```
pro locmoon, time, theta0, radius, orbrate, x=x, y=y, z=z, ang=ang

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;
;; - calculates the coordinates of each moon given a "final" orbital longitude and
;;   a time difference
;; - Where was moon (time) seconds ago?
;;    theta = (-time [s]) * (orbrate [rad/s]) + theta0 [rad]
;;
;; INPUTS
;; * time: Time before moon was at "theta0" (seconds)
;; * theta0: final orbital longitude of each moon (radians)
;; * radius: orbital radius of each moon (R_plan)
;; * orbrate: angular speed of each moon (rad/s)
;;
;; OUTPUTS
;; * x: moon's x-position relative to planet "time" seconds before
;;         it was at "theta0" (R_J)
;; * y:    "   y            "                "
;; * ang: theta  _time_ seconds ago
;;
;;                                +y
;;                                 |        +z is out of the screen (Jup north)
;;                                 |
;;                                 O---------- West,+x
;;                                 |
;;                                 |
;;                                \ /
;;                                 v
;;                         To Earth/Sun
;;
;;  n = n_elements(time) = number of packets
;;  m = n_elements(radius) = number of moons
;;  x = float(n,m) = x position of each moon at each time step requested
;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;Calculate orbital longitude (radians)
i = replicate(1.,n_elements(time))
ang = double(-time)#orbrate + i#double(theta0)  ;;[n#m + n#m]

;Calculate x and y coordinates
x = -(i#radius) * sin(ang)
y = (i#radius) * cos(ang)
z = x * 0. ;; Assume inclination = 0

end
```