<tr>
<td>KTP1000 Basic color DP and Basic color PN, 10.4&quot; TFT color Touch-Display, 8 Function keys, PROFIBUS-DP / MPI-interface or Ethernet interface</td>
</tr>
<tr>
<td>TP1500 Basic color PN, 15.0&quot; TFT color Touch-Display, Ethernet interface</td>
</tr>
</tbody>
</table>
CPU work memory size

<table>
<thead>
<tr>
<th>CPU 224XP</th>
<th>CPU 1214C</th>
</tr>
</thead>
<tbody>
<tr>
<td>50 kB</td>
<td>64 kB</td>
</tr>
<tr>
<td>26 kB</td>
<td></td>
</tr>
</tbody>
</table>

- **Fixed**
- **Floating**

- **Resources**
- **Technology**
- **Communication**
- **HMI**
- **Memory**
- **Block concept**
- **Instruction set**
- **New data types**
- **Timers**
- **Counters**

**Instruction set**

**New data types**
STEP 7 Basic program objects in S7-1200 CPU memory

Program Tag names and comments are stored in S7-1200 CPU memory and are available on-line. For STEP 7-Micro/WIN and the S7-200, you must have the original project files to match symbolic tag names and comments with the on-line program logic.

All tags, block comments, network comments, and instruction comments are downloaded to the S7-1200 controller. This makes it possible to go online with a controller and debug it without the original project.
Step 7 Basic can optimize a Data Block size

If the data type declarations are mixed, then memory space is wasted.

Data types are organized to save memory space.

When a Data Block is created, you can select the optimized format by checking the "Symbolic access only" box. The "Classic" data format is also supported for compatibility with legacy code.
Retentive memory assignments preserve data during CPU power interruptions

A “Symbolic access only” DB can select individual data elements for retention. If “Symbolic access only” is not selected, then a block of DB data can be retentive. The 2048 byte total is shared between M and DB memory.

The S7-1200 CPU automatically stores retained data in internal flash memory. The S7-200 uses a supercap, optional battery cartridge, or programmed writes to a flash memory cartridge to retain memory data.

### System block settings can assign 6 retentive ranges in V, T and C actual values, or M

<table>
<thead>
<tr>
<th>Name</th>
<th>Data type</th>
<th>Initial value</th>
<th>Retain</th>
</tr>
</thead>
<tbody>
<tr>
<td>Symbolic name 1</td>
<td>Byte</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Symbolic name 2</td>
<td>Byte</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Symbolic name 3</td>
<td>Real</td>
<td>3.141592654</td>
<td></td>
</tr>
<tr>
<td>Symbolic name 4</td>
<td>Byte</td>
<td>1.5</td>
<td></td>
</tr>
<tr>
<td>Symbolic name 5</td>
<td>Real</td>
<td>2.831592654</td>
<td></td>
</tr>
</tbody>
</table>

The PLC tag table button can assign a range of M memory as retentive (2048 byte maximum).
Memory Card - Memory size

- S7-200 MC two sizes
- S7-1200 MC two sizes

- 1MB
- 10MB
- 10MB
- 100MB

- 64 KB
- 256 KB
- 2 MB

- 2 MB
- 24 MB
What can be stored on the Memory Card?

<table>
<thead>
<tr>
<th></th>
<th>S7-200</th>
<th>S7-1200</th>
</tr>
</thead>
<tbody>
<tr>
<td>Program</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Data</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>System data</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Recipes</td>
<td>✓</td>
<td>planned</td>
</tr>
<tr>
<td>Data Log</td>
<td>✓</td>
<td>planned</td>
</tr>
<tr>
<td>Files</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Projects</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

SIMATIC memory cards have a Windows file system and comply with the necessary industrial requirements. The memory card can be written to and read in any PC and then also used again for operation in the CPUs.
**How can the S7-1200 use the pre-formatted SIMATIC MC?**

- You can create a “Program” card that functions as CPU memory. If you remove the “Program” card from the CPU, the CPU loses all project memory.

- You can create a “Transfer” card that can copy your project (and updates) to multiple CPUs.

You can create a “Transfer” card using STEP 7 Basic on a PC with a standard SD card reader/writer.

**Memory card usage**

- Before programming a memory card, be sure there is a valid network configuration in your project so that it can connect to the PLC after installing the card.

- SIMATIC memory cards are pre-formatted with a SIMATIC memory format that must be preserved. Do not use a PC to delete the two hidden files __log__ (system file) and crdinfo.bin (bin file). Do not use a PC to reformat the memory card or the card will become unusable.

- Refer to the S7-1200 Programmable controller system Manual 11/2009 version for details on how to create and use a “Program” card and “Transfer” card.
S7-1200 Organization Block (OB) Types:
- Program Cycle OB1
- Startup OB100 series
- Time delay interrupt OB200 series
- Cyclic interrupt OB200 series
- Hardware interrupt OB200 series
- Time error interrupt OB80
- Diagnostic error interrupt OB82

Non-fatal error handing
- S7-200: By default, continue RUN mode
- S7-1200: By default, go to STOP mode
  If OB80 or OB82 error OB blocks exist in your program, then continue RUN mode.
  OB80 and OB82 may be empty or contain your programmed error reaction

STEP 7 Basic programming methods not supported initial V 10.5 release
- Hot keys
- Indirect addressing
- Run-time edit
The maximum nesting depth for subroutines from the Main is 8, and from an Interrupt routine is 1.
STEP 7 Basic has the same block architecture used by the S7-300

- Modularization and reuse is easier
- Technology objects (for example, PID control) can be standardized and called multiple times
- Symbolic reference is possible

The maximum nesting depth is 16
S7-1200 Instance Data Block types

STEP 7 Basic can use single-instance and multi-instance Data Blocks
- A function block (FB) can be called multiple times
- One FB type (for example, FB "Motor") can control several drives
- The actual data of the different drives can be stored in different single-instance or multi-instance DBs
- Two FBs can share a multi-instance DB for more efficient memory usage
Interrupt events are attached and detached to interrupt routines at program run-time only.
Several optional Startup and Time-delay OBs may be used.

Attachment of event to OB can be made at configuration-time and run-time.
**What’s different?**

**Set/Reset instructions**
- **S7-200**: S (Set) and R (Reset)
- **S7-1200**: S (Set) and R (Reset) for single point, SET_BF (Set Bit Field) and RESET_BF (Reset Bit Field) for multiple points

**Immediate instructions**
- **S7-200**: I (Immediate), SI (Set Immediate), and RI (Reset Immediate)
- **S7-1200**: Direct (Immediate) peripheral address (example, Q0.0:P or I0.0:P)

**Edge Detection instructions**
- **S7-200**: P (Positive Transition) and N (Negative Transition)
- **S7-1200**: P_TRIG (Positive Transition detector) and N_TRIG (Negative Transition detector)
**What’s new?**
TP (Pulse timer) and TONR (On-delay retentive) are now in the same group.

**What’s different?**
- **S7-200:** A timer number selection sets a 1ms, 10ms, or 100ms time resolution that is multiplied by the WORD size time current value for preset and elapsed times.
- **S7-1200:** All timers are 1ms timers that use a new DWORD size Time data type for the preset and elapsed time values.
- **S7-200:** STEP 7 Micro/WIN has a SIMATIC and IEC programming mode. In SIMATIC mode, a T-bit and T-current value corresponding to the timer number are used to signal the timeout condition and the current value (elapsed time).
- **S7-1200:** In STEP 7 Basic, all timers are IEC style timers that have a Q output bit that signals the timeout condition and an ET output that gives the elapsed time.
- **S7-200:** BGN_ITIME (Begin Interval Time), CAL_ITIME (Calculate Interval Time)
- **S7-1200:** Use Clock instructions RD_SYS_T (read system time) and T_SUB (time difference) to calculate time intervals.
**What’s different?**

**Counter instructions**

- S7-200: STEP 7 Micro/WIN has a SIMATIC and IEC programming mode. In SIMATIC mode, a C-bit and C-current value corresponding to the counter number are used to signal the count condition and the current count value.
- S7-1200: In STEP 7 Basic, all counters are IEC style counters that have a Q output bit that signals the count condition and a CV output that gives the current count value.

**High-Speed Counter instructions**

- S7-200: HDEF (High-Speed Counter Definition) HSC (High-Speed Counter)
- S7-1200: CTRL_HSC

- S7-200: Special memory assignments (SM addresses) are used to set the high-speed counter configuration and operating parameters.
- S7-1200: High-speed counters are set up in the PLC Device configuration properties. The operating parameters are available as inputs and outputs of the CTRL_HSC instruction.
**Compare instructions**

**What's new?**

The Compare instructions now support the 64 bit LREAL data type.

**Compare instructions:**
- **IN_RANGE** tests whether an input value is in a specified value range.
- **OUT_RANGE** tests whether an input value is out of a specified value range.
- **[OK]** tests whether an input data reference is a real number.
- **NOT_OK** instruction tests whether an input data reference is not a real number.

**What's different?**
- **S7-200**: The instruction name determines the data type.
- **S7-1200**: The data type is selected after placing the instruction.
Math instructions

What’s new?
The S7-1200 Floating Point Math instructions now support the 64 bit LREAL data type.

What’s different?

Math instructions

- **S7-200:** The instruction selection determines the data type.
- **S7-1200:** The data type is selected after placing the instruction.
Move instructions

What's new?
The MOVE instructions now support the 64 bit LREAL data type.

New Move instructions
- UMOVE_BLK (uninterruptible move block)
- UFILL_BLK (uninterruptible fill block)

What's different?

Move instructions
- S7-200: The instruction name determines the data type.
- S7-1200: The data type is selected after placing the instruction.
Convert instructions

What’s new?
The Convert instructions (except SCALE_X and NORM_X) now support the 64 bit LREAL data type.

New Convert instructions
- CEIL (ceiling) converts a real number to the next higher integer
- FLOOR converts a real number to the next lower integer
- SCALE_X scales a normalized real parameter value
- NORM_X normalizes a parameter value

What’s different?
- S7-200: The instruction name determines the data type.
- S7-1200: The data type is selected after placing the instruction.

String Convert instructions
- S7-200: S_I, S_DI, S_R, I_S, DI_S, R_S, ITA, DTA, and RTA
- S7-1200: S_CONV, STRG_VAL, VAL_STRG

S7-200 conversions instructions ATH, HTA, and SEG are not supported by the S7-1200.
# Program Control instructions

## What's new?
- **JMPN**: Jump if no power flow to a JMP coil
- **GetError**: Get information about program block execution errors
- **GetErrorID**: Get the ID of an execution error

## What's different?
### Re-Trigger Cycle Time Monitoring
- **S7-200**: WDR Watchdog reset
- **S7-1200**: RE_TRIGR

### Terminate execution of current block
- **S7-200**: END/RET
- **S7-1200**: RET

### Execution control
- **S7-200** FOR-NEXT loop instructions are not supported on the S7-1200. This function must be created from Jump, Add, and Compare instructions.
- **Sequence Control Relay instructions (SCR, SORT, SCRE)** are not supported in the S7-1200.
- **The DIAG_LED Diagnostic LED instruction is not supported in the S7-1200.**
Logical operation instructions

What's new?
- SEL select instruction selects one of two inputs
- MUX multiplex instruction selects one of multiple inputs

What's different?
- S7-200: The instruction selection determines the data type.
- S7-1200: The data type is selected after placing the instruction.

AND instruction
- S7-200: WAND_B, WAND_W, WAND_DW
- S7-1200: AND

OR instruction
- S7-200: WOR_B, WOR_W, WOR_DW
- S7-1200: OR

XOR instructions
- S7-200: WXOR_B, WXOR_W, WXOR_DW
- S7-1200: XOR

Invert instruction
- S7-200: INV_B, INV_W, INV_DW
- S7-1200: INVERT
Shift + Rotate instructions

What's different?

- S7-200: The instruction selection determines the data type.
- S7-1200: The data type is selected after placing the instruction.

Shift Right instruction
- S7-200: SHR_B, SHR_W, SHR_DW
- S7-1200: SHR

Shift Left instruction
- S7-200: SHL_B, SHL_W, SHL_DW
- S7-1200: SHL

Rotate Right instructions
- S7-200: ROR_B, ROR_W, ROR_DW
- S7-1200: ROR

Rotate Left instructions
- S7-200: ROL_B, ROL_W, ROL_DW
- S7-1200: ROL
Clock instructions

What’s new?
New TIME and DTL (Date and time long) data types

- T_CONV converts data type of time value
- T_ADD adds TIME and DTL values
- T_SUB subtracts TIME and DTL values
- T_DIFF provides difference
- RD_LOC_T read local time

What’s different?

Write system time instructions
- S7-200: SET_RTC, SET_RTCX
- S7-1200: WR_SYS_T (write system time)

Read system time instructions
- S7-200: READ_RTC, READ_RTCX
- S7-1200: RD_SYS_T (read system time)
## String instructions

### What's new?

**String data format**
- S7-200: Length byte followed by character bytes
- S7-1200: Maximum length byte followed by the actual length byte and character bytes

**New String operations**
- LEFT, RIGHT, DELETE, INSERT, REPLACE, VAL_STRG, STRG_VAL, and S_CONV

### What's different?

- S7-200: STR_LEN, STR_CAT, SSTR_CPY
- S7-1200: LEN, CONCAT, MID

**FIND substring or character in string**
- S7-200: STR_FIND, CHR_FIND
- S7-1200: FIND

**Copy string**
- S7-200: STR_CPY
- S7-1200: S_CONV (string convert) can have a STRING data type as input and output to "copy" the string.
### Communications instructions

#### What's different?

**PPI network communication**
- **S7-200**: The integrated PPI network RS485 port is programmed with the NETR, NETW, GET_ADDR, and SET_ADDR instructions.
- **S7-1200**: The integrated CPU port is now an Ethernet connection. The S7-200 PPI network instructions are not supported in STEP 7 Basic version 10.5.

**Point-to-Point (PtP) instructions** (Freeport communications)
- **S7-200**: XMT, RCV
- **S7-1200**: PORT_CFG, SEND_CFG, RCV_CFG, SEND_PTP, RCV_PTP, RCV_RST, SGN_GET, and SGN_SET.

**Ethernet communication**
- **S7-200**: The Ethernet wizard generates subroutines that operate the Ethernet module.
- **S7-1200**: The Ethernet wizard generates subroutines that operate the Ethernet module.

**Valid Ethernet program to program communication paths**
- An S7-200 executes ETH0_CTRL / ETH0_XFR subroutines to read from or write to a passive S7-1200.
- An S7-300/S7-400 executes GET/PUT to read from or write to a passive S7-1200.
- For an S7-1200 connected to a multiple S7-1200/S7-300/S7-400 network, all partner programs can execute T-block instructions to read from and write to each other.
Interrupt instructions

What's different?

<table>
<thead>
<tr>
<th>Event Interrupt</th>
<th>S7-200</th>
<th>S7-1200</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ATCH and DTCH events 0-7</td>
<td>Device configuration property assignment</td>
</tr>
<tr>
<td>Time Delay Interrupt</td>
<td>ATCH and DTCH (time delay interrupt events 21 and 22)</td>
<td>SRT_DINT, CAN_DINT</td>
</tr>
<tr>
<td>Asynchronous Interrupt Control</td>
<td>ENI and DISI</td>
<td>DIS_AIRT, EN_AIRT</td>
</tr>
<tr>
<td>Terminate Execution of current interrupt block</td>
<td>RETI</td>
<td>RET</td>
</tr>
<tr>
<td>Clear queued Interrupt events</td>
<td>CLR_EVENT</td>
<td>Current and queued events are cleared by the DETACH instruction</td>
</tr>
</tbody>
</table>

Interrupts:
- ATCH
- DETACH

Time delay interrupt:
- SRT_DINT
- CAN_DINT

Asynchronous event:
- DIS_AIRT
- EN_AIRT

Technology
- Timers
- Counters
- Memory
- HMI
- Communication
- Hardware
- Instruction set
- New data types
- Block concept
- Resources

Page 36/48
### Table and PID Loop control instructions

#### What's different?

<table>
<thead>
<tr>
<th>PID instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>S7-200:</strong> The PID instruction is usually operated with code generated by the STEP 7-Micro/WIN PID wizard.</td>
</tr>
<tr>
<td><strong>S7-1200:</strong> PID_Compact smart instruction</td>
</tr>
</tbody>
</table>

#### Table instructions

- The S7-200 table instructions are not supported in STEP 7 Basic version 10.5.

#### PID Loop control

<table>
<thead>
<tr>
<th>S7-200</th>
<th>S7-1200</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Diagram showing PID]</td>
<td>[PID_Compact]</td>
</tr>
</tbody>
</table>

#### Table

<table>
<thead>
<tr>
<th>S7-200</th>
</tr>
</thead>
<tbody>
<tr>
<td>[Diagram showing table]</td>
</tr>
</tbody>
</table>

---

The content provided is a screenshot of a page from a Siemens manual, which discusses differences between S7-200 and S7-1200 in terms of PID control instructions and table instructions. The page is part of a transition manual, emphasizing compatibility and differences between the two systems, particularly focusing on PID loop control and table instructions.
Basic Motion and Pulse instructions

What's different?

Motion control instructions
- S7-200: The EM253 Position module wizard creates subroutines that can be called from your program.
- S7-1200: PLC Open motion control instructions
  - S7-200: The motion subroutines are level triggered. An extra edge instruction is required for the trigger signal, if called from the multi-scan Main routine instead of from a single-scan interrupt routine.
  - S7-1200: The motion instructions are internally rising edge-triggered. This is not a problem for a multi-scan program block (OB1 for example). However, if a motion instruction is placed in a single-scan interrupt OB, then the instruction must be executed twice in succession to provide an edge signal. If an external motion event triggers a single-scan interrupt block, then execute the motion instruction once with the enable request parameter assigned a constant “1” and once with a constant “0”.

Pulse instructions
- S7-200: The PTO/PWM wizard creates subroutines that can be called from your program. Special memory assignments (SM addresses) are used to set the pulse generator configuration and operating parameters.
- S7-1200: Pulse generators are set up in the PLC Device configuration properties. The operating parameters are available as outputs as defined in the configuration properties. The CTRL_PWM instruction starts and stops the pulse generator.
Modbus library instructions

What's different?

S7-200 Modbus library simplified

- MB_COMM_LOAD provides first pass initialization for master and slave operations.
- MB_MASTER and MB_SLAVE control the message and port assignment.

S7-1200 Modbus library

- MB_COMM_LOAD provides first pass initialization for master and slave operations.
- MB_MASTER and MB_SLAVE control the message and port assignment.
**USS drive library instructions**

What’s different?
- **USS_DRV** and **USS_PORT** replace **USS_INT** and **USS_CTRL**.
- **S7-200**: Two fixed PZD parameters (control and speed)
- **S7-1200**: Added up to eight user-defined PZD parameters
- **S7-200**: Update rate fixed (as fast as possible)
- **S7-1200**: Place in a cyclic interrupt OB for user-defined update rate.

One Read instruction
- **S7-200**: USS_RPM_W_P1, USS_RPM_D_P1, USS_RPM_R_P1
- **S7-1200**: USS_RPM

One Write instruction
- **S7-200**: USS_WPM_W_P1, USS_WPM_D_P1, USS_WPM_R_P1
- **S7-1200**: USS_WPM

**USS drive library**

<table>
<thead>
<tr>
<th>S7-200</th>
<th>S7-1200</th>
</tr>
</thead>
<tbody>
<tr>
<td>USS_DRV</td>
<td>USS_DRV</td>
</tr>
<tr>
<td>USS_PORT</td>
<td>USS_CTRL</td>
</tr>
<tr>
<td>USS_RPM_W_P1</td>
<td>USS_CTRL_P1</td>
</tr>
<tr>
<td>USS_RPM_D_P1</td>
<td>USS_RPM_D_P1</td>
</tr>
<tr>
<td>USS_RPM_R_P1</td>
<td>USS_RPM_R_P1</td>
</tr>
<tr>
<td>USS_WPM_W_P1</td>
<td>USS_WPM_D_P1</td>
</tr>
<tr>
<td>USS_WPM_R_P1</td>
<td>USS_WPM_R_P1</td>
</tr>
<tr>
<td>USS_RPM</td>
<td>USS_WPM</td>
</tr>
</tbody>
</table>
New elementary data types for the S7-1200

**Short integer data types can save resources**
- **Sint** - Byte size (-128 to 127)

**Unsigned data types increase the positive range**
- **USint** - Byte size (0 to 255)
- **Uln** - Word size (0 to 65,535)
- **UDint** - Dword size (0 to 4,294,967,295)

**Long Real for greater floating-point precision**
- **LReal** - 64 bit size
  - $ \pm 2.2250738585072020 \times 10^{-308} $ to $ \pm 1.7976931348623157 \times 10^{308} $

**Time data type has new entry format and range**
- **Time** - Dword size
  - $ \# -24d_{-20h_{-31m_{-23s_{-648ms}}}} $ to $ \# 24d_{_{20h_{_{31m_{_{23s_{_{647ms}}}}}}} $ stored as $ -2,147,483,648 $ ms to $ +2,147,483,647 $ ms
  - **ex.** $ \# 50ms $  
  - $ \# 5m_{_{30s}} $  
  - $ \# 1d_{_{2h_{_{15m_{_{30s_{_{45ms}}}}}}}} $
New complex data types for the S7-1200

Array

<table>
<thead>
<tr>
<th>Name</th>
<th>Data type</th>
<th>Initial value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Static</td>
<td>Byte</td>
<td>Byte</td>
</tr>
<tr>
<td>Element1</td>
<td>Byte</td>
<td>Byte</td>
</tr>
<tr>
<td>Element2</td>
<td>Byte</td>
<td>Byte</td>
</tr>
<tr>
<td>Element3</td>
<td>Byte</td>
<td>Byte</td>
</tr>
<tr>
<td>Element4</td>
<td>Byte</td>
<td>Byte</td>
</tr>
<tr>
<td>Element5</td>
<td>Byte</td>
<td>Byte</td>
</tr>
<tr>
<td>Element6</td>
<td>Byte</td>
<td>Byte</td>
</tr>
<tr>
<td>Element7</td>
<td>Byte</td>
<td>Byte</td>
</tr>
<tr>
<td>Element8</td>
<td>Byte</td>
<td>Byte</td>
</tr>
<tr>
<td>Element9</td>
<td>Byte</td>
<td>Byte</td>
</tr>
</tbody>
</table>

DTL (Date and Time Long)

<table>
<thead>
<tr>
<th>Name</th>
<th>Data type</th>
<th>Initial value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Static</td>
<td>Byte</td>
<td>Byte</td>
</tr>
<tr>
<td>Element1</td>
<td>Byte</td>
<td>Byte</td>
</tr>
<tr>
<td>Element2</td>
<td>Byte</td>
<td>Byte</td>
</tr>
<tr>
<td>Element3</td>
<td>Byte</td>
<td>Byte</td>
</tr>
<tr>
<td>Element4</td>
<td>Byte</td>
<td>Byte</td>
</tr>
<tr>
<td>Element5</td>
<td>Byte</td>
<td>Byte</td>
</tr>
<tr>
<td>Element6</td>
<td>Byte</td>
<td>Byte</td>
</tr>
<tr>
<td>Element7</td>
<td>Byte</td>
<td>Byte</td>
</tr>
<tr>
<td>Element8</td>
<td>Byte</td>
<td>Byte</td>
</tr>
</tbody>
</table>

Symbolic access is available to all elements of an Array or DTL structure.
Timer operation
S7-200 and S7-1200

For STEP 7 Basic, all timers are 1 ms and the time value can be entered directly. For STEP 7-Micro/WIN, the preset time and current time are entered as a number to be multiplied by a time base (1 / 10 / 100 ms) that depends on the selected timer number.

Timer bit T37 = 1 when current value >= PT.

Output Q = 1 when elapsed time ET >= PT.
Counter operation
S7-200 and S7-1200

**S7-200**
- SIMATIC type counter
  - Range: 0 – 32767
  - Counter bit C5 = 1 when current value >= PV

**S7-1200**
- IEC type counter
  - Range: selectable data type from SINT to UDINT
  - Output Q = 1 when current value >= PV
Count and measure with high-speed counters (HSC)

The two-phase or quadrature encoder is the most widely used of all rotary encoders due to better precision.

<table>
<thead>
<tr>
<th>HSC devices total</th>
<th>S7-200 CPU 224XP</th>
<th>S7-1200 CPU 1214C</th>
</tr>
</thead>
<tbody>
<tr>
<td>6 single-phase or</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>4 two-phase</td>
<td></td>
<td></td>
</tr>
<tr>
<td>100kHz. max. single-phase or</td>
<td>2 or</td>
<td>3</td>
</tr>
<tr>
<td>80kHz. max. two-phase</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>30kHz. max. single-phase or</td>
<td>4 or</td>
<td>3</td>
</tr>
<tr>
<td>20kHz. max. two-phase</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>
### S7-1200 Axis Technology Object

- **S7-200** and **STEP 7 Micro/WIN** use a Position module and motion control wizard for motion control. You must rerun the wizard to make configuration changes.

- **S7-1200** and **STEP 7 Basic** use integrated pulse outputs and Axis Technology Object configuration for control of stepper motors and servo drives. PLCopen standard instructions are then placed in your program.

#### Resources

<table>
<thead>
<tr>
<th>MC_Force</th>
<th>MC_Intert</th>
<th>MC_Home</th>
<th>MC_IHalt</th>
</tr>
</thead>
<tbody>
<tr>
<td>EN</td>
<td>EN</td>
<td>EN</td>
<td>EN</td>
</tr>
<tr>
<td>Ax</td>
<td>Ax</td>
<td>Ax</td>
<td>Ax</td>
</tr>
<tr>
<td>Done</td>
<td>Done</td>
<td>Done</td>
<td>Done</td>
</tr>
<tr>
<td>Error</td>
<td>Error</td>
<td>Error</td>
<td>Error</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>MC_MoveAbsolute</th>
<th>MC_MoveRelative</th>
<th>MC_MoveVelocity</th>
<th>MC_MoveLog</th>
</tr>
</thead>
<tbody>
<tr>
<td>EN</td>
<td>EN</td>
<td>EN</td>
<td>EN</td>
</tr>
<tr>
<td>Ax</td>
<td>Ax</td>
<td>Ax</td>
<td>Ax</td>
</tr>
<tr>
<td>Done</td>
<td>Done</td>
<td>Done</td>
<td>Done</td>
</tr>
<tr>
<td>Error</td>
<td>Error</td>
<td>Error</td>
<td>Error</td>
</tr>
<tr>
<td>Position</td>
<td>Execute</td>
<td>Velocity</td>
<td>Current</td>
</tr>
<tr>
<td>_</td>
<td>_</td>
<td>_</td>
<td>_</td>
</tr>
</tbody>
</table>

- S7-200 and STEP 7 Micro/WIN use a PID wizard and PID tune control panel tool for controlling up to eight PID loops.
- S7-1200 and STEP 7 Basic use the PID controller Technology Object configuration for up to 16 PID control loops. The PID_Compact instruction is then placed in your program.
Refer to the SIMATIC S7-1200 and S7-200 documentation on the Internet at:
http://www.siemens.com/automation/service&support

Contact your Siemens distributor or sales office for assistance in answering any technical questions, for training, or for ordering S7 products.