U. S. DEPARTMENT OF COMMERCE
Patent and Trademark Office

CLASSIFICATION ORDER 1863

JUNE 5, 2007

Project No. Y-7133

**The following classification changes will be effected by this order:**

| | **Class** | **Subclass** | **Art Unit** | **Ex'r Search Room No.** |
|---|---|---|---|---|
| **Abolished:** | None | | | |
| **Established:** | 712 | E9.001-E9.009,<br>E9.01, E9.011-E9.019,<br>E9.02, E9.021-E9.029,<br>E9.03, E9.031-E9.039,<br>E9.04, E9.041-E9.049,<br>E9.05, E9.051-E9.059,<br>E9.06, E9.061-E9.069,<br>E9.07, E9.071-E9.079,<br>E9.08, E9.081-E9.086 | 2183 | ELEL0000 |

**No other classes were impacted by this order.**

**This order includes the following:**

A. CLASSIFICATION MANUAL CHANGES,

D. DEFINITION CHANGES AND NEW OR ADDITIONAL DEFINITIONS.

CLASSIFICATION ORDER 1863

JUNE 5, 2007

Project No. Y-7133

**Project Leader:** Yen M. Nguyen

**Editor:** James E. Doyle, Jr.

**Editorial Assistant:** Louise Bogans

CLASS 712 ELECTRICAL COMPUTERS AND DIGITAL PROCESSING SYSTEMS: PROCESSING ARCHITECTURES AND INSTRUCTION PROCESSING (E.G., PROCESSORS)

JUNE 2007

1 PROCESSING ARCHITECTURE
2 .Vector processor
3 ..Scalar/vector processor interface
4 ..Distributing of vector data to vector registers
5 ...Masking to control an access to data in vector register
6 ..Controlling access to external vector data
7 ..Vector processor operation
8 ...Sequential
9 ...Concurrent
10 .Array processor
11 ..Array processor element interconnection
12 ...Cube or hypercube
13 ...Partitioning
14 ...Processing element memory
15 ...Reconfiguring
16 ..Array processor operation
17 ...Application specific
18 ...Data flow array processor
19 ...Systolic array processor
20 ...Multimode (e.g., MIMD to SIMD, etc.)
21 ...Multiple instruction, Multiple data (MIMD)
22 ...Single instruction, multiple data (SIMD)
23 .Superscalar
24 .Long instruction word
25 .Data driven or demand driven processor
26 ..Detection/pairing based on destination, ID tag, or data
27 ..Particular data driven memory structure
28 .Distributed processing system
29 ..Interface
30 ..Operation
31 ...Master/slave
32 .Microprocessor or multichip or multimodule processor having sequential program control
33 ..Having multiple internal buses
34 ..Including coprocessor
35 ...Digital Signal processor
36 ..Application specific
37 ..Programmable (e.g., EPROM)
38 ..Offchip interface
39 ...Externally controlled internal mode switching via pin
40 ...External sync or interrupt signal
41 ..RISC
42 ..Operation
43 ...Mode switching
200 ARCHITECTURE BASED INSTRUCTION PROCESSING
201 .Data flow based system
202 .Stack based computer
203 .Multiprocessor instruction
204 INSTRUCTION ALIGNMENT
205 INSTRUCTION FETCHING
206 .Of multiple instructions simultaneously
207 .Prefetching
208 INSTRUCTION DECODING (E.G., BY MICROINSTRUCTION, START ADDRESS GENERATOR, HARDWIRED)
209 .Decoding instruction to accommodate plural instruction interpretations (e.g., different dialects, languages, emulation, etc.)
210 .Decoding instruction to accommodate variable length instruction or operand
211 .Decoding instruction to generate an address of a microroutine
212 .Decoding by plural parallel decoders
213 .Predecoding of instruction component
214 INSTRUCTION ISSUING
215 .Simultaneous issuance of multiple instructions
216 DYNAMIC INSTRUCTION DEPENDENCY CHECKING, MONITORING OR CONFLICT RESOLUTION
217 .Scoreboarding, reservation station, or aliasing
218 .Commitment control or register bypass
219 .Reducing an impact of a stall or pipeline bubble
220 PROCESSING CONTROL
221 .Arithmetic operation instruction processing
222 ..Floating point or vector
223 .Logic operation instruction processing
224 ..Masking
225 .Processing control for data transfer
226 .Instruction modification based on condition
227 .Specialized instruction processing in support of testing, debugging, emulation
228 .Context preserving (e.g., context swapping, checkpointing, register windowing
229 .Mode switch or change
230 .Generating next microinstruction address
231 .Detecting end or completion of microprogram
232 .Hardwired controller
233 .Branching (e.g., delayed branch, loop control, branch predict, interrupt)
234 ..Conditional branching
235 ...Simultaneous parallel fetching or executing of both branch and fall-through path
236 ...Evaluation of multiple conditions or multiway branching
237 ...Prefetching a branch target (i.e., look ahead)
238 ....Branch target buffer
239 ...Branch prediction
240 ....History table
241 ..Loop execution
242 ..To macro-instruction routine
243 ..To microinstruction subroutine

PROCESSING CONTROL

.Branching (e.g., delayed branch, loop control, branch predict, interrupt)

244 ..Exeception processing (e.g., interrupts and traps)

245 .Processing sequence control (i.e., microsequencing)

246 ..Plural microsequencers (e.g., dual microsequencers)

247 ..Multilevel microcontroller (e.g., dual-level control store)

248 ..Writable/changeable control store architecture

300 BYTE-WORD REARRANGING, BIT-FIELD INSERTION OR EXTRACTION, STRING LENGTH DETECTING, OR SEQUENCE DETECTING

E-SUBCLASSES

The folllowing subclasses beginning with the letter E are E-subclasses. Each E-subclass corresponds in scope to a classification in a foreign classification system, for example, the European Classification system (ECLA). The foreign classification equivalent to an E-subclass is identified in the subclass definition. In addition to US documents classified in E-subclasses by US examiners, documents are regularly classified in E-subclasses according to the classification practices of any foreign Offices identified in parentheses at the end of the title. For example, "(EPO)" at the end of a title indicates both European and US patent documents, as classified by the EPO, are regularly added to the subclass. E-subclasses may contain subject matter outside the scope of this class. Consult their definitions, or the documents themselves, to clarify or interpret titles.

* E9.001 ARRANGEMENTS FOR PROGRAM CONTROL, E.G., CONTROL UNIT (EPO)

* E9.002 .Using wired connections, e.g.,plugboard (EPO)

* E9.003 .Using stored program, i.e., using internal store of processing (EPO)

* E9.004 ..Micro-control or micro-program arrangements (EPO)

* E9.005 ...Execution means for micro-instructions irrespective of the micro-instruction function, e.g., decoding of micro-instructions and nano-instructions; timing of micro instructions; programmable logic arrays; delays and fan-out problems (EPO)

* E9.006 ...Micro instruction function e.g., input/output micro-instruction; diagnostic micro-instruction; micro-instruction format (EPO)

* E9.007 ...Loading of the micro-program (EPO)

* E9.008 ...Enhancement of operational speed, e.g., by using several micro-control devices operating in parallel (EPO)

* E9.009 ...Address formation of the next micro-instruction (EPO)

* E9.01 ....Micro-instruction address formation(EPO)

* E9.011 ....Arrangements for next micro-instruction selection (EPO)

* E9.012 .....Micro-instruction selection based on results of processing (EPO)

* E9.013 ......By address selection on input of storage (EPO)

* E9.014 ......By instruction selection on output of storage (EPO)

* E9.015 .....Micro-instruction selection not based on processing results, e.g., interrupt, patch, first cycle store, diagnostic programs (EPO)

* E9.016 ..Arrangements for executing machine-instructions, e.g., instruction decode (EPO)

* E9.017 ...Controlling the executing of arithmetic operations (EPO)

* E9.018 ...Controlling the executing of logical operations (EPO)

* E9.019 ...Controlling single bit operations (EPO)

* E9.02 ...For comparing (EPO)

* E9.021 ...For format conversion (EPO)

* E9.022 ...Using storage based on relative movement between record carrier and transducer (EPO)

* E9.023 ...Register arrangements, e.g., register files, special registers (EPO)

* E9.024 ....Special purpose registers, e.g., segment register, profile register (EPO)

* E9.025 ....Register structure, e.g., multigauged registers (EPO)

* E9.026 .....Implementation provisions thereof, e.g., ports, bypass paths (EPO)

* E9.027 .....Organization of register space, e.g., distributed register files, register banks (EPO)

* E9.028 ...Instruction analysis, e.g., decoding, instruction word fields (EPO)

* E9.029 ....Variable length instructions or constant length instructions whereby the relative length of operation and operand part is variable (EPO)

* E9.03 ....Decoding the operand specifier, e.g., specifier format (EPO)Speech classification or search (EPO)

* E9.031 ....With implied specifier, e.g., top of stack (EPO)

* E9.032 ...For specific instructions not covered by the preceding groups, e.g., halt, synchronize (EPO)

ARRANGEMENTS FOR PROGRAM CONTROL, E.G., CONTROL UNIT (EPO)

.Using stored program, i.e., using internal store of processing (EPO)

..Arrangements for executing machine-instructions, e.g., instruction decode (EPO)

* E9.033 ...Controlling loading, storing, or clearing operations (EPO)

* E9.034 ...Controlling moving, shifting, or rotation operations (EPO)

* E9.035 ...With operation extension or modification (EPO)

* E9.036 ....Using data descriptors, e.g., dynamic data typing (EPO)

* E9.037 ....Using run time instruction translation (EPO)

* E9.038 ...Addressing or accessing the instruction operand or the result (EPO)

* E9.039 ....Of multiple operands or results(EPO)

* E9.04 ....Indirect addressing (EPO)

* E9.041 ....Indexed addressing (EPO)

* E9.042 .....Using index register, e.g., adding index to base address (EPO)

* E9.043 ......Using wraparound, e.g., modulo or circular addressing (EPO)

* E9.044 ......Using scaling, e.g., multiplication of index (EPO)

* E9.045 ...Concurrent instruction execution, e.g., pipeline, look ahead (EPO)

* E9.046 ....Data or operand accessing, e.g., operand prefetch, operand bypass (EPO)

* E9.047 .....Operand prefetch, e.g., prefetch instruction, address prediction (EPO)

* E9.048 .....Maintaining memory consistency (EPO)

* E9.049 ....Instruction issuing, e.g., dynamic instruction scheduling, out of order instruction execution (EPO)

* E9.05 .....Speculative instruction execution, e.g., conditional execution, procedural dependencies, instruction invalidation (EPO)

* E9.051 ......Using dynamic prediction, e.g., branch history table (EPO)

* E9.052 ......Using static prediction, e.g., branch taken strategy (EPO)

* E9.053 .....From multiple instruction streams, e.g., multistreaming (EPO)

* E9.054 .....Of compound instructions (EPO)

* E9.055 ....Instruction prefetch, e.g., instruction buffer (EPO)

* E9.056 .....For branches, e.g., hedging branch folding (EPO)

* E9.057 ......Using address buffers, e.g., return stack (EPO)

* E9.058 .....For loops, e.g., loop buffer (EPO)

* E9.059 .....With instruction modification, e.g. store into instruction stream (EPO)

* E9.06 ....Recovery, e.g., branch miss-prediction, exception handling (EPO)

* E9.061 .....Using multiple copies of the architectural state, e.g., shadow registers (EPO)

* E9.062 ....Using instruction pipelines (EPO)

* E9.063 .....Synchronization, e.g., clock skew (EPO)

* E9.064 .....Technology-related problems thereof, e.g., GaAs pipelines (EPO)

* E9.065 .....Pipelining a single stage, e.g., superpipelining (EPO)

* E9.066 ....Using a slave processor, e.g., coprocessor (EPO)

* E9.067 .....Which is not visible to the instruction set architecture, e.g., using memory mapping, illegal opcodes (EPO)

* E9.068 ......For non-native instruction set architecture (EPO)

* E9.069 .....Which is visible to the instruction set architecture (EPO)

* E9.07 ......Having access to instruction memory (EPO)

* E9.071 ....Using a plurality of independent parallel functional units (EPO)

* E9.072 .....Decoding (EPO)

* E9.073 ...Address formation of the next instruction, e.g., incrementing the instruction counter, jump (EPO)

* E9.074 ....Program or instruction counter, e.g., incrementing (EPO)

* E9.075 ....Branch or jump to non-sequential address (EPO)

* E9.076 .....Unconditional, e.g., indirect jump (EPO)

* E9.077 .....Conditional (EPO)

* E9.078 .....For cyclically repeating instructions, e.g., iterative operation, loop counter (EPO)

* E9.079 ....Condition code generation, e.g., status register (EPO)

* E9.08 ....Selective instruction skip or conditional execution, e.g., dummy cycle (EPO)

* E9.081 ....Sequential commutation, e.g., ring counter, cyclical pulse distribution (EPO)

* E9.082 ..Arrangements for executing sub-programs, i.e., combinations of several instructions (EPO)

* E9.083 ...Formation of sub-program jump address or of return address (EPO)

* E9.084 ....Object Oriented Method Invocation (EPO)

* E9.085 .....Optimizing for Receiver Type (EPO)

* E9.086 .Using record carriers containing only program instructions (EPO)

**********

FOREIGN ART COLLECTION

**********

FOR 000 CLASS-RELATED FOREIGN DOCUMENTS

CLASS 712 - ELECTRICAL COMPUTERS AND DIGITAL PROCESSING SYSTEMS: PROCESSING ARCHITECTURES AND INSTRUCTION PROCESSING

The E-subclasses in U.S. Class 712 provide for arrangements for program to control the execution, processing, or sequencing of instruction data within a processor such as Micro-control or micro-program arrangements; arrangements for executing machine-instructions; arrangements for executing sub-programs, i.e. combinations of several instructions; etc.

**E9.001 ARRANGEMENTS FOR PROGRAM CONTROL, E.G., CONTROL UNIT (EPO):**
This main group provides for the control of execution, processing, or sequencing of instruction data within a processor. This subclass is substantially the same in scope as ECLA classification G06F9/00.

**E9.002 Using wired connections, e.g., plugboard (EPO):**
This subclass is indented under subclass E9.001. This subclass is substantially the same in scope as ECLA classification G06F9/02.

**E9.003 Using stored program, i.e., using internal store of processing (EPO):**
This subclass is indented under subclass E9.001. This subclass is substantially the same in scope as ECLA classification G06F9/06.

**E9.004 Micro-control or micro-program arrangements (EPO):**
This subclass is indented under subclass E9.003. This subclass is substantially the same in scope as ECLA classification G06F9/22.

**E9.005 Execution means for micro-instructions irrespective of the micro-instruction function, e.g., decoding of micro-instructions and nano-instructions; timing of micro instructions; programmable logic arrays; delays and fan-out problems (EPO):**
This subclass is indented under subclass E9.004. This subclass is substantially the same in scope as ECLA classification G06F9/22D.

**E9.006 Micro instruction function e.g., input/output micro-instruction; diagnostic micro-instruction; micro-instruction format (EPO):**
This subclass is indented under subclass E9.004. This subclass is substantially the same in scope as ECLA classification G06F9/22F.

**E9.007 Loading of the micro-program (EPO):**
This subclass is indented under subclass E9.004. This subclass is substantially the same in scope as ECLA classification G06F9/24.

**E9.008 Enhancement of operational speed, e.g., by using several micro-control devices operating in parallel (EPO):**
This subclass is indented under subclass E9.004. This subclass is substantially the same in scope as ECLA classification G06F9/28.

**E9.009 Address formation of the next micro-instruction (EPO):**
This subclass is indented under subclass E9.004. This subclass is substantially the same in scope as ECLA classification G06F9/26

(1) Note. This subgroup includes microprogram storage or retrieval arrangements.

**E9.01 Micro-instruction address formation(EPO):**
This subclass is indented under subclass E9.009. This subclass is substantially the same in scope as ECLA classification G06F9/26F.

**E9.011 Arrangements for next micro-instruction selection (EPO):**
This subclass is indented under subclass E.009. This subclass is substantially the same in scope as ECLA classification G06F9/26N.

**E9.012 Micro-instruction selection based on results of processing (EPO):**
This subclass is indented under subclass E9.011. This subclass is substantially the same in scope as ECLA classification G06F9/26N1.

**E9.013 By address selection on input of storage (EPO):**
This subclass is indented under subclass E9.012. This subclass is substantially the same in scope as ECLA classification G06F9/26N1E.

**E9.014 By instruction selection on output of storage (EPO):**
This subclass is indented under subclass E9.012. This subclass is substantially the same in scope as ECLA classification G06F9/26N1S.

**E9.015 Micro-instruction selection not based on processing results, e.g., interrupt, patch, first cycle store, diagnostic programs (EPO):**
This subclass is indented under subclass E9.011. This subclass is substantially the same in scope as ECLA classification G06F9/26N2.

**E9.016 Arrangements for executing machine-instructions, e.g., instruction decode (EPO):**
This subclass is indented under subclass E9.003. This subclass is substantially the same in scope as ECLA classification G06F9/30.

SEE OR SEARCH THIS CLASS, SUBCLASS:

E9.004 for executing micro-instructions.

E9.082 for executing subprograms.

**E9.017 Controlling the executing of arithmetic operations (EPO):**
This subclass is indented under subclass E9.016. This subclass is substantially the same in scope as ECLA classification G06F9/302.

**E9.018 Controlling the executing of logical operations (EPO):**
This subclass is indented under subclass E9.016. This subclass is substantially the same in scope as ECLA classification G06F9/305.

**E9.019 Controlling single bit operations (EPO):**
This subclass is indented under subclass E9.016. This subclass is substantially the same in scope as ECLA classification G06F9/308.

**E9.02 For comparing (EPO):**
This subclass is indented under subclass E9.016. This subclass is substantially the same in scope as ECLA classification G06F9/30C.

**E9.021 For format conversion (EPO):**
This subclass is indented under subclass E9.016. This subclass is substantially the same in scope as ECLA classification G06F9/30F.

**E9.022 Using storage based on relative movement between record carrier and transducer (EPO):**
This subclass is indented under subclass E9.016. This subclass is substantially the same in scope as ECLA classification G06F9/30Q.

**E9.023 Register arrangements, e.g., register files, special registers (EPO):**
This subclass is indented under subclass E9.016. This subclass is substantially the same in scope as ECLA classification G06F9/30R.

**E9.024 Special purpose registers, e.g., segment register, profile register (EPO):**
This subclass is indented under subclass E9.023. This subclass is substantially the same in scope as ECLA classification G06F9/30R2.

**E9.025 Register structure, e.g., multigauged registers (EPO):**
This subclass is indented under subclass E9.023. This subclass is substantially the same in scope as ECLA classification G06F9/30R4.

**E9.026 Implementation provisions thereof, e.g., ports, bypass paths (EPO):**
This subclass is indented under subclass E9.025. This subclass is substantially the same in scope as ECLA classification G06F9/30R4P.

**E9.027 Organization of register space, e.g., distributed register files, register banks (EPO):**
This subclass is indented under subclass E9.025. This subclass is substantially the same in scope as ECLA classification G06F9/30R4S.

**E9.028 Instruction analysis, e.g., decoding, instruction word fields (EPO):**
This subclass is indented under subclass E9.016. This subclass is substantially the same in scope as ECLA classification G06F9/30T.

**E9.029 Variable length instructions or constant length instructions whereby the relative length of operation and operand part is variable (EPO):**
This subclass is indented under subclass E9.028. This subclass is substantially the same in scope as ECLA classification G06F9/30T2.

**E9.03 Decoding the operand specifier, e.g., specifier format (EPO):**
This subclass is indented under subclass E9.028. This subclass is substantially the same in scope as ECLA classification 0G06F9/30T4.

**E9.031 With implied specifier, e.g., top of stack (EPO):**
This subclass is indented under subclass E9.028. This subclass is substantially the same in scope as ECLA classification G06F9/30T4S.

**E9.032 For specific instructions not covered by the preceding groups, e.g., halt, synchronize (EPO):**
This subclass is indented under subclass E9.016. This subclass is substantially the same in scope as ECLA classification G06F9/30Z.

**E9.033 Controlling loading, storing, or clearing operations (EPO):**
This subclass is indented under subclass E9.016. This subclass is substantially the same in scope as ECLA classification G06F9/312.

**E9.034 Controlling moving, shifting, or rotation operations (EPO):**
This subclass is indented under subclass E9.016. This subclass is substantially the same in scope as ECLA classification G06F9/315.

**E9.035 With operation extension or modification (EPO):**
This subclass is indented under subclass E9.016. This subclass is substantially the same in scope as ECLA classification G06F9/318.

**E9.036 Using data descriptors, e.g., dynamic data typing (EPO):**
This subclass is indented under subclass E9.035. This subclass is substantially the same in scope as ECLA classification G06F9/318D.

**E9.037 Using run time instruction translation (EPO):**
This subclass is indented under subclass E9.035. This subclass is substantially the same in scope as ECLA classification G06F9/318T.

**E9.038 Addressing or accessing the instruction operand or the result (EPO):**
This subclass is indented under subclass E9.016. This subclass is substantially the same in scope as ECLA classification G06F9/34.

**E9.039 Of multiple operands or results(EPO):**
This subclass is indented under subclass E9.038. This subclass is substantially the same in scope as ECLA classification G06F9/345.

**E9.04 Indirect addressing (EPO):**
This subclass is indented under subclass E9.038. This subclass is substantially the same in scope as ECLA classification G06F9/35.

(1) Note: Subject matter of this subgroup type includes using a single address operand, e.g., address register.

**E9.041 Indexed addressing (EPO):**
This subclass is indented under subclass E9.038. This subclass is substantially the same in scope as ECLA classification G06F9/355.

(1) Note: Subject matter of this subgroup type includes using more than one address operand.

**E9.042 Using index register, e.g., adding index to base address (EPO):**
This subclass is indented under subclass E9.041. This subclass is substantially the same in scope as ECLA classification G06F9/355A.

**E9.043 Using wraparound, e.g., modulo or circular addressing (EPO):**
This subclass is indented under subclass E9.042. This subclass is substantially the same in scope as ECLA classification G06F9/355A2.

**E9.044 Using scaling, e.g., multiplication of index (EPO):**
This subclass is indented under subclass E9.042. This subclass is substantially the same in scope as ECLA classification G06F9/355A4.

**E9.045 Concurrent instruction execution, e.g., pipeline, look ahead (EPO):**
This subclass is indented under subclass E9.016. This subclass is substantially the same in scope as ECLA classification G06F9/38.

**E9.046 Data or operand accessing, e.g., operand prefetch, operand bypass (EPO):**
This subclass is indented under subclass E9.045. This subclass is substantially the same in scope as ECLA classification G06F9/38D.

**E9.047 Operand prefetch, e.g., prefetch instruction, address prediction (EPO):**
This subclass is indented under subclass E9.046. This subclass is substantially the same in scope as ECLA classification G06F9/38D2.

**E9.048 Maintaining memory consistency (EPO):**
This subclass is indented under subclass E9.046. This subclass is substantially the same in scope as ECLA classification G06F9/38D4.

**E9.049 Instruction issuing, e.g., dynamic instruction scheduling, out of order instruction execution (EPO):**
This subclass is indented under subclass E9.045. This subclass is substantially the same in scope as ECLA classification G06F9/38E.

**E9.05 Speculative instruction execution, e.g., conditional execution, procedural dependencies, instruction invalidation (EPO):**
This subclass is indented under subclass E9.049. This subclass is substantially the same in scope as ECLA classification G06F9/38E2.

**E9.051 Using dynamic prediction, e.g., branch history table (EPO):**
This subclass is indented under subclass E9.05. This subclass is substantially the same in scope as ECLA classification G06F9/38E2D.

**E9.052 Using static prediction, e.g., branch taken strategy (EPO):**
This subclass is indented under subclass E9.05. This subclass is substantially the same in scope as ECLA classification G06F9/38E2S.

**E9.053 From multiple instruction streams, e.g., multistreaming (EPO):**
This subclass is indented under subclass E9.049. This subclass is substantially the same in scope as ECLA classification G06F9/38E4.

**E9.054 Of compound instructions (EPO):**
This subclass is indented under subclass E9.049. This subclass is substantially the same in scope as ECLA classification G06F9/38E6.

**E9.055 Instruction prefetch, e.g., instruction buffer (EPO):**
This subclass is indented under subclass E9.045. This subclass is substantially the same in scope as ECLA classification G06F9/38F.

**E9.056 For branches, e.g., hedging branch folding (EPO):**
This subclass is indented under subclass E9.055. This subclass is substantially the same in scope as ECLA classification G06F9/38F2.

**E9.057 Using address buffers, e.g., return stack (EPO):**
This subclass is indented under subclass E9.056. This subclass is substantially the same in scope as ECLA classification G06F9/38F2B.

**E9.058 For loops, e.g., loop buffer (EPO):**
This subclass is indented under subclass E9.055. This subclass is substantially the same in scope as ECLA classification G06F9/38F4.

**E9.059 With instruction modification, e.g., store into instruction stream (EPO):**
This subclass is indented under subclass E9.055. This subclass is substantially the same in scope as ECLA classification G06F9/38F6.

**E9.06 Recovery, e.g., branch miss-prediction, exception handling (EPO):**

This subclass is indented under subclass E9.045. This subclass is substantially the same in scope as ECLA classification G06F9/38H.

**E9.061 Using multiple copies of the architectural state, e.g., shadow registers (EPO):**
This subclass is indented under subclass E9.06. This subclass is substantially the same in scope as ECLA classification G06F9/38H2.

**E9.062 Using instruction pipelines (EPO):**
This subclass is indented under subclass E9.045. This subclass is substantially the same in scope as ECLA classification G06F9/38P.

**E9.063 Synchronization, e.g., clock skew (EPO):**
This subclass is indented under subclass E9.062. This subclass is substantially the same in scope as ECLA classification G06F9/38P2.

**E9.064 Technology-related problems thereof, e.g., GaAs pipelines (EPO):**
This subclass is indented under subclass E9.062. This subclass is substantially the same in scope as ECLA classification G06F9/38P4.

**E9.065 Pipelining a single stage, e.g., superpipelining (EPO):**
This subclass is indented under subclass E9.062. This subclass is substantially the same in scope as ECLA classification G06F9/38P6.

**E9.066 Using a slave processor, e.g., coprocessor (EPO):**
This subclass is indented under subclass E9.045. This subclass is substantially the same in scope as ECLA classification G06F9/38S.

**E9.067 Which is not visible to the instruction set architecture, e.g., using memory mapping, illegal opcodes (EPO):**
This subclass is indented under subclass E9.066. This subclass is substantially the same in scope as ECLA classification G06F9/38S4.

**E9.068 For non-native instruction set architecture (EPO):**
This subclass is indented under subclass E9.067. This subclass is substantially the same in scope as ECLA classification G06F9/38S4L.

**E9.069 Which is visible to the instruction set architecture (EPO):**
This subclass is indented under subclass E9.066. This subclass is substantially the same in scope as ECLA classification G06F9/38S6.

**E9.07 Having access to instruction memory (EPO):**
This subclass is indented under subclass E9.069. This subclass is substantially the same in scope as ECLA classification G06F9/38S6C.

**E9.071 Using a plurality of independent parallel functional units (EPO):**
This subclass is indented under subclass E9.045. This subclass is substantially the same in scope as ECLA classification G06F9/38T.

**E9.072 Decoding (EPO):**
This subclass is indented under subclass E9.071. This subclass is substantially the same in scope as ECLA classification G06F9/38T2.

**E9.073 Address formation of the next instruction, e.g., incrementing the instruction counter, jump (EPO):**
This subclass is indented under subclass E9.016. This subclass is substantially the same in scope as ECLA classification G06F9/32.

SEE OR SEARCH THIS CLASS, SUBCLASS:

E9.083 for sub-program jumps.

**E9.074 Program or instruction counter, e.g., incrementing (EPO):**
This subclass is indented under subclass E9.073. This subclass is substantially the same in scope as ECLA classification G06F9/32A.

**E9.075 Branch or jump to non-sequential address (EPO):**
This subclass is indented under subclass E9.073. This subclass is substantially the same in scope as ECLA classification G06F9/32B.

**E9.076 Unconditional, e.g., indirect jump (EPO):**
This subclass is indented under subclass E9.075. This subclass is substantially the same in scope as ECLA classification G06F9/32B2.

**E9.077 Conditional (EPO):**
This subclass is indented under subclass E9.075. This subclass is substantially the same in scope as ECLA classification G06F9/32B4.

**E9.078 For cyclically repeating instructions, e.g., iterative operation, loop counter (EPO):**
This subclass is indented under subclass E9.075. This subclass is substantially the same in scope as ECLA classification G06F9/32B6.

**E9.079 Condition code generation, e.g., status register (EPO):**
This subclass is indented under subclass E9.073. This subclass is substantially the same in scope as ECLA classification G06F9/32C.

**E9.08 Selective instruction skip or conditional execution, e.g., dummy cycle (EPO):**
This subclass is indented under subclass E9.073. This subclass is substantially the same in scope as ECLA classification G06F9/32S.

**E9.081 Sequential commutation, e.g., ring counter, cyclical pulse distribution (EPO):**
This subclass is indented under subclass E9.073. This subclass is substantially the same in scope as ECLA classification G06F9/32T.

**E9.082 Arrangements for executing sub-programs, i.e., combinations of several instructions (EPO):**
This subclass is indented under subclass E9.003. This subclass is substantially the same in scope as ECLA classification G06F9/40.

**E9.083 Formation of sub-program jump address or of return address (EPO):**
This subclass is indented under subclass E9.082. This subclass is substantially the same in scope as ECLA classification G06F9/42.

SEE OR SEARCH THIS CLASS, SUBCLASS:

E9.051 and E9.052 for branch prediction in a pipelined system.

**E9.084 Object Oriented Method Invocation (EPO):**
This subclass is indented under subclass E9.083. This subclass is substantially the same in scope as ECLA classification G06F9/42M.

**E9.085 Optimizing for Receiver Type (EPO):**
This subclass is indented under subclass E9.084. This subclass is substantially the same in scope as ECLA classification G06F9/42M1.

**E9.086 Using record carriers containing only program instructions (EPO):**
This subclass is indented under subclass E9.001. This subclass is substantially the same in scope as ECLA classification G06F9/04.