The Elements of *Mechanical Design*

by

James G. Skakoon

ASME Press New York 2008

Library of Congress Cataloging-in-Publication Data

Skakoon, James G.
The elements of mechanical design / by James G. Skakoon.
p. cm.
Includes bibliographical references and index.
ISBN 978-0-7918-0267-0
1. Machine design. 2. Engineering design. I. Title.

TJ230.S546 2008
621.8'15--dc22
2007040811

Table of Contents

PART III
SOME PRACTICAL ADVICE

Preface

This book contains principles and practices for mechanical designers. They come from the experience, know-how, and intuition of expert designers, but they represent engineering fundamentals in a practical way.

Consider two examples. Even children quickly learn that carrying two pails of water, one on each side, is easier than carrying a single pail on the right or left. This is not an isolated observation, but a useful principle of design described in this book (self-help). Or haven't we all spilled a drinking glass on a table that teeters side-to-side on two legs, resting now on the third, now on the fourth? Four-legged tables are fundamentally flawed and represent another design principle described in this book (over-constraint). Agreed, these cases are obvious enough. But in this book they become principles and guidelines, explicitly stated, to be applied to other design problems—where they may be rather less obvious.

This book is not about engineering science. Established books exist for machine design, structural analysis, and kinematics. Neither is it about design for manufacturability nor about the design process; excellent books have been written, especially in recent years, for these subjects as well. Nonetheless, despite an increased emphasis on design and manufacturing in both university curricula and practical literature, existing books have little about mechanical design as practiced by experienced designers.

Good designers often understand and use the ideas in this book whether or not they recognize them as distinct principles. Many designers, myself included, learned them either by trial and error or by exploiting colleagues' experience. Perhaps everything here would be part of all designers' practice were they to design long enough. But all too often designers are unaware of or do not fully grasp ideas that experts use to great advantage.

Therefore, in this book I explicitly state design principles and practices: 1) so beginning designers do not have to discover them on their own as I had to, and 2) so all designers can apply them as fundamental concepts throughout their designs.

Although nothing in this book is new, its narrow focus on basic, detail-level mechanical design is unique. Use it as a primer, and a refresher, on good mechanical design.

James G. Skakoon

Acknowledgements

I would like to acknowledge the people who generously agreed to be interviewed for this book:

Benge Ambrogi
Joel Bartholomew
Douglass Blanding
David Chastain
Ron Franklin
Gilbert Fryklund
Larry Gray
Michael Knipfer
Herbert Loeffler
Brian Nelson
David Schuelke

I interviewed designers to confirm and add to the book's content. Comments repeated themselves with remarkable consistency, but many were unique. You can read some of the good ones in Appendix D.

Also generously, Paul D. Lucas, Herbert Loeffler, and Gilbert Fryklund reviewed the manuscript, as did many others anonymously, for which I am grateful.

Part I

Elementary Rules of Mechanical Design

1. Create designs that are explicitly simple—keep complexity intrinsic.

You have heard this often, usually as "Keep it simple!" But for good designers, just keeping it simple is not enough. If you only just keep things simple, you will still have complicated designs. You must simplify, simplify, simplify!

So what makes a design simple? Can your intuition alone judge simplicity? Will you know it when you see it?

The less thought and the less knowledge a device requires, the simpler it is. This applies equally to its production, testing, and use. Use these criteria—how much thought, how much knowledge—to judge your designs. Judge best by comparing one solution to another. Of course, it may take lots of thought and knowledge to get to a design requiring little of either; that is design.

But some devices are elaborate affairs, and you may have to design one. How can it possibly be simple amid great complexity?

What a simple design means is that everyone involved with its production and use sees nothing that looks complicated from his or her own perspective or convention. Complexity is buried and invisible. In other words, there is a hierarchy for knowledge and thought. Each hierarchical level may be intrinsically complex, yet the device remains simple if the complexity resides only within its own level.

Screw threads are a perfect example. Despite their abundant scientific and manufacturing complexity, you and I specify screw threads and threaded fasteners with the click of a mouse. Whatever it takes to make them is largely invisible to designers. We just say, "¼-20."

Another, very contemporary example is part geometry. Complex geometry no longer implies a complicated design. Computerized methods can control tooling and manufacturing processes without human interpretation. Yes, the part's geometry is complicated, yet its production and use may be no more difficult than for one with much simpler geometry. The designer easily understands the geometry; no one else needs to. Complex geometry has a cost, of course, but the hierarchical nature of knowledge permits complexity within a level if for some gain.

Simplicity can be subtle. A good example of keeping things simple is designing symmetry into components. If a part is asymmetric, knowledge and thought are needed to orient it (Figure 1-1). This comes from the assembler or user during hand assembly, or by part orientation and feed equipment during automated assembly. Assembling or using symmetrical parts requires less knowledge and thought than asymmetric parts.

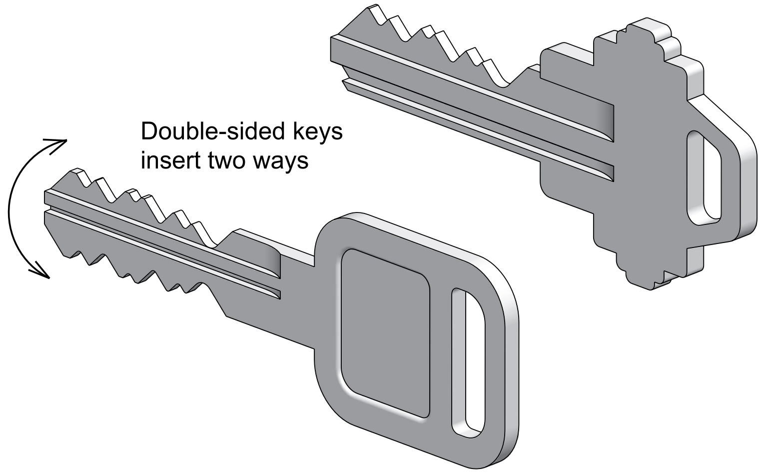

Figure 1-1 Symmetric items are simpler—to use or assemble, even if they are more complicated to produce.

Simplicity can be paradoxical. Symmetry adds information to the component part, thus adding cost; the double-sided key may cost more. So although the symmetrical key is simpler for the user, it is more complicated to manufacture, a different hierarchical level. But manufactured assemblies usually favor assembly simplicity over component simplicity.

Two common techniques for keeping things simple deserve specific mention:

1. Purchasing rather than making components
2. Specifying components by standards

These are good techniques not because they reduce the overall complexity, but because they hide some of it. Substantial information exists, but resides implicitly in components, most of which is no concern to designers.

2. Keep the functions of a design independent from one another.

During the concept development stage of the design process, you will decompose a device or system into basic functions [3, 4]. (If you don't, you should!) After that, a vital part of your mission as a mechanical designer is to keep those functions separate [5, 6].

This is not as easy as it sounds. There will be misunderstandings that unsuitably combine functions and compromise the design. There will be temptations to combine functions and features, resulting in a confusing overlap. Avoid the misunderstandings and resist the temptations.

Figure 2-1 shows a ball-and-socket style locking locator as might be used to hold, for example, vise jaws, dial indicators, or, in this case, a camera.

Two apparent functions are:

1. Define the camera position
2. Lock the camera in place

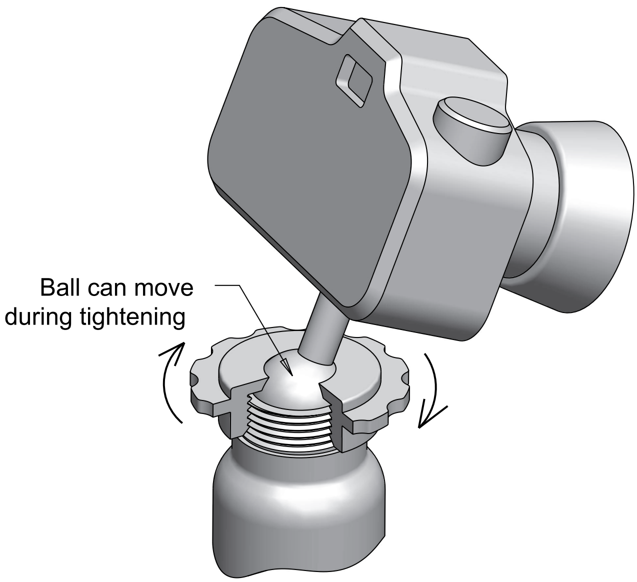

Figure 2-1 Ball-and-socket tripod head for camera. There is no functional independence of positioning and locking.

Although this embodiment is simple—it has only three easily manufactured parts—the above functions are not independent. Tightening the collar moves the ball in the socket because of friction between the nut and the ball. There is a race between the ball-to-socket friction and that of the nut-to-ball. Lose this frictional race, and the ball moves as the nut is tightened—quite perplexing for finely-positioned devices. You get it exactly where you want it, then you find it moved when you locked it down!

Figure 2-2 shows an improvement in separating functions. In this embodiment, tightening the nut creates no rotational friction between nut and ball, which eliminates one dependence. But all things have their cost! This design requires an over-center, or undercut geometry as well as relieving slots for the socket—costly features.

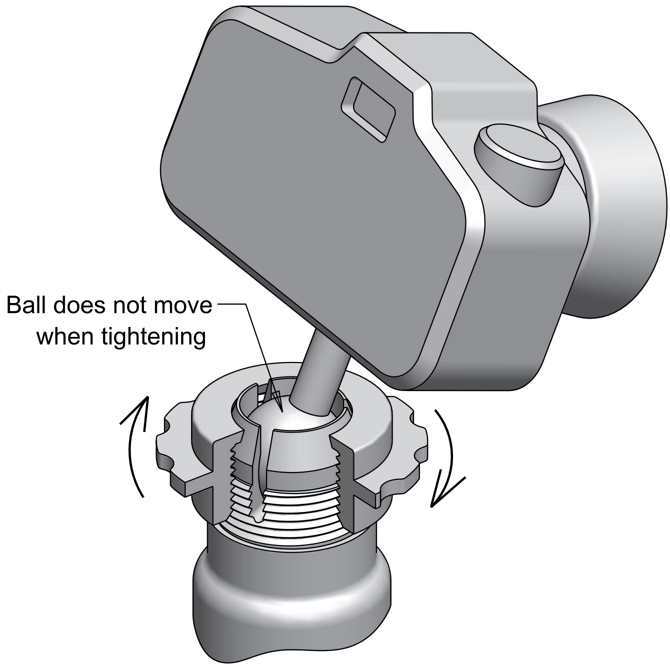

Figure 2-2 Slotted collet for ball-and-socket tripod head. Locking function is independent of locating function.

But a deeper look suggests decomposing functions differently. The ball-and-socket joint allows rotation about three different (and non-stationary) axes simultaneously: pitch, roll, and yaw. Perhaps these should be independent?

"But this is a great design!" you say. "I have infinite possibilities for adjustment, and I can position the camera—all in one motion—right where I want it!" Yes, but you are forced to adjust pitch, roll, and yaw every time you adjust any one of them. No functional independence there!

A different solution results by decomposing functions as follows:

1. Position and fix pitch
2. Position and fix roll
3. Position and fix yaw

An example of one common approach to this set of functional requirements is a multi-axis pan head for camera tripods, shown in Figure 2-3.

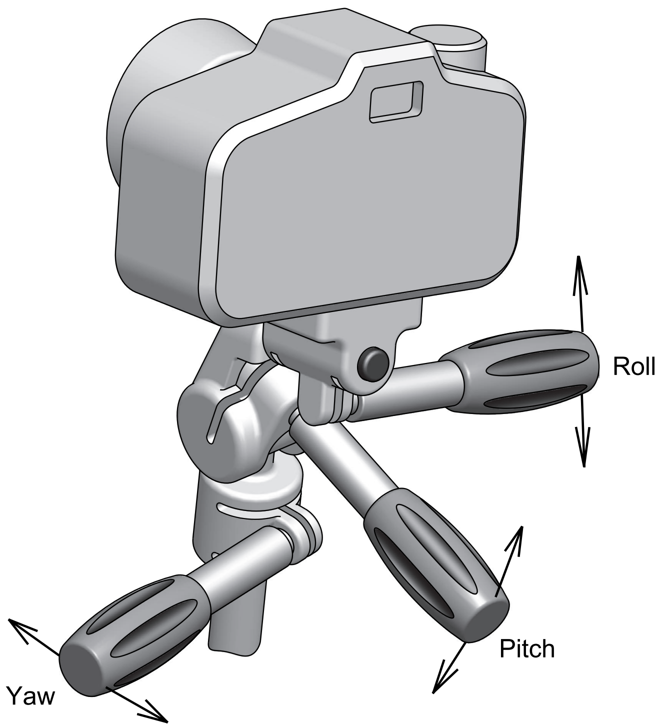

Figure 2-3 Multi-axis pan head camera mount has functional independence of all three rotational locking and positioning functions.

The nice thing here is that you adjust only what needs it, which is less awkward for fine adjust, if sometimes slower, than a ball-and-socket. The pitch, roll, and yaw adjustments are functionally independent.

If you are familiar with these examples, you already know that camera tripods with both ball-and-socket heads and multi-axis heads are successfully marketed. Users adapt to the functional dependence, and some, apparently, prefer flexibility over a more theoretically pure solution. Perhaps each is superior to the other for a specific use and specific user.

Accept that everything in design, including independence of function, is a compromise. You need not take an unyielding posture to this or any other rule of design, but always understand how and why you yield.

Finally, seek independence of the functions of a device, but do not preclude combining functions within parts. The locking handles of the multi-axis pan head offer a perfect example of this. These handles are used for two functions, positioning and locking (Figure 2-4).

Combining multiple functions into single parts saves cost and reduces complexity. This makes the cited example quite elegant and intuitive.

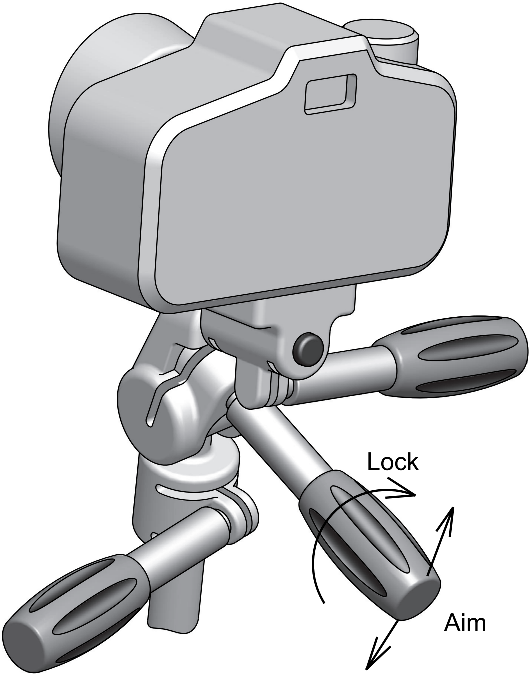

Figure 2-4 Combined adjustment and locking functionality reduces parts count and operates intuitively.