Part 1 kp#3 to mDn@trgt--
slow microstrategy

SLOW-kp/ m: Combined with SLOW-m/ c the "before" way

SLOW-m/ c -- w/ POG to trgt prior to move (see notes in "activities & microstrategies" CPM-GOMS.)
Part 2 mDn@trgt to mDn@exp2 btn
slow microstrategy

START mouse Dn

mouse Up

mvCrsr

verify selection

init mvCrsr

attend target

init POG

verify btn loc

attend crsr@trgt

verify crsr@trgt

initiate mouse Dn

mouse Dn

new crsr loc

perceive zoom-pt selected

display change

PERCEIVE EXP2 btn

PERCEIVE crsr@trgt

eye mvmt

SLOW-c/m

SLOW-m/c

END part 2

Same as best-fitting
Part 3 mDn@exp2 btn to mDn@way-point
slow microstrategy
Part 4 mDn@way-point to KP#2
slow microstrategy
SLOW-kp/m: There are three ways to join SLOW-kp/m with SLOW-m/c. This method is the AFTER way.
Part 1 kp#3 to mDn@zoom-pt--
BEST-fitting microstrategy

1. This puts "mvCrsr" on the critical path. That is a very good thing.
2. Assumes that POG is fixed on center at time of kpDn.
3. "verify display active" is needed
4. another POG not needed to "perceive zoom pt" as zoom-pt is close to center.
5. Will not click until verify that crsr@loc.

NOTE: fastest-reasonable is also the best-fitting ms for this Ss. This is the only case in wh the fastest-reasonable ms differ bet. P1 and P2.

FAST-kp/ m

SLOW-m/c: move to location such as a button, click after verify that cursor is in the button
Part 2 mDn@zoom-pt to mDn@exp2 btn
BEST-fitting microstrategy

SLOW-m/c: move to location such as a button, click after verify that cursor is in the button

SAME as slowest
Part 3 mDn@exp2 btn to mDn@way-point
BEST-fitting microstrategy

MED-c/m: after mDn/Up (before verify); e.g., following mDn on an object such as a button where mUp must take place in the button else the change will not occur.

SLOW-m/c: move to location such as a button, click after verify that cursor is in the button
Part 4 mDn@way-point to KP#2
BEST-fitting microstrategy

START mouse Dn

mouse Up

display \"+\" at trgt loc

perceive trgt selected

perceive finger in position

attend finger pos

verify trgt selected

init POG to key

verify finger position

init KP

KP down

POG to KEY

END

SLOW-c/ kp
Part 5 KP#2 to mDn@NavDes
BEST-fitting microstrategy

SLOW-kp/m: There are three ways to join SLOW-kp/m with SLOW-m/c. This method is the MIDDLE way. The second POG is inited BEFORE to "init mvCrsr" whereas its verify "verify btn loc" comes AFTER "init mvCrsr"
Part 1 kp#3 to mDn@zoom-pt--
fastest-reasonable microstrategy

1. Assume that Ss knows that surface radar display will be infallibly selected by kp (verify not needed)
2. POG to center precedes kp#3; hence "init POG" etc not needed here. Just needs the "attend zoom pt."
3. Must "perceive zoom-pt"
4. This puts "mvCrsr" on the critical path. That is a very good thing.
5. Will not click until verify that crsr@loc.
5. To make this work -- to keep mvCrsr on CP -- requires that the "attend zoom pt" precede "init mvCrsr"

THIS IS THE FASTEST REASONABLE MODEL BECAUSE:
need the perceive to loc zoom-pt.

FAST-kp/ m

SLOW-m/ c: move to location such as a button, click after verify that cursor is in the button
Part 2 mDn@zoom-pt to mDn@exp2 btn fastest-reasonable microstrategy

FAST-c/m: after mDn (during mUp and before verify); e.g., following mDn on a large object or one in which only a mDn is needed.

SLOW-m/c: move to location such as a button, click after verify that cursor is in the button.
Part 3 mDn@exp2 btn to mDn@way-point
fastest-reasonable microstrategy

MED-c/m: after mDn/Up (before verify); e.g,
following mDn on an object such as a button where
mUp must take place in the button else the change will
not occur.

SLOW-m/c: move to location such as a button, click
after verify that cursor is in the button
Part 4 mDn@way-point to KP#2
fastest-reasonable microstrategy

FAST-c/ kp: this is the fastest.
Part 5 KP#2 to mDn@NavDes
fastest-reasonable microstrategy

FAST-kp/m: Assume POG already at display & target is known

SLOW-m/c: move to location such as a button, click after verify that cursor is in the button