EUROPEAN PATENT OFFICE
U.S. PATENT AND TRADEMARK OFFICE

CPC NOTICE OF CHANGES 1476

DATE: AUGUST 1, 2023

PROJECT RP10466

The following classification changes will be effected by this Notice of Changes:

Action	**Subclass**	**Group(s)**
SCHEME:		
Symbols Deleted:	G06F	9/3855, 9/3857, 9/3859
Symbols New:	G06F	9/30038, 9/323, 9/3854, 9/3856, 9/3858, 9/38585, 9/38873, 9/38875, 9/3888, 9/38885
Titles Changed:	G06F	9/30018, 9/30036, 9/30072, 9/30079, 9/30112, 9/325, 9/35, 9/355, 9/3826, 9/3834, 9/3836, 9/3844, 9/3851, 9/3887
Indents Changed:	G06F	9/30061
Warnings New:	G06F	9/30018, 9/30036, 9/30038, 9/3005, 9/30054, 9/30058, 9/30061, 9/322, 9/323, 9/38, 9/3851, 9/3854, 9/3858, 9/3887, 9/38873, 9/3888, 9/38885
DEFINITIONS:		
Definitions Deleted:	G06F	9/3855, 9/3857, 9/3859
Definitions New:	G06F	9/30038, 9/323, 9/3854, 9/3856, 9/3858, 9/38585, 9/38873, 9/38875, 9/3888, 9/38885
Definitions Modified:	G06F	9/30003, 9/30007, 9/3001, 9/30014, 9/30018, 9/30029, 9/30032, 9/30036, 9/30043, 9/30047, 9/3005, 9/30054, 9/30058, 9/30061, 9/30065, 9/30072, 9/30076, 9/30094, 9/30109, 9/30112, 9/30116, 9/30141, 9/30145, 9/30149, 9/30152, 9/30156, 9/3016, 9/3017, 9/30174, 9/30185, 9/30189, 9/30196, 9/3806, 9/3808, 9/381, 9/3822, 9/3826, 9/3828, 9/3834, 9/3836, 9/3838, 9/3842, 9/3844, 9/3846, 9/3848, 9/3851, 9/3863, 9/3865, 9/3873, 9/3875, 9/3879, 9/3887, 9/3891, 9/3893

The following subclasses/groups are also impacted by this Notice of Changes (indicate subclasses/groups outside of the project scope, such as those listed in the CRL): G06F9/384, G06F9/3861

This Notice of Changes includes the following *[Check the ones included]:*

1. CLASSIFICATION SCHEME CHANGES
   - ☒ A. New, Modified or Deleted Group(s)
   - ☒ B. New, Modified or Deleted Warning(s)
   - ☐ C. New, Modified or Deleted Note(s)
   - ☐ D. New, Modified or Deleted Guidance Heading(s)
2. DEFINITIONS
   - ☒ A. New or Modified Definitions (Full definition template)
   - ☒ B. Modified or Deleted Definitions (Definitions Quick Fix)
3. ☒ REVISION CONCORDANCE LIST (RCL)
4. ☒ CHANGES TO THE CPC-TO-IPC CONCORDANCE LIST (CICL)
5. ☒ CHANGES TO THE CROSS-REFERENCE LIST (CRL)

CPC NOTICE OF CHANGES 1476

DATE: AUGUST 1, 2023

PROJECT RP10466

1. CLASSIFICATION SCHEME CHANGES

A. New, Modified or Deleted Group(s)

SUBCLASS G06F - ELECTRIC DIGITAL DATA PROCESSING

Type*	Symbol	Indent Level Number of dots (e.g. 0, 1, 2)	Title "CPC only" text should normally be enclosed in {curly brackets}**	Transferred to#
C	G06F 9/30018	5	{Bit or string instructions}	G06F 9/30018, G06F 9/30038
C	G06F 9/30036	5	{Instructions to perform operations on packed data, e.g. vector, tile or matrix operations}	G06F 9/30036, G06F 9/30038
N	G06F 9/30038	6	{using a mask}	
C	G06F 9/3005	4	{to perform operations for flow control}	G06F 9/3005, G06F 9/323,
C	G06F 9/30054	5	{Unconditional branch instructions}	G06F 9/30054, G06F 9/323
C	G06F 9/30058	5	{Conditional branch instructions}	G06F 9/30058, G06F 9/323
C	G06F 9/30061	5	{Multi-way branch instructions, e.g. CASE}	G06F 9/30054, G06F 9/30061, G06F 9/323
M	G06F 9/30072	4	{to perform conditional operations, e.g. using predicates or guards}	
M	G06F 9/30079	5	{Pipeline control instructions, e.g. multicycle NOP}	
M	G06F 9/30112	5	{comprising data of variable length}	
C	G06F 9/322	4	{for non-sequential address}	G06F 9/322, G06F 9/323
N	G06F 9/323	5	{for indirect branch instructions}	
M	G06F 9/325	5	{for loops, e.g. loop detection or loop counter}	
M	G06F 9/35	4	Indirect addressing	
M	G06F 9/355	4	Indexed addressing	
C	G06F 9/38	3	Concurrent instruction execution, e.g. pipeline, look ahead	G06F 9/38, G06F 9/3854
M	G06F 9/3826	5	{Bypassing or forwarding of data results, e.g. locally between pipeline stages or within a pipeline stage}	
M	G06F 9/3834	5	{Maintaining memory consistency}	
M	G06F 9/3836	4	{Instruction issuing, e.g. dynamic instruction scheduling or out of order instruction execution}	

M	G06F 9/3844	6	{using dynamic branch prediction, e.g. using branch history tables}	
C	G06F 9/3851	5	{from multiple instruction streams, e.g. multistreaming}	G06F 9/3851, G06F 9/3888
U	G06F 9/3853	5	{of compound instructions}	
N	G06F 9/3854	4	{Instruction completion, e.g. retiring, committing or graduating}	
D	G06F 9/3855	5	{Reordering, e.g. using a queue, age tags}	<administrative transfer to G06F 9/3856>
N	G06F 9/3856	5	{Reordering of instructions, e.g. using queues or age tags}	
D	G06F 9/3857	5	{Result writeback, i.e. updating the architectural state}	<administrative transfer to G06F 9/3858>
Q	G06F 9/3858	5	{Result writeback, i.e. updating the architectural state or memory}	G06F 9/3854, G06F 9/3858
D	G06F 9/3859	6	{with result invalidation, e.g. nullification}	<administrative transfer to G06F 9/38585>
N	G06F 9/38585	6	{with result invalidation, e.g. nullification}	
C	G06F 9/3887	5	{controlled by a single instruction for multiple data lanes [SIMD]}	G06F 9/3887, G06F 9/38873, G06F 9/38875, G06F 9/3888, G06F 9/38885
N	G06F 9/38873	6	{Iterative single instructions for multiple data lanes [SIMD]}	
N	G06F 9/38875	7	{for adaptable or variable architectural vector length}	
N	G06F 9/3888	5	{controlled by a single instruction for multiple threads [SIMT] in parallel}	
N	G06F 9/38885	6	{Divergence aspects}	

*N = new entries where reclassification into entries is involved; C = entries with modified file scope where reclassification of documents from the entries is involved; Q = new entries which are firstly populated with documents via administrative transfers from deleted (D) entries. Afterwards, the transferred documents into the Q entry will either stay or be moved to more appropriate entries, as determined by intellectual reclassification; T = existing entries with enlarged file scope, which receive documents from C or D entries, e.g. when a limiting reference is removed from the entry title; M = entries with no change to the file scope (no reclassification); D = deleted entries; F = frozen entries will be deleted once reclassification of documents from the entries is completed; U = entries that are unchanged.

NOTES:

- **No {curly brackets} are used for titles in CPC only subclasses, e.g. C12Y, A23Y; 2000 series symbol titles of groups found at the end of schemes (orthogonal codes); or the Y section titles. The {curly brackets} are used for 2000 series symbol titles found interspersed throughout the main trunk schemes (breakdown codes).
- U groups: it is obligatory to display the required "anchor" symbol (U group), i.e. the entry immediately preceding a new group or an array of new groups to be created (in case new groups are not clearly subgroups of C-type groups). Always include the symbol, indent level and title of the U group in the table above.
- All entry types should be included in the scheme changes table above for better understanding of the overall scheme change picture. Symbol, indent level, and title are required for all types.

- "Transferred to" column must be completed for all C, D, F, and Q type entries. F groups will be deleted once reclassification is completed.
- When multiple symbols are included in the "Transferred to" column, avoid using ranges of symbols in order to be as precise as possible.
- For administrative transfer of documents, the following text should be used: "<administrative transfer to XX>", "<administrative transfer to XX and YY simultaneously>", or "<administrative transfer to XX, YY, ...and ZZ simultaneously>" when administrative transfer of the same documents is to more than one place.
- Administrative transfer to main trunk groups is assumed to be the source allocation type, unless otherwise indicated.
- Administrative transfer to 2000/Y series groups is assumed to be "additional information".
- If needed, instructions for allocation type should be indicated within the angle brackets using the abbreviations "ADD" or "INV": <administrative transfer to XX ADD>, <administrative transfer to XX INV>, or <administrative transfer to XX ADD, YY INV, ... and ZZ ADD simultaneously>.
- In certain situations, the "D" entries of 2000-series or Y-series groups may not require a destination ("Transferred to") symbol, however it is required to specify "<no transfer>" in the "Transferred to" column for such cases.
- For finalization projects, the deleted "F" symbols should have <no transfer> in the "Transferred to" column.
- For more details about the types of scheme change, see CPC Guide.

B. New, Modified or Deleted Warning notice(s)

SUBCLASS G06F - ELECTRIC DIGITAL DATA PROCESSING

Type*	Location	Old Warning notice	New/Modified Warning
N	G06F 9/30018		Group G06F 9/30018 is impacted by reclassification into group G06F 9/30038. Groups G06F 9/30018 and G06F 9/30038 should be considered in order to perform a complete search.
N	G06F 9/30036		Group G06F 9/30036 is impacted by reclassification into group G06F 9/30038. Groups G06F 9/30036 and G06F 9/30038 should be considered in order to perform a complete search.
N	G06F 9/30038		Group G06F 9/30038 is incomplete pending reclassification of documents from groups G06F 9/30018 and G06F 9/30036. Groups G06F 9/30018, G06F 9/30036 and G06F 9/30038 should be considered in order to perform a complete search.
N	G06F 9/3005		Group G06F 9/3005 is impacted by reclassification into group G06F 9/323. Groups G06F 9/3005 and G06F 9/323 should be considered in order to perform a complete search.
N	G06F 9/30054		Group G06F 9/30054 is incomplete pending reclassification of documents from group G06F 9/30061. Group G06F 9/30054 is also impacted by reclassification into group G06F 9/323. Groups G06F 9/30054, G06F 9/30061 and G06F 9/323 should be considered in order to perform a complete search.
N	G06F 9/30058		Group G06F 9/30058 is impacted by reclassification into group G06F 9/323. Groups G06F 9/30058 and G06F 9/323 should be considered in order to perform a complete search.
N	G06F 9/30061		Group G06F 9/30061 is impacted by reclassification into groups G06F 9/30054 and G06F 9/323. Groups G06F 9/30061, G06F 9/30054 and G06F 9/323 should be considered in order to perform a complete search.
N	G06F 9/322		Group G06F 9/322 is impacted by reclassification into group G06F 9/323. Groups G06F 9/322 and G06F 9/323 should be considered in order to perform a complete search.
N	G06F 9/323		Group G06F 9/323 is incomplete pending reclassification of documents from groups G06F 9/3005, G06F 9/30054, G06F 9/30058, G06F 9/30061 and G06F 9/322. All groups listed in this Warning should be considered in order to perform a complete search.

N	G06F 9/38		Group G06F 9/38 is impacted by reclassification into group G06F 9/3854. Groups G06F 9/38 and G06F 9/3854 should be considered in order to perform a complete search.
N	G06F 9/3851		Group G06F 9/3851 is impacted by reclassification into group G06F 9/3888. Groups G06F 9/3851 and G06F 9/3888 should be considered in order to perform a complete search.
N	G06F 9/3854		Group G06F 9/3854 is incomplete pending reclassification of documents from groups G06F 9/38 and G06F 9/3858. Groups G06F 9/38, G06F 9/3858 and G06F 9/3854 should be considered in order to perform a complete search.
N	G06F 9/3858		Group G06F 9/3858 is impacted by reclassification into group G06F 9/3854. Groups G06F 9/3858 and G06F 9/3854 should be considered in order to perform a complete search.
N	G06F 9/3887		Group G06F 9/3887 is impacted by reclassification into groups G06F 9/38873, G06F 9/38875, G06F 9/3888 and G06F 9/38885. All groups listed in this Warning should be considered in order to perform a complete search.
N	G06F 9/38873		Groups G06F 9/38873 and G06F 9/38875 are incomplete pending reclassification of documents from group G06F 9/3887. Groups G06F 9/3887, G06F 9/38873 and G06F 9/38875 should be considered in order to perform a complete search.
N	G06F 9/3888		Group G06F 9/3888 is incomplete pending reclassification of documents from groups G06F 9/3851 and G06F 9/3887. Groups G06F 9/3851, G06F 9/3887 and G06F 9/3888 should be considered in order to perform a complete search.
N	G06F 9/38885		Group G06F 9/38885 is incomplete pending reclassification of documents from group G06F 9/3887. Groups G06F 9/3887 and G06F 9/38885 should be considered in order to perform a complete search.

*N = new warning, M = modified warning, D = deleted warning

NOTE: The "Location" column only requires the symbol PRIOR to the location of the warning. No further directions such as "before" or "after" are required.

2. A. DEFINITIONS (new)

Insert: The following new Definitions.

G06F 9/30038

Definition statement:
This place covers:

Using a mask while operating on and/or generating packed data. A mask may contain one or more bits for each element of packed data and may be located in a mask register.

Generating a mask used for operating on and/or generating packed data.

Synonyms and Keywords
In patent documents, the following words/expressions are often used as synonyms:

- mask and predicate

G06F 9/323

Definition statement:
This place covers:

Determination of program counter for an instruction that specifies where an address is located (rather than specifying the address itself) and branches using the address. The specification of the address location can be explicit (e.g. Branch R1) or implicit (e.g. RETURN). The address can be the target address or a base address used to calculate the target address.

References

Informative references

Attention is drawn to the following places, which may be of interest for search:

Special adaptations or details of handling of a specific unconditional indirect branch instruction that are distinct from program counter determination	G06F 9/30054

Special adaptations or details of handling of a specific conditional indirect branch instruction that are distinct from program counter determination	G06F 9/30058

Synonyms and Keywords

In patent documents, the following words/expressions are often used as synonyms:

- Indirect branch, computed branch, register-indirect branch and indirect jump

G06F 9/3854

Definition statement

This place covers:

Post-execution stages

G06F 9/3856

Definition statement

This place covers:

Special arrangements for reordering of instructions issued out-of-order;

Queue arrangements include reorder buffers;

Age tags include marking the instructions with the original program order.

Glossary of terms

In this place, the following terms or expressions are used with the meaning indicated:

Reordering	Restoring the program order after instruction execution, ensuring that the instructions complete in the correct order.
Age tag	An indicator associated with an instruction to indicate its original program order, e.g. in the case of instructions executed out-of-order.

G06F 9/3858

Definition statement

This place covers:

Special arrangements to write back results to the architectural state or memory, which may be for ensuring correctness of the architectural state;

Special arrangements to write back multiple results from a low-level atomic or transactional block, e.g. machine instructions between a start transactional execution machine instruction and an end transactional execution machine instruction.

References

Informative references

Attention is drawn to the following places, which may be of interest for search:

Specific machine instruction to store data to memory	G06F 9/30043
Maintaining memory consistency	G06F9/3834
Recovery of architectural state after an exception	G06F 9/3861

Glossary of terms

In this place, the following terms or expressions are used with the meaning indicated:

Architectural state	Runtime data in the pipeline resources, including program counter, instruction queue, status register, condition codes, general purpose and special purpose registers, rename data, pipeline registers, etc. The state is updated when one of these resources is written to.

G06F 9/38585

Definition statement

This place covers:

Ensuring correctness of the architectural state by nullifying the results of wrongly executed instructions.

Nullifying may use, for example, preventing writeback; tagging the result as invalid; clearing of result.

Arrangements for nullifying the result of a predicted instruction where the predicate resolves to false.

Nullifying multiple results from a low-level atomic or transactional block, e.g. machine instructions between a start transactional execution machine instruction and an end transactional execution machine instruction.

References:

Informative references:

Attention is drawn to the following places, which may be of interest for search:

Instructions which execute conditionally	G06F 9/30072
Recovery from exceptions	G06F 9/3861

Glossary of terms

In this place, the following terms or expressions are used with the meaning indicated:

Nullification	Invalidation of an instruction result. The instruction has already executed, but the results are invalid, and must not update the architectural state.

G06F 9/38873

Definition statement

This place covers:

Special arrangements at runtime for performing multiple iterations of a loop in parallel using SIMD lanes.

References

Informative references

Attention is drawn to the following places, which may be of interest for search:

Reducing the execution time required by program code via optimisations performed during compilation	G06F 8/4441
Software pipelining using a compiler	G06F 8/4452
Distributing iterations of parallelizable loops among processors using a compiler	G06F 8/452
Parallelism detection by a compiler	G06F 8/456

G06F 9/38875

Definition statement

This place covers:

Runtime determination of a vector length used for performing multiple iterations of a loop in parallel.

Using a vector length variable stored in a vector length register for performing multiple iterations of a loop in parallel.

G06F 9/3888

Definition statement

This place covers:

An execution model where multiple independent threads execute a same instruction in parallel, typically on different data elements (SIMT), where conditional instructions may cause different threads to follow divergent execution paths through a program, which may result in individual threads being inactive at times, and each thread may have its own instruction address counter and register state.

References

Informative references

Attention is drawn to the following places, which may be of interest for search:

Specific instructions to control multi-threading, e.g. FORK or JOIN instructions	G06F 9/3009

Application-oriented references

Examples of places where the subject matter of this place is covered when specially adapted, used for a particular purpose, or incorporated in a larger system:

Processor architectures or configurations for image data processing	G06T 1/20

Glossary of terms

In this place, the following terms or expressions are used with the meaning indicated:

Warp	A warp is a set of parallel threads that execute the same instruction together.
Thread Block	A thread block is a set of concurrent threads that can cooperate among themselves through barrier synchronization and shared access to a memory space private to the thread block. Once a thread block is assigned to a streaming multiprocessor, it is further partitioned into warps.
Grid	A grid is a set of thread blocks that may each be executed independently and thus may execute in parallel.
Streaming multiprocessor	A streaming multiprocessor executes warps and comprises multiple stream processors.

Synonyms and Keywords

In patent documents, the following abbreviations are often used:

SIMT	Single Instruction Multiple Thread
CTA	Cooperative Thread Array
GPGPU	General-Purpose Computing on Graphics Processing Units
SM	Streaming Multiprocessor
SP	Streaming Processor
CUDA	Compute Unified Device Architecture
OpenCL	Open Computing Language

In patent documents, the following words/expressions are often used as synonyms:

- “CUDA”, “OpenCL” and “GPGPU”
- "thread block”, “work group” and “cooperative thread array”
- “thread” and “work item”
- “warp”, “wavefront” and “thread group”
- “streaming multi-processor” and “compute unit”

G06F 9/38885

Definition statement

This place covers:

Special arrangements, in a SIMT architecture, to handle different threads following divergent execution paths (e.g., control flow paths) through a program due to a conditional instruction.

Reconvergence of threads in a SIMT architecture.

2. A. DEFINITIONS (modified)

G06F 9/30003

Definition statement

Replace: The existing Definition statement text with the following updated text.

> Execution of specific individual machine instructions with a specific opcode and/or instruction format.
>
> Adaptation of hardware, and hardware control, to carry out the execution of a specific machine instruction with a specific opcode and/or instruction format.

Special rules of classification

Replace: The existing Special rules of classification text with the following updated text.

> In the subgroups of G06F9/30003, if the execution of the machine instruction includes special arrangements for the setting of a condition code or flag, then also use G06F9/30094.
>
> In the case of a single machine instruction which carries out a combination of operations, use a subgroup for each operation.
>
> In the subgroups hereof, the terms in capitals which are used as examples, refer to well-known types of instructions characteristic to that subgroup.

Delete: The entire Glossary of terms section.

G06F9/30007

Definition statement

Insert: The following new second line of text so that the Definition statement reads as follows.

Specific instruction to perform operation between input data operands, usually returning an output data operand as the result.

Instructions for complex operations on data, e.g. checksum, hash, transforms, cryptography and random number generator instructions.

G06F 9/3001

Definition statement

Replace: The existing Definition statement text with the following updated text.

Specific arithmetic instruction, e.g. add, multiply and multiply accumulate.

Includes how to select the specific operation to execute in an ALU.

Synonyms and Keywords

Replace: The existing Synonyms and Keywords table text with the following updated text.

In patent documents, the following abbreviations are often used:

ALU	Arithmetic Logic Unit
MAC	Multiply-Accumulate
FMA	Fused Multiply Add/Accumulate

G06F 9/30014

Definition statement

Replace: The existing Definition statement text with the following updated text.

Arithmetic operation where the bit width operated on may be variable precision; e.g. floating point with rounding to fit register; double precision arithmetic.

Bit-sliced arithmetic operation.

G06F 9/30018

Definition statement

Replace: The existing Definition statement text with the following updated text.

Specific instruction for operation on a series of connected bits, bytes, or characters.

Examples include the EDIT instruction which alters a portion of a character string, or a Find-First-One instruction which detects the position of the first '1' in a string of bits.

Includes cyclic redundancy check instructions.

G06F 9/30029

Definition statement

Replace: The existing Definition statement text with the following updated text.

Specific instruction for logical operation or combination, e.g. XOR, NOT.

G06F 9/30032

Definition statement

Replace: The existing Definition statement text with the following updated text.

Specific instruction for moving, rearranging, or re-ordering data within a register.

Examples include: Move instruction which transfers data between registers; Permute/Shuffle instruction which changes the order of data in a register; Rotate or Shift instruction which moves bits or bytes within a register.

G06F 9/30036

Definition statement

Replace: The existing Definition statement text with the following updated text.

Specific instruction operating on multiple data stored in a single register, thereby effecting a SIMD operation.

Instructions operating on packed arrays of elements, e.g. vector, tile or matrix operations.

Special rules of classification

Replace: The existing Special rules text with the following updated text.

This subgroup may be used in combination with other subgroups of G06F 9/30007, according to the operation performed.

G06F 9/30043

Definition statement

Replace: The existing Definition statement text with the following updated text.

Specific instruction to read or write data from a memory location, e.g. LOAD, STORE, Load Multiple.

Specific instruction to clear or reset a memory location, e.g. CLEAR.

Register reset or clear instructions.

Context saving or restoring instructions.

Special rules of classification

Delete: The first two lines of the Special rules of classification section so that the text reads as follows.

> For Load Multiple when executed as an iterative instruction use also G06F 9/30065.

G06F 9/30047

Definition statement

Replace: The existing Definition statement text with the following updated text.

> Specific instructions for control data or instruction prefetching from memory, e.g. Hint instruction.
>
> Specific instructions to control cache operation, e.g. Cache Flush.
>
> Specific instructions to control a TLB or a page table, e.g. page table entry clearing instruction.

G06F 9/3005

Definition statement

Replace: The existing Definition statement text with the following updated text.

> Specific instruction to control program flow in general.
>
> Execution of an instruction to select a next instruction other than the next sequential instruction, e.g. for branching.
>
> Execution of an instruction for facilitating branching, e.g. Prepare-To-Branch instruction.

Specific instruction for monitoring or tracing program flow e.g. breakpoint instruction; flow signature instruction.

G06F 9/30054

Definition statement

Replace: The existing Definition statement text with the following updated text.

> Special adaptations to execute a specific instruction which unconditionally branches to a target address independent of any condition.
>
> Examples of unconditional branch instructions are CALL, GOTO and RETURN insofar as these are unconditional.

Special rules of classification

Replace: The existing Special rules of classification text with the following updated text.

> Only to be used when there is subject matter relating to special adaptations or details of handling of an unconditional branch instruction.

G06F 9/30058

Definition statement

Replace: The existing Definition statement text with the following updated text.

> Specific instruction which causes conditional branching to a target address dependent on a runtime condition, else continues execution with the next sequential instruction.
>
> Includes IF-THEN-ELSE constructions.

Special rules of classification

Replace: The existing Special rules text with the following updated text.

> Only to be used when there is subject matter relating to special adaptations or details of handling of a specific conditional branch instruction.

G06F 9/30061

Definition statement

Delete: The second line of the Definition statement text so that the Definition statement reads as follows.

> Specific instruction which causes a branching to one of several alternative target addresses depending on a runtime condition.

G06F 9/30065

Definition statement

Replace: The existing Definition statement text with the following updated text.

> Specific instruction used for loop control, e.g. specific loop start or end instructions.
>
> Specific instruction which is repeatedly executed, thereby forming a (short) loop, e.g. REPEAT.

G06F 9/30072

Definition statement

Replace: The existing Definition statement text with the following updated text.

> Specific instruction for conditional operation depending on a runtime condition, which is not for control of program flow, i.e. instruction that is not a branch.
>
> The operation carried out depends on a runtime condition, for example ADD or SUBTRACT depending on the value of the sign bit. Another example is a MOVE which is executed or not depending on a runtime condition.
>
> Includes instructions which are executed conditional on a predicate or guard.
>
> Includes conditional instructions in a branch shadow.

G06F 9/30076

Definition statement

Replace: The existing Definition statement text with the following updated text.

Specific instructions for operation control in general.

Includes mode switching instructions.

Includes no-operation instructions [NOP].

References

Informative references

Replace: The existing Informative references table with the following updated table.

Specific instructions for program flow control	G06F 9/3005
Multi-cycle NOP used as a pipeline delay instruction	G06F 9/30079

G06F 9/30094

Definition statement

Replace: The existing Definition statement text with the following updated text.

Special arrangements for the generation or storage of runtime conditions, e.g. flags (Carry, Zero flag, etc.); writing to status register.

References

Informative references

Replace: The existing Informative references table with the following updated table.

Execution of instructions according to a runtime mode	G06F 9/30189

G06F 9/30109

Definition statement

Replace: The existing Definition statement text with the following updated text.

> Registers which are logically partitioned into multiple operands, e.g. for packed data or parallel operations.

G06F 9/30112

Definition statement

Replace: The existing Definition statement text with the following updated text.

> Register structure for variable length operands, i.e. variable length data can be stored, e.g. single register for storing an M-bit integer or an N-bit integer, or a single register for storing an X-bit integer or a Y-bit floating point value.
>
> Use of partial registers for short data.
>
> Combinations of registers for longer or higher-precision data, e.g. by concatenation.
>
> Accessing of variable length registers.

References

Informative references

Replace: The existing Informative references table text with the following updated text.

A single register for multiple operands	G06F 9/30109

G06F 9/30116

Definition statement

Replace: The existing Definition statement text with the following updated text.

Registers which cannot be addressed by an instruction, and hence are invisible to the architecture, e.g. coupled registers, not forming part of the register space.

Register with an associated copy, e.g. for saving of architectural state.

G06F 9/30141

Definition statement

Replace: The existing Definition statement text with the following updated text.

Hardware implementation of register files.

Register file port architecture; address or data ports.

Internal bypass path of register files.

Adaptations of register file hardware for particular problems, e.g. for power saving; for fault tolerance.

Includes transposing register file being accessible vertically or horizontally.

G06F 9/30145

Definition statement

Replace: The existing Definition statement text with the following updated text.

Decoding of instructions in general, of opcode in particular.

Instruction format, instruction encoding, instruction word fields.

Instruction set as a whole.

References

Informative references

Attention is drawn to the following places, which may be of interest for search:

Insert: The following new reference in the Informative references table.

Decoding of microinstructions	G06F9/223

G06F 9/30149

Definition statement

Replace: The existing Definition statement text with the following updated text.

Decoding of variable length instructions.

Includes instruction where the relative length of operation and operand part is variable.

Ensuring a whole instruction is decoded. Parsing variable length instructions (VLI).

Delete: The entire Glossary of terms section.

Delete: The entire Synonyms and Keywords section.

G06F 9/30152

Definition statement

Replace: The existing Definition statement text with the following updated text.

Arrangements for determining and/or marking the boundaries of a variable length instruction (VLI); Special arrangements for determining the length of a variable length instruction other than by decoding the length.

Delete: The entire Glossary of terms section.

Delete: The entire Synonyms and Keywords section.

G06F 9/30156

Definition statement

Replace: The existing Definition statement text with the following updated text.

Instruction encodings (e.g. Gray coding) to achieve a secondary effect, e.g. power saving, saving memory space, security, fault tolerance.

G06F 9/3016

Definition statement

Replace: The existing Definition statement text with the following updated text.

Decoding operand fields of instructions; Format of operand fields of instructions, e.g. specifier format.

G06F 9/3017

Definition statement

Replace: The existing Definition statement text with the following updated text.

Runtime translation of an instruction by decoding an instruction which is non-native to produce an instruction or set of instructions that can be decoded by the processor. The decoding of machine instructions of the executing processor's instruction set, or decoding of lower level microcode is not meant to be included here.

Altering the format or encoding of the input instruction, e.g. length of fields.

Translating a single instruction, e.g. macro, into multiple executable instructions, or the reverse (macro formation).

Delete: The entire Relationships with other classification places section.

References

Informative references

Attention is drawn to the following places, which may be of interest for search:

Insert: The following two new references in the Informative references table.

Decoding of microinstructions	G06F 9/223
Instruction emulation or interpretation	G06F 9/455

Glossary of terms

In this place, the following terms or expressions are used with the meaning indicated:

Replace: The existing Glossary of terms table text with the following updated text.

Macro	An opcode which is an alias for a series of instructions, i.e. a function.
Non-native instruction	An instruction which is not executable in the architecture of the processor.

G06F 9/30174

Definition statement

Replace: The existing Definition statement text with the following updated text.

Runtime translation of a non-native instruction (e.g. Javabyte, legacy code) into an executable native instruction using hardware means, e.g. decoder, look-up table.

Runtime translation for the purpose of ISA emulation in hardware.

G06F 9/30185

Definition statement

Replace: The existing Definition statement text with the following updated text.

Modification of the operation of an instruction according to one or more bits encoded within, or appended to, the instruction, e.g. prefix, sub-opcode.

Insert: The following new Informative references section.

Informative references

Attention is drawn to the following places, which may be of interest for search:

Modification of the operation of an instruction according to an execution mode	G06F9/30189

G06F 9/30189

Definition statement

Replace: The existing Definition statement text with the following updated text.

Modification of the operation of one or more instructions according to a mode of operation, e.g. mode flag in a mode register.

References

Informative references

Attention is drawn to the following places, which may be of interest for search:

Insert: The following new reference in the Informative references table.

Instruction operation extension or modification according to one or more bits in the instruction, e.g. prefix, sub-opcode	G06F 9/30185

G06F 9/30196

Definition statement

Replace: The existing Definition statement text with the following updated text.

Modification of the operation of an instruction by modifying the decoding of the instruction using more than one decoder, or a decoder which is adaptable or programmable.

Extension of the instruction set using multiple decoders for multiple instruction sets.

G06F 9/3806

Definition statement

Replace: The existing Definition statement text with the following updated text.

Using a history of previous branch target addresses to predict the address to fetch from, e.g. branch target buffer, branch history buffer;

Address buffers for predicting next fetch address for a branch, e.g. return address stack.

Glossary of terms

In this place, the following terms or expressions are used with the meaning indicated:

Replace: The existing Glossary of terms table with the following updated table.

BTB	buffer indexed by an instruction fetch address or PC, which returns the predicted target address if the instruction is a taken branch.
BHT	buffer indexed by a branch instruction address, which returns a prediction of whether the branch is taken or not.
Return address stack	Stack to hold the program address to return to after a Call-type branch. The stack structure allows nesting of Calls.

Synonyms and Keywords

In patent documents, the following abbreviations are often used:

Replace: The existing Synonyms and Keywords table with the following updated table.

BTB	Branch Target Buffer
BHT	Branch History Table
BTAC	Branch Target Address Cache

G06F 9/3808

Definition statement

Replace: The existing Definition statement text with the following updated text.

Prefetching of instructions intended to be used more than once, thereby saving fetch time.

Buffering of instructions for reuse, e.g. trace cache.

Branch target caches for caching a branch target instruction.

The storing of addresses, e.g. branch target address caches (BTAC), is not meant to be stored here.

G06F 9/381

Definition statement

Replace: The existing Definition statement text with the following updated text.

Prefetching of instructions intended to be used in a loop, thereby saving fetch time;

Buffering of instructions for loops.

Delete: The entire Glossary of terms section.

G06F 9/3822

Definition statement

Replace: The existing Definition statement text with the following updated text.

Decoding for enabling the parallel execution of instructions, e.g. parallel decode units.

Special details of decoding multiple instructions in parallel, e.g. decoding of Very Long Instruction Word format field.

G06F 9/3826

Definition statement

Replace: The existing Definition statement text with the following updated text.

Arrangements for the transfer of an instruction result to a dependent instruction, without first storing in the architected state, e.g. bypassing the register file;

Transfer of operand data from the output of a functional unit to the input of another functional unit, without waiting for the completion of the data producing instruction, or without waiting for the data to be stored in the register file, e.g. locally between pipeline stages, within a pipeline stage.

G06F 9/3828

Definition statement

Replace: The existing Definition statement text with the following updated text.

Bypass of an instruction result to a dependent instruction in another pipeline, or group of execution units, e.g. between pipelines, between clusters;

Bypass arrangements for global data.

References

Informative references

Attention is drawn to the following places, which may be of interest for search:

Replace: The existing reference symbol with the following updated symbol.

Parallel execution units organised in clusters	G06F9/3891

G06F 9/3834

References

Informative references

Attention is drawn to the following places, which may be of interest for search:

Replace: In the third reference, the existing reference symbol with the following updated symbol.

Consistency of architectural state	G06F9/3854

G06F 9/3836

Definition statement

Replace: The existing Definition statement text with the following updated text.

Runtime scheduling or issuing of instructions, e.g. dynamic instruction scheduling, out of order instruction execution.

Issuing policies or mechanisms for instructions. Instruction dispatching to execution units or execution buffers.

Concurrent execution of instructions.

Synchronisation of instruction execution.

G06F 9/3838

Definition statement

Replace: The existing Definition statement text with the following updated text.

Special arrangements to detect or record data dependencies between instruction operands at issue time, e.g. register scoreboarding.

G06F 9/3842

Definition statement

Replace: The existing Definition statement text with the following updated text.

Execution of instructions ahead of program order, with the presumption that execution will prove to be correct, e.g. speculative loads, boosting.

Speculative instructions which are executed, e.g. alternative paths of a branch.

Execution of instructions dependent on a branch before its outcome is known.

Execution of instructions in a low-level transaction, e.g. machine instructions between a start transactional execution machine instruction and an end transactional execution machine instruction.

References

Informative references

Attention is drawn to the following places, which may be of interest for search:

Replace: In the second reference, the existing reference symbol with the following updated symbol.

Result nullification for executed instructions	G06F9/38585

Glossary of terms

In this place, the following terms or expressions are used with the meaning indicated:

Insert: The following new row in the Glossary of terms table.

Transaction	A sequence of instructions executed as an atomic group. A transaction either commits (its updates take effect), or aborts and is rolled back (its updates are discarded).

G06F 9/3844

Definition statement

Replace: The existing Definition statement text with the following updated text.

Speculative execution of instructions using dynamic branch prediction;

Using runtime conditions, and the previous behaviour of branches, to predict the outcome of a branch, without having to wait for its execution, e.g. branch history table;

Early generation of branch results.

References

Delete: The entire Limiting references section.

Insert: The following new Informative references section.

Informative references

Attention is drawn to the following places, which may be of interest for search:

Prediction of a branch address/target	G06F9/3806
Using hybrid branch prediction	G06F9/3848

Glossary of terms

In this place, the following terms or expressions are used with the meaning indicated:

Replace: The existing Glossary of terms table with the following updated table.

Dynamic prediction	Branch prediction based on runtime conditions, as opposed to compile-time branch prediction.
Branch history table	Branch prediction based on runtime conditions, as opposed to compile-time branch prediction.
Branch Target Buffer	Buffer indexed by an instruction fetch address or PC, which returns the predicted target address if the instruction is a taken branch.
Branch History Table	Buffer indexed by a branch instruction address, which returns a prediction of whether the branch is taken or not.
Branch Prediction Counter	saturating counter used to obtain a weighting for a branch prediction based on several branch executions.

Synonyms and Keywords

In patent documents, the following abbreviations are often used:

Delete: The following row from the Synonyms and Keywords table.

BDT	Branch Decode Table

G06F 9/3846

Definition statement

Replace: The existing Definition statement text with the following updated text.

Speculative execution of instructions using static branch prediction, e.g. branch taken strategy;

Branch prediction performed by compiler, and not dependent on runtime conditions, e.g. hint bits.

References

Delete: The entire Limiting references section.

Insert: The following new Informative references section.

Informative references

Attention is drawn to the following places, which may be of interest for search:

Using hybrid branch prediction	G06F9/3848
Prediction of a branch address/target	G06F9/3806

G06F 9/3848

Definition statement

Replace: The existing Definition statement text with the following updated text.

Prediction schemes involving more than one type of predictor, e.g. selection between prediction techniques;

Static and dynamic prediction used alternately;

Local and global prediction mechanisms;

Two-level branch prediction.

References

Delete: The entire Limiting references section.

Insert: The following new Informative references section.

Informative references

Attention is drawn to the following places, which may be of interest for search:

Using dynamic prediction, e.g. branch history table	G06F9/3844

Using static prediction, e.g. branch taken strategy	G06F9/3846
Prediction of a branch address/target	G06F9/3806

Glossary of terms

In this place, the following terms or expressions are used with the meaning indicated:

Replace: The existing Glossary of terms table with the following updated table.

Two-level branch prediction	Two-dimensional prediction where the output of one method is used as an index into another method to provide a prediction (e.g. branch history's output as an index into a pattern history table).

G06F 9/3851

Definition statement

Replace: The existing Definition statement text with the following updated text.

- Issuing instructions from multiple threads each having a context, including at least a program counter, and possibly registers and execution resources;
- Includes multiple streams for different threads, or from both directions of a branch;
- Interleaved execution of threads in a single or in multiple streams;
- Stream selection.

References

Insert: The following new Limiting references section.

Limiting references

This place does not cover:

Dispatching of multiple tasks or threads	G06F 9/48

G06F 9/3863

Definition statement

Replace: The existing Definition statement text with the following updated text.

Recovery using multiple copies of architectural state;

Restoring the architectural state to that previous to an exception using a previous version of the state, e.g. checkpoint, future file, shadow registers. Also known as rollback.

G06F 9/3865

Definition statement

Replace: The existing Definition statement text with the following updated text.

Handling or nullification of an instruction exception, e.g. using exception flags, which does not occur in the cycle in which the exception is detected, but later, e.g. at writeback stage.

G06F9/3873

Definition statement

Replace: The existing Definition statement text with the following updated text.

Pipeline with dynamically varying length, e.g. elastic pipeline.

Multiple pipelines having different lengths.

G06F 9/3875

Definition statement

Replace: The existing Definition statement text with the following updated text.

Pipeline architecture where a single stage is split into sub-stages (e.g. superpipelining) using pipeline buffer, with higher clocking rate implied for that stage, e.g. pipelined execution unit; pipelined decode unit.

Pipeline architecture having multiple stages for the same function, e.g. two execution stages, without higher clock rate.

G06F 9/3879

Definition statement

Replace: The existing Definition statement text with the following updated text.

Slave processors which receive and decode instructions which are not explicit in the instruction set of the host, e.g. commands; function calls; using an escape code; using memory-mapped commands;

Slave processors which are adapted to execute a non-native instruction set, e.g. Java coprocessor.

G06F 9/3887

Definition statement

Replace: The existing Definition statement text with the following updated text.

Multiple parallel functional units controlled by a single instruction, e.g. SIMD.

For SIMD execution, this class contains details relevant to the execution aspects, e.g. executing a global instruction according to local conditions.

Relationships with other classification places

Replace: The Relationships text with the following updated text with updated spacing, so that the text appears as follows.

SIMD architectures: G06F 15/80.

G06F 9/3891

Definition statement

Replace: The existing Definition statement text with the following updated text.

Control of parallel execution by groups of functional units, such as multiple execution units sharing local memory, e.g. clusters;

Partitioned architectures, e.g. for hardware multistreaming.

G06F 9/3893

Definition statement

Replace: The existing Definition statement text with the following updated text.

Multiple functional units which are controlled in tandem or cascade to carry out an instruction.

Multiple functional units controlled by the same instruction but not in the same cycle, e.g. multiplier-accumulator.

Synonyms and Keywords

In patent documents, the following abbreviations are often used:

Insert: The following new row in the Synonyms and Keywords table.

FMA	Fused multiplier-accumulator unit

2. B. DEFINITIONS QUICK FIX

Symbol	**Location of change** (e.g., section title)	**Existing reference symbol or text**	**Action; New symbol; New text**
G06F9/3855			Delete entire definition
G06F9/3857			Delete entire definition
G06F9/3859			Delete entire definition

Notes:
Use this Definitions Quick Fix (DQF) table to:

- Delete an entire definition
- Delete an entire section
- Change a reference symbol
- Delete a reference symbol
- Delete text in a References section
- Correct one error in spelling, article use, or verb tense

Otherwise, use the standard template.

Reminder: Never delete F symbol definitions.

3. REVISION CONCORDANCE LIST (RCL)

Type*	**From CPC Symbol (existing)**	**To CPC Symbol(s)**
C	G06F 9/30018	G06F 9/30018, G06F 9/30038
C	G06F 9/30036	G06F 9/30036, G06F 9/30038
C	G06F 9/3005	G06F 9/3005, G06F 9/323,
C	G06F 9/30054	G06F 9/30054, G06F 9/323
C	G06F 9/30058	G06F 9/30058, G06F 9/323
C	G06F 9/30061	G06F 9/30054, G06F 9/30061, G06F 9/323
C	G06F 9/322	G06F 9/322, G06F 9/323
C	G06F 9/38	G06F 9/38, G06F 9/3854
C	G06F 9/3851	G06F 9/3851, G06F 9/3888
D	G06F 9/3855	<administrative transfer to G06F 9/3856>
D	G06F 9/3857	<administrative transfer to G06F 9/3858>
Q	G06F 9/3858	G06F 9/3854, G06F 9/3858
D	G06F 9/3859	<administrative transfer to G06F 9/38585>
C	G06F 9/3887	G06F 9/3887, G06F 9/38873, G06F 9/38875, G06F 9/3888, G06F 9/38885

* C = entries with modified file scope where reclassification of documents from the entries is involved; Q = new entries which are firstly populated with documents via administrative transfers from deleted (D) entries. Afterwards, the transferred documents into the Q entry will either stay or be moved to more appropriate entries, as determined by intellectual reclassification; D = deleted entries; F = frozen entries will be deleted once reclassification of documents from the entries is completed.

NOTES:

- Only C, D, F, and Q type entries are included in the table above.
- When multiple symbols are included in the "To" column, do not use ranges of symbols.
- For administrative transfer of documents, the following text should be used: "<administrative transfer to XX>", "<administrative transfer to XX and YY simultaneously>", or "<administrative transfer to XX, YY, ...and ZZ simultaneously>" when administrative transfer of the same documents is to more than one place.
- Administrative transfer to main trunk groups is assumed to be the source allocation type, unless otherwise indicated.
- Administrative transfer to 2000/Y series groups is assumed to be "additional information".
- If needed, instructions for allocation type should be indicated within the angle brackets using the abbreviations "ADD" or "INV": <administrative transfer to XX ADD>, <administrative transfer to XX INV>, or < administrative transfer to XX ADD, YY INV, ... and ZZ ADD simultaneously>.
- In certain situations, the "D" entries of 2000-series or Y-series groups may not require a destination ("To") symbol, however it is required to specify "<no transfer>" in the "To" column for such cases.
- RCL is not needed for finalisation projects.

4. CHANGES TO THE CPC-TO-IPC CONCORDANCE LIST (CICL)

CPC	**IPC**	**Action***
G06F 9/30038	G06F 9/30	NEW
G06F 9/323	G06F 9/32	NEW
G06F 9/3854	G06F 9/38	NEW
G06F 9/3855		DELETE
G06F 9/3856	G06F 9/38	NEW
G06F 9/3857		DELETE
G06F 9/3858	G06F 9/38	NEW
G06F 9/38585	G06F 9/38	NEW
G06F 9/3859		DELETE
G06F 9/38873	G06F 9/38	NEW
G06F 9/38875	G06F 9/38	NEW
G06F 9/3888	G06F 9/38	NEW
G06F 9/38885	G06F 9/38	NEW

*Action column:

- For an (N) or (Q) entry, provide an IPC symbol and complete the Action column with "NEW."
- For an existing CPC main trunk entry or indexing entry where the existing IPC symbol needs to be changed, provide an updated IPC symbol and complete the Action column with "UPDATED."
- For a (D) CPC entry or indexing entry complete the Action column with "DELETE." IPC symbol does not need to be included in the IPC column.
- For an (N) 2000 series CPC entry which is positioned within the main trunk scheme (breakdown code) provide an IPC symbol and complete the action column with "NEW".
- For an (N) 2000 series CPC entry positioned at the end of the CPC scheme (orthogonal code), with no IPC equivalent, complete the IPC column with "CPCONLY" and complete the action column with "NEW".

NOTES:

- F symbols are <u>not</u> included in the CICL table above.
- T and M symbols are not included in the CICL table above unless a change to the existing IPC is desired.

5. CROSS-REFERENCE LIST (CRL)

Definitions references impacted by this revision project

Location of reference to be changed	**Referenced subclass or group to be changed**	**Section of definition**	**Action; New reference symbol; New text**
G06F9/384	G06F9/3855	Informative references	G06F 9/3856
G06F9/3861	G06F9/3859	Informative references	G06F 9/38585

NOTES:

- The CRL tables above are used for changes to locations **outside** of the project scope. Changes to references in scheme titles or definitions inside the project scope will be reflected in the "scheme change" template or one of the "definition" templates.
- In addition to other changes proposed in the tables above, in the column titled "Referenced subclass or group to be changed," **referenced** D symbols should indicate an action of "delete" or should indicate a replacement symbol and **referenced** F symbols should indicate a replacement symbol.
- When a reference is deleted, text related to that reference will also be deleted unless other references or a range of references associated with the same text remain.