

Why does Semi-Monthly payroll force the use of 0.5x Overtime Reporting?

Last Modified on 05/22/2023 3:45 pm MDT

Teamwork limits users to 0.5 overtime reporting with Semi-Monthly payroll accounting.

If you use Bi-Weekly payroll, you have the choice between 0.5 and 1.5.

Semi-monthly overtime calculation is complex, and proper accounting is often overlooked.

Teamwork has built a system which is intended to provide proper accounting not only for straightforward single-location, single-wage scenarios, but also multi-location and multi-wage scenarios.

When semi-monthly pay cycles are used, it is likely that the start of a pay cycle, i.e the 1st or 16th of a month, will fall mid-way through a workweek. It is advised (universally from what we've seen) that overtime is calculated only when the week is concluded. Thus, overtime for the partial week will be paid in the *FOLLOWING* pay period. Many scenarios can arise which make accounting fickle. By using a *half-time overtime method*, all instances can be solved. This means that if an employee has 40 regular hours and 10 overtime hours, then they would be paid as follows: 50 hours of regular pay + 10 hours of half pay. This may seem like the same thing as 40 hours of regular pay + 10 hours of 1.5 pay, but I will provide an example below which shows how the "1.5 x pay" method cannot hold up. While I've created an example of an unlikely schedule, it is to illustrate that an accounting system should theoretically handle any situation.

In the following example

- In the first pay period – the 1st through the 15th, the five hours of overtime accrued by the 15th of the month cannot yet be calculated. However, the employee needs to be paid for the 45 hours that they've worked.
- In the second pay period, the 16th through the 31st, the 10 hours of overtime from the first pay period are now calculated and paid. All ten hours cannot be counted at a 1.5 x rate, because this would be duplicitous to the five hours already paid the 1st through the 15th.
- Paying a regular wage for all hours, and then an additional overtime wage of 0.5 x the overtime hours solves this, in addition to all scenarios of properly weighting multi-wage, double-time, multi-location, pay-raise instances.

| | | | | | | | | Week | | | | |
|-------------------|--------|--------|---------|-----------|----------|--------|----------|-------|---------|----|-------------|-------------|
| | Sunday | Monday | Tuesday | Wednesday | Thursday | Friday | Saturday | Total | Regular | OT | | |
| | 4th | 5th | 6th | 7th | 8th | 9th | 10th | | | | | |
| <i>hrs worked</i> | | | | | | | | 0 | 0 | 0 | | |
| | 11th | 12th | 13th | 14th | 15th | 16th | 17th | | | | Hours Paid | OT Paid |
| | | | | | | | | | | | 1st - 15th | 1st - 15th |
| <i>hrs worked</i> | 5 | 10 | 10 | 10 | 10 | 5 | | 50 | 40 | 10 | 45 | 0 |
| | 18th | 19th | 20th | 21st | 22nd | 23rd | 24th | | | | | |
| <i>hrs worked</i> | | | 2 | | | | | 2 | 2 | 0 | | |
| | 25th | 26th | 27th | 28th | 29th | 30th | 31st | | | | Hours Paid | OT Paid |
| | | | | | | | | | | | 16th - 31st | 16th - 31st |
| <i>hrs worked</i> | | | 2 | | | | | 2 | 2 | 0 | 4 | 10 |

Following is an article which references the FLSA method of using 0.5. The case scenario is different than ours, but the mentality is the same.

<https://www.opm.gov/policy-data-oversight/pay-leave/pay-administration/fact-sheets/how-to-compute-flsa-overtime-pay/>

If you wish to have things align in a more straightforward manner, please consider a bi-weekly payroll. All overtime/paychecks/pay periods align. The 1.5 method can be employed.

To setup 0.5x overtime in your system, you must create the payroll item type for OT as 0.5x.