$ID(I,f) = ((w_{II1}, (n_{II1}, f)), (w_{II2}, (n_{II2}, f)), \ldots (w_{IIm}, (n_{IIm}, f)))$
$ID(S,f) = ((w_{SI1}, (n_{SI1}, f)), (w_{SI2}, (n_{SI1}, f)), \ldots (w_{SIn}, (n_{SI1}, f))$

f

$W^P = prework$	$W = work$
$M(I, f) = MI$	$M(S, f) = MS$
$e(W^P, f) = prepeek$	$e(W,f) = peek$
$o(W^P, f) = prepop$	$o(W, f) = pop$
$u(W^P, f) = prepush$	$u(W, f) = push$
$C(f) = copydown$	

$OD(I,f) = ((w_{IO1}, d_{IO1}), (w_{IO2}, d_{IO2}), \ldots (w_{IOIr}, d_{IOIr}))$
$OD(S,f) = ((w_{SO1}, d_{SO1}), (w_{SO2}, d_{SO12}), \ldots (w_{SOs}, d_{SOs}))$

(a) The original filter f.

1. Create P copies of f and set their rates and work functions according to (c).
2. Create two identity nodes, ID_I and ID_O.
3. Move the initialization stage computation of f to f_1 according to (c).
4. Move input distribution of f to ID_I replacing occurrences of f with ID_I in edges.
5. Move output distribution of f to ID_O, replacing occurrences of f with ID_O in edges.
6. Create the fission duplication pattern in the output distribution of ID_I.
7. Create a round robin joining pattern for the output identity filter ID_O to receive from each fission product.
8. For each node p that is a producer of f, replace the occurrences of f with ID_I in the edges of the dupsets of p's output distribution.
9. For each node c that is a consumer of f, replace the occurrences of f with ID_O in incoming edges c's input distribution.
10. SYNCHREMOVE(ID_I)
11. SYNCHREMOVE(ID_O)

(b) Steps of fission of f.

Fiss f by P

$ID(I,f) = ((w_{II1}, (n_{II1}, ID_I)), (w_{II2}, (n_{II2}, ID_I)), \ldots (w_{IIm}, (n_{IIm}, ID_I)))$
$ID(S,f) = ((w_{SI1}, (n_{SI1}, ID_I)), (w_{SI2}, (n_{SI1}, ID_I)), \ldots (w_{SIn}, (n_{SI1}, ID_I))$

ID_I

Shorthand Variables:
$dup = peek - pop$
$newpop = MS / P \times pop + dup$
$newpush = MS / P \times push$

$OD(I) = ((1, ((ID_I, F_1)))$
$OD(S) = ((newpop - C(f) - dup, ((ID_I, F_1))), (dup, ((ID_I, F_1), (ID_I, F_2))),$
$(newpop - 2\times dup, ((ID_I, F_2))), (dup, ((ID_I, F_2), (ID_I, F_3))),$
$\ldots,$
$(newpop - 2\times dup, ((ID_I, F_{P-1}))), (dup, ((ID_I, F_{P-1}), (ID_I, F_P))),$
$(newpop - 2\times dup, ((ID_I, F_P))), (dup, ((ID_I, F_P), (ID_I, F_1))),$
$(C(f) - dup, ((ID_I, F_1))))$

$ID(I) = ((1, (ID_I, F_1))$
$ID(S) = ((1, (ID_I, F_1))$

f_1

$M(I) = MI$
$M(S) = MS / P$
$e(W^P) = \max(prepeek,$
$\quad prepop + (MI - 1) \times pop + dup)$
$o(W^P) = prepop + (MI \times pop)$
$u(W^P) = prepush + (MI \times push)$
$e(W) = newpop$
$o(W) = newpop$
$u(W) = newpush$
$C = C(f)$
$W =$
`for (`$M(S,f) / P$`)` *work*
`for (`*dup*`) pop()`
$W^P =$
prework
`for (`$MI - 1$`)` *work*

$OD(I) = ((1, ((F_1, ID_O)))$
$OD(S) = ((1, ((F_1, ID_O)))$

$ID(I) = (), ID(S) = ((1, (ID_I, F_2))$

f_2

$M(I) = 0$
$M(S) = MS / P$
$e(W^P) = 0$
$o(W^P) = 0$
$u(W^P) = 0$
$e(W) = newpop$
$o(W) = newpop$
$u(W) = newpush$
$C = 0$
$W =$
`for (`$M(S,f) / P$`)` *work*
`for (`*dup*`) pop()`
$W^P = \oslash$

$OD(I) = (), OD(S) = ((1, ((F_2, ID_O)))$

…

$ID(I) = (), ID(S) = ((1, (ID_I, F_P))$

f_P

$M(I) = 0$
$M(S) = MS / P$
$e(W^P) = 0$
$o(W^P) = 0$
$u(W^P) = 0$
$e(W) = newpop$
$o(W) = newpop$
$u(W) = newpush$
$C = 0$
$W =$
`for (`$M(S,f) / P$`)` *work*
`for (`*dup*`) pop()`
$W^P = \oslash$

$OD(I) = (), OD(S) = ((1, ((F_P, ID_O)))$

$ID(I) = ((1, (F_1, ID_O))$
$ID(S) = ((newpush, (F_1, ID_O), (newpush, (F_2, ID_O), \ldots, (newpush, (F_P, ID_O))$

ID_O

$OD(I,f) = ((w_{IO1}, d_{IO1}), (w_{IO2}, d_{IO2}), \ldots (w_{IOIr}, d_{IOIr}))$ where ID_O replaces f in edges of d_{IOi}
$OD(S,f) = ((w_{SO1}, d_{SO1}), (w_{SO2}, d_{SO2}), \ldots (w_{SOs}, d_{SOs}))$ where ID_O replaces f in edges of d_{SOi}

(c) Details Steps 1-9 when fissing f by P.