

## Kompilering, assemblering och länkning

- Kompilering, assemblering och länkning omvandlar *källkoden* till ett format lämpligt att köra på *målsystemet*.
- Kallas *korskompilering* (eng. *cross compiling*) då *host* och *målsystem* är olika plattformar.
- Omvandlingen sker i tre steg, *kompilering* och *assemblering*, *länkning* och *relokering*.

## Datorsystemteknik för D2, ICT2, E3 och Mek3

Nicholas Wickström

Högskolan i Halmstad  
Sverige

## Kompilering/Assemblering

- Omvandlar källkod till opkoder (processorspecifika).
- Resultatet är en objektfil, vanligen i COFF (Common Object File Format) eller ELF (Extended Linker Format) format.
- Objektfilen är uppdelad i ett antal sektioner.

## Outline

- Sammanfattning av föregående fl.
- Flöde
- Kompilering, assemblering och länkning
- Nedladdning på målsystemet

## Relokering

- Oftast integrerad med länkaren.
- Tilldelar adresser till funktioner och variabler.
- Man kan styra var de olika sektionerna skall hamna i minnet med "script".

Datorsystemteknik – p.7/8

## Nedladdning på målsystemet

- EPROM (flash)
- "Remote Debugger" (JTAG, Monitor)

Datorsystemteknik – p.8/8

## Sektioner

- **.text** - Exekverbar kod.
- **.data** - Initierade globala variabler.
- **.bss** - Ej initierade globala variabler.
- **.const** - Strängar och konstanter (dekl med **const**).
- **.stack** - Minne för system stacken.
- **.system** - Minne för dynamiskt allokerad data (heap).

Datorsystemteknik – p.5/8

## Länkning

- Länkarens uppgift är att sammanfoga objektfilerna från kompileringen.
- Efter länkning ligger alla objektfilers **.text** sektion i en gemensam **.text** sektion (pss med andra sektioner).
- "Unresolved Symbols"
- Efter länkningen är allt klart, förutom tilldelning av adresser till funktioner och variabler.

Datorsystemteknik – p.6/8

