DAXPY (*Double precision- real Alpha X Plus Y*)

- Unidades funcionales (UF):
  - 1 unidad de carga (instrucción (1)) segmentada en 2 etapas de 1 ciclo cada etapa,
  - 1 unidad para almacenamiento (inst. (7)) de 0 ciclos,
  - 2 unidades ALU para enteros (`addl`, `addq`, `cmpl`) de 1 ciclo,
  - 1 unidad para `mulpd` de segmentada en 4 etapas de 1 ciclo cada etapa,
  - 1 unidad para `addpd` segmentada en 2 etapas de 1 ciclo cada etapa, y
  - 1 unidad de saltos (`jb`) de 1 ciclo en la etapa de ejecución.

```
for (i=0; i<N; i++) {
    y[i]=a*x[i]+y[i];
}
```

- Se captan, decodifican, emiten y se retiran del ROB 4 instrucciones ciclo.
  - ROB: **E**ntrada - **r**egistro renombrado – **o**perando - **v**alidez del operando – **m**arca(i,x,f)) – **f**lush
  - Se pueden captar junto a una instrucción de salto las instrucciones a ejecutar según la predicción del salto.
- Hay 5 etapas: captación (IF), decodificación+emisión (ID), envío a UF (E), ejecución (EX) y terminación del procesamiento (W). Todas las etapas consumen 1 ciclo de reloj, excepto EX cuyo consumo dependerá de la instrucción.
- Los valores de los registros están disponibles (válidos) al comienzo del procesamiento.
- Estación de reserva centralizada:
  - La etapa ID (decodificación+emisión) emite instrucciones a la estación de reserva
  - La etapa E (envío a UF) puede enviar a ejecutar instrucciones que tienen todos sus operandos disponibles. No puede enviar dos instrucciones en el mismo ciclo a la misma unidad funcional.

ROB y ventana de instruccio centralizada. Envío desorden

```
...
for (i=0; i<N; i++) {
   y[i]=a*x[i]+y[i];
}
...
```


instr.	fuente, fuente/destino	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	ok	ROB
movapd	(%rsi), %xmm1 ;(1) load x	IF	ID																									1-	
movapd	(%rax), %xmm3 ;(2) load y	IF	ID																									1-	E-r-o-v-m-f
addl	$1, %edx ;(3) i++	IF	ID																									11	
mulpd	%xmm2, %xmm1 ;(4) a*x	IF	ID																									10	
addq	$16, %rax ;(5)		IF	ID																								11	
addq	$16, %rsi ;(6)		IF	ID																								11	
addpd	%xmm3, %xmm1 ;(7) a*x+y		IF	ID																								00	
movaps	%xmm1,-16(%rax);(8) store y		IF	ID																								00	
cmpl	%edi, %edx ;(9) i<N?			IF	ID																							10	
jb	.L7 ;(10)			IF	ID																							01	
movapd	(%rsi), %xmm1 ;(11) load x			IF	ID																							0-	
movapd	(%rax), %xmm3 ;(12) load y			IF	ID																							0-	
addl	$1, %edx ;(13) i++				IF	ID																						10	
mulpd	%xmm2, %xmm1 ;(14) a*x				IF	ID																						10	
addq	$16, %rax ;(15)				IF	ID																						10	
addq	$16, %rsi ;(16)				IF	ID																						10	
addpd	%xmm3, %xmm1 ;(17)a*x+y					IF	ID																					00	
movaps	%xmm1,-16(%rax);(18)storey					IF	ID																					00	
cmpl	%edi, %edx ;(19) i<N?					IF	ID																					10	
jb	.L7 ;(20)					IF	ID																					01	

Situación al finalizar el ciclo 2 (almacenamiento y etapas usadas)

instr.	fuente, fuente/destino	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	ok	ROB
movapd	**(%rsi), %xmm1 ;(1) load**	IF	ID																									1-	1-x1- -0-i-0
movapd	(%rax), %xmm3 ;(2) load	IF	ID																									1-	2-x3- -0-i-0
addl	**$1, %edx ;(3)**	IF	ID																									11	3-dx- -0-i-0
mulpd	%xmm2, %xmm1 ;(4)	IF	ID																									10	4-x1- -0-i-0
addq	$16, %rax ;(5)		IF																									11	
addq	$16, %rsi ;(6)		IF																									11	
addpd	%xmm3, %xmm1 ;(7)		IF																									00	
movaps	%xmm1,-16(%rax);(8) store		IF																									00	
cmpl	%edi, %edx ;(9)																											10	
jb	.L7 ;(10)																											01	
movapd	(%rsi), %xmm1 ;(11) load																											0-	
movapd	(%rax), %xmm3 ;(12) load																											0-	
addl	$1, %edx ;(13)																											10	
mulpd	%xmm2, %xmm1 ;(14)																											10	
addq	$16, %rax ;(15)																											10	
addq	$16, %rsi ;(16)																											10	
addpd	%xmm3, %xmm1 ;(17)																											00	
movaps	%xmm1,-16(%rax);(18) store																											00	
cmpl	%edi, %edx ;(19)																											10	
jb	.L7 ;(20)																											01	

Situación al finalizar el ciclo 3. Se predice saltar con antelación

instr.	fuente, fuente/destino	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	ok	ROB
movapd	(%rsi), %xmm1 ;(1) load	IF	ID	E																								1-	1-x1- -0-x-0
movapd	**(%rax), %xmm3 ;(2) load**	IF	ID																									1-	2-x3- -0-i-0
addl	$1, %edx ;(3)	IF	ID	E																								11	3-dx- -0-x-0
mulpd	%xmm2, %xmm1 ;(4)	IF	ID																									10	4-x1- -0-i-0
addq	**$16, %rax ;(5)**		IF	ID																								11	5-ax- -0-i-0
addq	**$16, %rsi ;(6)**		IF	ID																								11	6-si- -0-i-0
addpd	%xmm3, %xmm1 ;(7)		IF	ID																								00	7-x1- -0-i-0
movaps	%xmm1,-16(%rax);(8) store		IF	ID																								00	8- - -0-i-0
cmpl	%edi, %edx ;(9)			IF																								10	
jb	.L7 ;(10)			IF																								01	
movapd	(%rsi), %xmm1 ;(11) load			IF																								0-	
movapd	(%rax), %xmm3 ;(12) load			IF																								0-	
addl	$1, %edx ;(13)																											10	
mulpd	%xmm2, %xmm1 ;(14)																											10	
addq	$16, %rax ;(15)																											10	
addq	$16, %rsi ;(16)																											10	
addpd	%xmm3, %xmm1 ;(17)																											00	
movaps	%xmm1,-16(%rax);(18) store																											00	
cmpl	%edi, %edx ;(19)																											10	
jb	.L7 ;(20)																											01	

Situación al finalizar el ciclo 4 (almacenamiento y etapas usadas)

instr.	fuente, fuente/destino	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	ok	ROB
movapd	(%rsi), %xmm1 ;(1) load	IF	ID	E	EX	EX																						1-	1-x1- -0-x-0
movapd	(%rax), %xmm3 ;(2) load	IF	ID		E																							1-	2-x3- -0-x-0
addl	$1, %edx ;(3)	IF	ID	E	EX																							11	3-dx-v-1-f-0
mulpd	%xmm2, %xmm1 ;(4)	IF	ID																									10	4-x1- -0-i-0
addq	$16, %rax ;(5)		IF	ID	E																							11	5-ax- -0-x-0
addq	$16, %rsi ;(6)		IF	ID	E																							11	6-si- -0-x-0
addpd	%xmm3, %xmm1 ;(7)		IF	ID																								00	7-x1- -0-i-0
movaps	%xmm1,-16(%rax);(8) store		IF	ID																								00	8- - -0-i-0
cmpl	**%edi, %edx ;(9)**			IF	ID																							11	9-dx- -0-i-0
jb	.L7 ;(10)			IF	ID																							01	10- - -0-i-0
movapd	(%rsi), %xmm1 ;(11) load			IF	ID																							0-	11-x1- -0-i-0
movapd	(%rax), %xmm3 ;(12) load			IF	ID																							0-	12-x3- -0-i-0
addl	$1, %edx ;(13)				IF																							10	
mulpd	%xmm2, %xmm1 ;(14)				IF																							10	
addq	$16, %rax ;(15)				IF																							10	
addq	$16, %rsi ;(16)				IF																							10	
addpd	%xmm3, %xmm1 ;(17)																											00	
movaps	%xmm1,-16(%rax);(18) store																											00	
cmpl	%edi, %edx ;(19)																											10	
jb	.L7 ;(20)																											01	

Situación al finalizar el ciclo 5 (almacenamiento y etapas usadas)

instr.	fuente, fuente/destino	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	ok	ROB
movapd	(%rsi), %xmm1 ;(1) load	IF	ID	E	EX	EX																						1-	1-x1-v-1-f-0
movapd	(%rax), %xmm3 ;(2) load	IF	ID		E	EX	EX																					1-	2-x3- -0-x-0
addl	$1, %edx ;(3)	IF	ID	E	EX																							11	3-dx-v-1-f-0
mulpd	**%xmm2, %xmm1 ;(4)**	IF	ID																									11	4-x1- -0-i-0
addq	$16, %rax ;(5)		IF	ID	E	EX																						11	5-ax-v-1-f-0
addq	$16, %rsi ;(6)		IF	ID	E	EX																						11	6-si-v-1-f-0
addpd	%xmm3, %xmm1 ;(7)		IF	ID																								00	7-x1- -0-i-0
movaps	%xmm1,-16(%rax);(8) store		IF	ID																								01	8- - -0-i-0
cmpl	%edi, %edx ;(9)			IF	ID	E																						11	9-dx- -0-x-0
jb	.L7 ;(10)			IF	ID																							01	10- - -0-i-0
movapd	**(%rsi), %xmm1 ;(11) load**			IF	ID																							1-	11-x1- -0-i-0
movapd	(%rax), %xmm3 ;(12) load			IF	ID																							1-	12-x3- -0-i-0
addl	$1, %edx ;(13)				IF	ID																						10	13-dx--0-i-0
mulpd	%xmm2, %xmm1 ;(14)				IF	ID																						10	14-x1- -0-i-0
addq	**$16, %rax ;(15)**				IF	ID																						11	15-ax- -0-i-0
addq	**$16, %rsi ;(16)**				IF	ID																						11	16-si- -0-i-0
addpd	%xmm3, %xmm1 ;(17)					IF																						00	
movaps	%xmm1,-16(%rax);(18) store					IF																						00	
cmpl	%edi, %edx ;(19)					IF																						10	
jb	.L7 ;(20)					IF																						01	

Situación al finalizar el ciclo 6 (almacenamiento y etapas usadas)

instr.	fuente, fuente/destino	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	ok	ROB
movapd	(%rsi), %xmm1 ;(1) load	IF	ID	E	EX	EX	W																					1-	1-x1-v-1-f-0
movapd	(%rax), %xmm3 ;(2) load	IF	ID		E	EX	EX																					1-	2-x3-v-1-f-0
addl	$1, %edx ;(3)	IF	ID	E	EX																							11	3-dx-v-1-f-0
mulpd	%xmm2, %xmm1 ;(4)	IF	ID				E																					11	4-x1- -0-x-0
addq	$16, %rax ;(5)		IF	ID	E	EX																						11	5-ax-v-1-f-0
addq	$16, %rsi ;(6)		IF	ID	E	EX																						11	6-si-v-1-f-0
addpd	%xmm3, %xmm1 ;(7)		IF	ID																								10	7-x1- -0-i-0
movaps	%xmm1,-16(%rax);(8) store		IF	ID																								01	8- - -0-i-0
cmpl	%edi, %edx ;(9)			IF	ID	E	EX																					11	9-dx-v-1-f-0
jb	**.L7 ;(10)**			IF	ID																							11	10- - -0-i-0
movapd	(%rsi), %xmm1 ;(11) load			IF	ID		E																					1-	11-x1--0-x-0
movapd	**(%rax), %xmm3 ;(12) load**			IF	ID																							1-	12-x3- -0-i-0
addl	**$1, %edx ;(13)**				IF	ID																						11	13-dx--0-i-0
mulpd	%xmm2, %xmm1 ;(14)				IF	ID																						10	14-x1- -0-i-0
addq	$16, %rax ;(15)				IF	ID	E																					11	15-ax--0-x-0
addq	$16, %rsi ;(16)				IF	ID	E																					11	16-si- -0-x-0
addpd	%xmm3, %xmm1 ;(17)					IF	ID																					00	17-x1- -0-i-0
movaps	%xmm1,-16(%rax);(18) store					IF	ID																					00	18- - -0-i-0
cmpl	**%edi, %edx ;(19)**					IF	ID																					11	19-dx- -0-i-0
jb	.L7 ;(20)					IF	ID																					01	20- - -0-i-0

Situación al finalizar el ciclo 7 (almacenamiento y etapas usadas)

instr.	fuente, fuente/destino	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	ok	ROB
movapd	(%rsi), %xmm1 ;(1) load	IF	ID	E	EX	EX	W																					1-	1-x1-v-1-f-0
movapd	(%rax), %xmm3 ;(2) load	IF	ID		E	EX	EX	W																				1-	2-x3-v-1-f-0
addl	$1, %edx ;(3)	IF	ID	E	EX			W																				11	3-dx-v-1-f-0
mulpd	%xmm2, %xmm1 ;(4)	IF	ID				E	EX	EX	EX	EX																	11	4-x1- -0-x-0
addq	$16, %rax ;(5)		IF	ID	E	EX																						11	5-ax-v-1-f-0
addq	$16, %rsi ;(6)		IF	ID	E	EX																						11	6-si-v-1-f-0
addpd	%xmm3, %xmm1 ;(7)		IF	ID																								10	7-x1- -0-i-0
movaps	%xmm1,-16(%rax);(8) store		IF	ID																								01	8- - -0-i-0
cmpl	%edi, %edx ;(9)			IF	ID	E	EX																					11	9-dx-v-1-f-0
jb	.L7 ;(10)			IF	ID			E																				11	10- - -0-x-0
movapd	(%rsi), %xmm1 ;(11) load			IF	ID		E	EX	EX																			1-	11-x1--0-x-0
movapd	(%rax), %xmm3 ;(12) load			IF	ID			E																				1-	12-x3- -0-x-0
addl	$1, %edx ;(13)				IF	ID		E																				11	13-dx--0-x-0
mulpd	%xmm2, %xmm1 ;(14)				IF	ID																						10	14-x1- -0-i-0
addq	$16, %rax ;(15)				IF	ID	E	EX																				11	15-ax-v-1-f-0
addq	$16, %rsi ;(16)				IF	ID	E	EX																				11	16-si-v-1-f-0
addpd	%xmm3, %xmm1 ;(17)					IF	ID																					00	17-x1- -0-i-0
movaps	%xmm1,-16(%rax);(18) store					IF	ID																					01	18- - -0-i-0
cmpl	%edi, %edx ;(19)					IF	ID	E																				11	19-dx- -0-x-0
jb	.L7 ;(20)					IF	ID																					01	20- - -0-i-0

Situación al finalizar el ciclo 8. Predicción errónea

instr.	fuente, fuente/destino	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	ok	ROB
movapd	(%rsi), %xmm1 ;(1) load	IF	ID	E	EX	EX	W																					1-	1-x1-v-1-f-0
movapd	(%rax), %xmm3 ;(2) load	IF	ID		E	EX	EX	W																				1-	2-x3-v-1-f-0
addl	$1, %edx ;(3)	IF	ID	E	EX			W																				11	3-dx-v-1-f-0
mulpd	%xmm2, %xmm1 ;(4)	IF	ID				E	EX	EX	EX	EX																	11	4-x1- -0-x-0
addq	$16, %rax ;(5)		IF	ID	E	EX																						11	5-ax-v-1-f-0
addq	$16, %rsi ;(6)		IF	ID	E	EX																						11	6-si-v-1-f-0
addpd	%xmm3, %xmm1 ;(7)		IF	ID																								10	7-x1- -0-i-0
movaps	%xmm1,-16(%rax);(8) store		IF	ID																								01	8- - -0-i-0
cmpl	%edi, %edx ;(9)			IF	ID	E	EX																					11	9-dx-v-1-f-0
jb	.L7 ;(10)			IF	ID			E	EX																			11	10- -v-1-f-0
movapd	(%rsi), %xmm1 ;(11) load			IF	ID		E	EX	EX																			1-	11-x1-v-1-f-1
movapd	(%rax), %xmm3 ;(12) load			IF	ID			E	EX	EX																		1-	12-x3- -0-x-1
addl	$1, %edx ;(13)				IF	ID		E	EX																			11	13-dx-v-1-f-1
mulpd	**%xmm2, %xmm1 ;(14)**				IF	ID																						11	14-x1- -0-i-1
addq	$16, %rax ;(15)				IF	ID	E	EX																				11	15-ax-v-1-f-1
addq	$16, %rsi ;(16)				IF	ID	E	EX																				11	16-si-v-1-f-1
addpd	%xmm3, %xmm1 ;(17)					IF	ID																					00	17-x1- -0-i-1
movaps	%xmm1,-16(%rax);(18) store					IF	ID																					01	18- - -0-i-1
cmpl	%edi, %edx ;(19)					IF	ID	E	EX																			11	19-dx-v-1-f-1
jb	.L7 ;(20)					IF	ID																					11	20- - -0-i-1

Situación al finalizar el ciclo 8. Predicción correcta

instr.	fuente, fuente/destino	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	ok	ROB
movapd	(%rsi), %xmm1 ;(1) load	IF	ID	E	EX	EX	W																					1-	1-x1-v-1-f-0
movapd	(%rax), %xmm3 ;(2) load	IF	ID		E	EX	EX	W																				1-	2-x3-v-1-f-0
addl	$1, %edx ;(3)	IF	ID	E	EX			W																				11	3-dx-v-1-f-0
mulpd	%xmm2, %xmm1 ;(4)	IF	ID				E	EX	EX	EX	EX																	11	4-x1- -0-x-0
addq	$16, %rax ;(5)		IF	ID	E	EX																						11	5-ax-v-1-f-0
addq	$16, %rsi ;(6)		IF	ID	E	EX																						11	6-si-v-1-f-0
addpd	%xmm3, %xmm1 ;(7)		IF	ID																								10	7-x1- -0-i-0
movaps	%xmm1,-16(%rax);(8) store		IF	ID																								01	8- - -0-i-0
cmpl	%edi, %edx ;(9)			IF	ID	E	EX																					11	9-dx-v-1-f-0
jb	.L7 ;(10)			IF	ID			E	EX																			11	10- -v-1-f-0
movapd	(%rsi), %xmm1 ;(11) load			IF	ID		E	EX	EX																			1-	11-x1-v-1-f-0
movapd	(%rax), %xmm3 ;(12) load			IF	ID			E	EX	EX																		1-	12-x3- -0-x-0
addl	$1, %edx ;(13)				IF	ID		E	EX																			11	13-dx-v-1-f-0
mulpd	**%xmm2, %xmm1 ;(14)**				IF	ID																						11	14-x1- -0-i-0
addq	$16, %rax ;(15)				IF	ID	E	EX																				11	15-ax-v-1-f-0
addq	$16, %rsi ;(16)				IF	ID	E	EX																				11	16-si-v-1-f-0
addpd	%xmm3, %xmm1 ;(17)					IF	ID																					00	17-x1- -0-i-0
movaps	%xmm1,-16(%rax);(18) store					IF	ID																					01	18- - -0-i-0
cmpl	%edi, %edx ;(19)					IF	ID	E	EX																			11	19-dx-v-1-f-0
jb	**.L7 ;(20)**					IF	ID																					11	20- - -0-i-0

Situación al finalizar el ciclo 9 (almacenamiento y etapas usadas)

instr.	fuente, fuente/destino	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	ok	ROB
movapd	(%rsi), %xmm1 ;(1) load	IF	ID	E	EX	EX	W																					1-	1-x1-v-1-f-0
movapd	(%rax), %xmm3 ;(2) load	IF	ID		E	EX	EX	W																				1-	2-x3-v-1-f-0
addl	$1, %edx ;(3)	IF	ID	E	EX			W																				11	3-dx-v-1-f-0
mulpd	%xmm2, %xmm1 ;(4)	IF	ID				E	EX	EX	EX	EX																	11	4-x1- -0-x-0
addq	$16, %rax ;(5)		IF	ID	E	EX																						11	5-ax-v-1-f-0
addq	$16, %rsi ;(6)		IF	ID	E	EX																						11	6-si-v-1-f-0
addpd	%xmm3, %xmm1 ;(7)		IF	ID																								10	7-x1- -0-i-0
movaps	%xmm1,-16(%rax);(8) store		IF	ID																								01	8- - -0-i-0
cmpl	%edi, %edx ;(9)			IF	ID	E	EX																					11	9-dx-v-1-f-0
jb	.L7 ;(10)			IF	ID			E	EX																			11	10- -v-1-f-0
movapd	(%rsi), %xmm1 ;(11) load			IF	ID		E	EX	EX																			1-	11-x1-v-1-f-0
movapd	(%rax), %xmm3 ;(12) load			IF	ID			E	EX	EX																		1-	12-x3-v-1-f-0
addl	$1, %edx ;(13)				IF	ID		E	EX																			11	13-dx-v-1-f-0
mulpd	%xmm2, %xmm1 ;(14)				IF	ID				E																		11	14-x1- -0-x-0
addq	$16, %rax ;(15)				IF	ID	E	EX																				11	15-ax-v-1-f-0
addq	$16, %rsi ;(16)				IF	ID	E	EX																				11	16-si-v-1-f-0
addpd	%xmm3, %xmm1 ;(17)					IF	ID																					10	17-x1- -0-i-0
movaps	%xmm1,-16(%rax);(18) store					IF	ID																					01	18- - -0-i-0
cmpl	%edi, %edx ;(19)					IF	ID	E	EX																			11	19-dx-v-1-f-0
jb	.L7 ;(20)					IF	ID			E																		11	20- - -0-x-0

Situación al finalizar el ciclo 10 (almacenamiento y etapas usadas)

instr.	fuente, fuente/destino	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	ok	ROB
movapd	(%rsi), %xmm1 ;(1) load	IF	ID	E	EX	EX	W																					1-	1-x1-v-1-f-0
movapd	(%rax), %xmm3 ;(2) load	IF	ID		E	EX	EX	W																				1-	2-x3-v-1-f-0
addl	$1, %edx ;(3)	IF	ID	E	EX			W																				11	3-dx-v-1-f-0
mulpd	%xmm2, %xmm1 ;(4)	IF	ID				E	EX	EX	EX	EX																	11	4-x1-v-1-f-0
addq	$16, %rax ;(5)		IF	ID	E	EX																						11	5-ax-v-1-f-0
addq	$16, %rsi ;(6)		IF	ID	E	EX																						11	6-si-v-1-f-0
addpd	**%xmm3, %xmm1 ;(7)**		IF	ID																								11	7-x1- -0-i-0
movaps	%xmm1,-16(%rax);(8) store		IF	ID																								01	8- - -0-i-0
cmpl	%edi, %edx ;(9)			IF	ID	E	EX																					11	9-dx-v-1-f-0
jb	.L7 ;(10)			IF	ID			E	EX																			11	10- -v-1-f-0
movapd	(%rsi), %xmm1 ;(11) load			IF	ID		E	EX	EX																			1-	11-x1-v-1-f-0
movapd	(%rax), %xmm3 ;(12) load			IF	ID			E	EX	EX																		1-	12-x3-v-1-f-0
addl	$1, %edx ;(13)				IF	ID		E	EX																			11	13-dx-v-1-f-0
mulpd	%xmm2, %xmm1 ;(14)				IF	ID				E	EX	EX	EX	EX														11	14-x1- -0-x-0
addq	$16, %rax ;(15)				IF	ID	E	EX																				11	15-ax-v-1-f-0
addq	$16, %rsi ;(16)				IF	ID	E	EX																				11	16-si-v-1-f-0
addpd	%xmm3, %xmm1 ;(17)					IF	ID																					10	17-x1- -0-i-0
movaps	%xmm1,-16(%rax);(18) store					IF	ID																					01	18- - -0-i-0
cmpl	%edi, %edx ;(19)					IF	ID	E	EX																			11	19-dx-v-1-f-0
jb	.L7 ;(20)					IF	ID			E	EX																	11	20- -v-1-f-0

Situación al finalizar el ciclo 11 (almacenamiento y etapas usadas)

instr.	fuente, fuente/destino	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	ok	ROB
movapd	(%rsi), %xmm1 ;(1) load	IF	ID	E	EX	EX	W																					1-	1-x1-v-1-f-0
movapd	(%rax), %xmm3 ;(2) load	IF	ID		E	EX	EX	W																				1-	2-x3-v-1-f-0
addl	$1, %edx ;(3)	IF	ID	E	EX			W																				11	3-dx-v-1-f-0
mulpd	%xmm2, %xmm1 ;(4)	IF	ID				E	EX	EX	EX	EX	W																11	4-x1-v-1-f-0
addq	$16, %rax ;(5)		IF	ID	E	EX						W																11	5-ax-v-1-f-0
addq	$16, %rsi ;(6)		IF	ID	E	EX						W																11	6-si-v-1-f-0
addpd	**%xmm3, %xmm1 ;(7)**		IF	ID								E	EX	EX														11	7-x1- -0-x-0
movaps	%xmm1,-16(%rax);(8) store		IF	ID																								01	8- - -0-i-0
cmpl	%edi, %edx ;(9)			IF	ID	E	EX																					11	9-dx-v-1-f-0
jb	.L7 ;(10)			IF	ID			E	EX																			11	10- -v-1-f-0
movapd	(%rsi), %xmm1 ;(11) load			IF	ID		E	EX	EX																			1-	11-x1-v-1-f-0
movapd	(%rax), %xmm3 ;(12) load			IF	ID			E	EX	EX																		1-	12-x3-v-1-f-0
addl	$1, %edx ;(13)				IF	ID		E	EX																			11	13-dx-v-1-f-0
mulpd	%xmm2, %xmm1 ;(14)				IF	ID				E	EX	EX	EX	EX														11	14-x1- -0-x-0
addq	$16, %rax ;(15)				IF	ID	E	EX																				11	15-ax-v-1-f-0
addq	$16, %rsi ;(16)				IF	ID	E	EX																				11	16-si-v-1-f-0
addpd	%xmm3, %xmm1 ;(17)					IF	ID																					10	17-x1- -0-i-0
movaps	%xmm1,-16(%rax);(18) store					IF	ID																					01	18- - -0-i-0
cmpl	%edi, %edx ;(19)					IF	ID	E	EX																			11	19-dx-v-1-f-0
jb	.L7 ;(20)					IF	ID			E	EX																	11	20- -v-1-f-0

Situación al finalizar el ciclo 13 (almacenamiento y etapas usadas)

instr.	fuente, fuente/destino	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	ok	ROB
movapd	(%rsi), %xmm1 ;(1) load	IF	ID	E	EX	EX	W																					1-	1-x1-v-1-f-0
movapd	(%rax), %xmm3 ;(2) load	IF	ID		E	EX	EX	W																				1-	2-x3-v-1-f-0
addl	$1, %edx ;(3)	IF	ID	E	EX			W																				11	3-dx-v-1-f-0
mulpd	%xmm2, %xmm1 ;(4)	IF	ID				E	EX	EX	EX	EX	W																11	4-x1- v-1-f-0
addq	$16, %rax ;(5)		IF	ID	E	EX						W																11	5-ax-v-1-f-0
addq	$16, %rsi ;(6)		IF	ID	E	EX						W																11	6-si-v-1-f-0
addpd	%xmm3, %xmm1 ;(7)		IF	ID								E	EX	EX														11	7-x1-v-1-f-0
movaps	**%xmm1,-16(%rax);(8) store**		IF	ID																								11	8- - -0-i-0
cmpl	%edi, %edx ;(9)			IF	ID	E	EX																					11	9-dx-v-1-f-0
jb	.L7 ;(10)			IF	ID			E	EX																			11	10- -v-1-f-0
movapd	(%rsi), %xmm1 ;(11) load			IF	ID		E	EX	EX																			1-	11-x1-v-1-f-0
movapd	(%rax), %xmm3 ;(12) load			IF	ID			E	EX	EX																		1-	12-x3-v-1-f-0
addl	$1, %edx ;(13)				IF	ID		E	EX																			11	13-dx-v-1-f-0
mulpd	%xmm2, %xmm1 ;(14)				IF	ID				E	EX	EX	EX	EX														11	14-x1-v-1-f-0
addq	$16, %rax ;(15)				IF	ID	E	EX																				11	15-ax-v-1-f-0
addq	$16, %rsi ;(16)				IF	ID	E	EX																				11	16-si-v-1-f-0
addpd	**%xmm3, %xmm1 ;(17)**					IF	ID																					11	17-x1- -0-i-0
movaps	%xmm1,-16(%rax);(18) store					IF	ID																					01	18- - -0-i-0
cmpl	%edi, %edx ;(19)					IF	ID	E	EX																			11	19-dx-v-1-f-0
jb	.L7 ;(20)					IF	ID			E	EX																	11	20- -v-1-f-0

Situación al finalizar el ciclo 14 (almacenamiento y etapas usadas)

instr.	fuente, fuente/destino	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	ok	ROB
movapd	(%rsi), %xmm1 ;(1) load	IF	ID	E	EX	EX	W																					1-	1-x1-v-1-f-0
movapd	(%rax), %xmm3 ;(2) load	IF	ID		E	EX	EX	W																				1-	2-x3-v-1-f-0
addl	$1, %edx ;(3)	IF	ID	E	EX			W																				11	3-dx-v-1-f-0
mulpd	%xmm2, %xmm1 ;(4)	IF	ID				E	EX	EX	EX	EX	W																11	4-x1- v-1-f-0
addq	$16, %rax ;(5)		IF	ID	E	EX						W																11	5-ax-v-1-f-0
addq	$16, %rsi ;(6)		IF	ID	E	EX						W																11	6-si-v-1-f-0
addpd	%xmm3, %xmm1 ;(7)		IF	ID								E	EX	EX	W													11	7-x1-v-1-f-0
movaps	%xmm1,-16(%rax);(8) store		IF	ID											E													11	8- -v-1-f-0
cmpl	%edi, %edx ;(9)			IF	ID	E	EX																					11	9-dx-v-1-f-0
jb	.L7 ;(10)			IF	ID			E	EX																			11	10- -v-1-f-0
movapd	(%rsi), %xmm1 ;(11) load			IF	ID		E	EX	EX																			1-	11-x1-v-1-f-0
movapd	(%rax), %xmm3 ;(12) load			IF	ID			E	EX	EX																		1-	12-x3-v-1-f-0
addl	$1, %edx ;(13)				IF	ID		E	EX																			11	13-dx-v-1-f-0
mulpd	%xmm2, %xmm1 ;(14)				IF	ID				E	EX	EX	EX	EX															11	14-x1-v-1-f-0
addq	$16, %rax ;(15)				IF	ID	E	EX																				11	15-ax-v-1-f-0
addq	$16, %rsi ;(16)				IF	ID	E	EX																				11	16-si-v-1-f-0
addpd	%xmm3, %xmm1 ;(17)					IF	ID								E	EX	EX											11	17-x1- -0-x-0
movaps	%xmm1,-16(%rax);(18) store					IF	ID																					01	18- - -0-i-0
cmpl	%edi, %edx ;(19)					IF	ID	E	EX																			11	19-dx-v-1-f-0
jb	.L7 ;(20)					IF	ID			E	EX																	11	20- -v-1-f-0

Situación al finalizar el ciclo 15 (almacenamiento y etapas usadas)

instr.	fuente, fuente/destino	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	ok	ROB
movapd	(%rsi), %xmm1 ;(1) load	IF	ID	E	EX	EX	W																					1-	1-x1-v-1-f-0
movapd	(%rax), %xmm3 ;(2) load	IF	ID		E	EX	EX	W																				1-	2-x3-v-1-f-0
addl	$1, %edx ;(3)	IF	ID	E	EX			W																				11	3-dx-v-1-f-0
mulpd	%xmm2, %xmm1 ;(4)	IF	ID				E	EX	EX	EX	EX	W																11	4-x1- v-1-f-0
addq	$16, %rax ;(5)		IF	ID	E	EX						W																11	5-ax-v-1-f-0
addq	$16, %rsi ;(6)		IF	ID	E	EX						W																11	6-si-v-1-f-0
addpd	%xmm3, %xmm1 ;(7)		IF	ID								E	EX	EX	W													11	7-x1-v-1-f-0
movaps	%xmm1,-16(%rax);(8) store		IF	ID											E	W												11	8- -v-1-f-0
cmpl	%edi, %edx ;(9)			IF	ID	E	EX									W												11	9-dx-v-1-f-0
jb	.L7 ;(10)			IF	ID			E	EX							W												11	10- -v-1-f-0
movapd	(%rsi), %xmm1 ;(11) load			IF	ID		E	EX	EX							W												1-	11-x1-v-1-f-0
movapd	(%rax), %xmm3 ;(12) load			IF	ID			E	EX	EX																		1-	12-x3-v-1-f-0
addl	$1, %edx ;(13)				IF	ID		E	EX																			11	13-dx-v-1-f-0
mulpd	%xmm2, %xmm1 ;(14)				IF	ID				E	EX	EX	EX	EX														11	14-x1-v-1-f-0
addq	$16, %rax ;(15)				IF	ID	E	EX																				11	15-ax-v-1-f-0
addq	$16, %rsi ;(16)				IF	ID	E	EX																				11	16-si-v-1-f-0
addpd	%xmm3, %xmm1 ;(17)					IF	ID								E	EX	EX											11	17-x1- -0-x-0
movaps	%xmm1,-16(%rax);(18) store					IF	ID																					01	18- - -0-i-0
cmpl	%edi, %edx ;(19)					IF	ID	E	EX																			11	19-dx-v-1-f-0
jb	.L7 ;(20)					IF	ID			E	EX																	11	20- -v-1-f-0

Situación al finalizar el ciclo 16 (almacenamiento y etapas usadas)

instr.	fuente, fuente/destino	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	ok	ROB
movapd	(%rsi), %xmm1 ;(1) load	IF	ID	E	EX	EX	W																					1-	1-x1-v-1-f-0
movapd	(%rax), %xmm3 ;(2) load	IF	ID		E	EX	EX	W																				1-	2-x3-v-1-f-0
addl	$1, %edx ;(3)	IF	ID	E	EX			W																				11	3-dx-v-1-f-0
mulpd	%xmm2, %xmm1 ;(4)	IF	ID				E	EX	EX	EX	EX	W																11	4-x1- v-1-f-0
addq	$16, %rax ;(5)		IF	ID	E	EX						W																11	5-ax-v-1-f-0
addq	$16, %rsi ;(6)		IF	ID	E	EX						W																11	6-si-v-1-f-0
addpd	%xmm3, %xmm1 ;(7)		IF	ID								E	EX	EX	W													11	7-x1-v-1-f-0
movaps	%xmm1,-16(%rax);(8) store		IF	ID											E	W												11	8- -v-1-f-0
cmpl	%edi, %edx ;(9)			IF	ID	E	EX									W												11	9-dx-v-1-f-0
jb	.L7 ;(10)			IF	ID			E	EX							W												11	10- -v-1-f-0
movapd	(%rsi), %xmm1 ;(11) load			IF	ID		E	EX	EX							W												1-	11-x1-v-1-f-0
movapd	(%rax), %xmm3 ;(12) load			IF	ID			E	EX	EX							W											1-	12-x3-v-1-f-0
addl	$1, %edx ;(13)				IF	ID		E	EX								W											11	13-dx-v-1-f-0
mulpd	%xmm2, %xmm1 ;(14)				IF	ID				E	EX	EX	EX	EX			W											11	14-x1-v-1-f-0
addq	$16, %rax ;(15)				IF	ID	E	EX									W											11	15-ax-v-1-f-0
addq	$16, %rsi ;(16)				IF	ID	E	EX																				11	16-si-v-1-f-0
addpd	%xmm3, %xmm1 ;(17)					IF	ID								E	EX	EX											11	17-x1-v-1-f-0
movaps	%xmm1,-16(%rax);(18) store					IF	ID																					11	18- - -0-i-0
cmpl	%edi, %edx ;(19)					IF	ID	E	EX																			11	19-dx-v-1-f-0
jb	.L7 ;(20)					IF	ID			E	EX																	11	20- -v-1-f-0

Situación al finalizar el ciclo 17 (almacenamiento y etapas usadas)

instr.	fuente, fuente/destino	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	ok	ROB
movapd	(%rsi), %xmm1 ;(1) load	IF	ID	E	EX	EX	W																					1-	1-x1-v-1-f-0
movapd	(%rax), %xmm3 ;(2) load	IF	ID		E	EX	EX	W																				1-	2-x3-v-1-f-0
addl	$1, %edx ;(3)	IF	ID	E	EX			W																				11	3-dx-v-1-f-0
mulpd	%xmm2, %xmm1 ;(4)	IF	ID				E	EX	EX	EX	EX	W																11	4-x1- v-1-f-0
addq	$16, %rax ;(5)		IF	ID	E	EX						W																11	5-ax-v-1-f-0
addq	$16, %rsi ;(6)		IF	ID	E	EX						W																11	6-si-v-1-f-0
addpd	%xmm3, %xmm1 ;(7)		IF	ID								E	EX	EX	W													11	7-x1-v-1-f-0
movaps	%xmm1,-16(%rax);(8) store		IF	ID											E	W												11	8- -v-1-f-0
cmpl	%edi, %edx ;(9)			IF	ID	E	EX									W												11	9-dx-v-1-f-0
jb	.L7 ;(10)			IF	ID			E	EX							W												11	10- -v-1-f-0
movapd	(%rsi), %xmm1 ;(11) load			IF	ID		E	EX	EX							W												1-	11-x1-v-1-f-0
movapd	(%rax), %xmm3 ;(12) load			IF	ID			E	EX	EX							W											1-	12-x3-v-1-f-0
addl	$1, %edx ;(13)				IF	ID		E	EX								W											11	13-dx-v-1-f-0
mulpd	%xmm2, %xmm1 ;(14)				IF	ID				E	EX	EX	EX	EX			W											11	14-x1-v-1-f-0
addq	$16, %rax ;(15)				IF	ID	E	EX									W											11	15-ax-v-1-f-0
addq	$16, %rsi ;(16)				IF	ID	E	EX										W										11	16-si-v-1-f-0
addpd	%xmm3, %xmm1 ;(17)					IF	ID								E	EX	EX	W										11	17-x1-v-1-f-0
movaps	%xmm1,-16(%rax);(18) store					IF	ID											E										11	18- -v-1-f-0
cmpl	%edi, %edx ;(19)					IF	ID	E	EX																			11	19-dx-v-1-f-0
jb	.L7 ;(20)					IF	ID			E	EX																	11	20- -v-1-f-0

Situación al finalizar el ciclo 18 (almacenamiento y etapas usadas)

instr.	fuente, fuente/destino	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	ok	ROB
movapd	(%rsi), %xmm1 ;(1) load	IF	ID	E	EX	EX	W																					1-	1-x1-v-1-f-0
movapd	(%rax), %xmm3 ;(2) load	IF	ID		E	EX	EX	W																				1-	2-x3-v-1-f-0
addl	$1, %edx ;(3)	IF	ID	E	EX			W																				11	3-dx-v-1-f-0
mulpd	%xmm2, %xmm1 ;(4)	IF	ID				E	EX	EX	EX	EX	W																11	4-x1- v-1-f-0
addq	$16, %rax ;(5)		IF	ID	E	EX						W																11	5-ax-v-1-f-0
addq	$16, %rsi ;(6)		IF	ID	E	EX						W																11	6-si-v-1-f-0
addpd	%xmm3, %xmm1 ;(7)		IF	ID								E	EX	EX	W													11	7-x1-v-1-f-0
movaps	%xmm1,-16(%rax);(8) store		IF	ID											E	W												11	8- -v-1-f-0
cmpl	%edi, %edx ;(9)			IF	ID	E	EX									W												11	9-dx-v-1-f-0
jb	.L7 ;(10)			IF	ID			E	EX							W												11	10- -v-1-f-0
movapd	(%rsi), %xmm1 ;(11) load			IF	ID		E	EX	EX							W												1-	11-x1-v-1-f-0
movapd	(%rax), %xmm3 ;(12) load			IF	ID			E	EX	EX							W											1-	12-x3-v-1-f-0
addl	$1, %edx ;(13)				IF	ID		E	EX								W											11	13-dx-v-1-f-0
mulpd	%xmm2, %xmm1 ;(14)				IF	ID				E	EX	EX	EX	EX			W											11	14-x1-v-1-f-0
addq	$16, %rax ;(15)				IF	ID	E	EX									W											11	15-ax-v-1-f-0
addq	$16, %rsi ;(16)				IF	ID	E	EX										W										11	16-si-v-1-f-0
addpd	%xmm3, %xmm1 ;(17)					IF	ID								E	EX	EX	W										11	17-x1-v-1-f-0
movaps	%xmm1,-16(%rax);(18) store					IF	ID											E	W									11	18- -v-1-f-0
cmpl	%edi, %edx ;(19)					IF	ID	E	EX										W									11	19-dx-v-1-f-0
jb	.L7 ;(20)					IF	ID			E	EX								W									11	20- -v-1-f-0

T_{CPU}, CPI

$$T_{CPU} = NI \times CPI \times T_{ciclo} = 18 \times T_{ciclo}$$

$$CPI = \frac{ciclos}{instrucciones} = \frac{18}{20} = 0.9$$

$$T_{CPU}(n\ instr.) = 18 \times T_{ciclo} = T_{LatenciaPrimeraInstr} + T_{RestoInstr}$$
$$= 6 \times T_{ciclo} + (20 - 1) \times CPI \times T_{ciclo}$$

$$CPI = \frac{ciclos}{instrucciones} = \frac{18 - 6}{20 - 1} \approx 0.63$$